Network Working Group   D. Farinacci
Internet-Draft   V. Fuller
Intended status: Experimental   D. Meyer
Expires: October 27, 2010   D. Lewis
                                                          cisco Systems
                                                         April 25, 2010

              Locator/ID Separation Protocol (LISP)
                       draft-ietf-lisp-07

Abstract

   This draft describes a simple, incremental, network-based protocol to
   implement separation of Internet addresses into Endpoint Identifiers
   (EIDs) and Routing Locators (RLOCs). This mechanism requires no
   changes to host stacks and no major changes to existing database
   infrastructures. The proposed protocol can be implemented in a
   relatively small number of routers.

   This proposal was stimulated by the problem statement effort at the
   Amsterdam IAB Routing and Addressing Workshop (RAWS), which took
   place in October 2006.

Status of this Memo

   This Internet-Draft is submitted to IETF in full conformance with the
   provisions of BCP 78 and BCP 79.

   Internet-Drafts are working documents of the Internet Engineering
   Task Force (IETF), its areas, and its working groups. Note that
   other groups may also distribute working documents as Internet-
   Drafts.

   Internet-Drafts are draft documents valid for a maximum of six months
   and may be updated, replaced, or obsoleted by other documents at any
   time. It is inappropriate to use Internet-Drafts as reference
   material or to cite them other than as "work in progress."

   The list of current Internet-Drafts can be accessed at
   http://www.ietf.org/ietf/1id-abstracts.txt.

   The list of Internet-Draft Shadow Directories can be accessed at
   http://www.ietf.org/shadow.html.

   This Internet-Draft will expire on October 27, 2010.

Copyright Notice

Table of Contents

1. 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].

2. Introduction

   Many years of discussion about the current IP routing and addressing
   architecture have noted that its use of a single numbering space (the
   "IP address") for both host transport session identification and
   network routing creates scaling issues (see [CHIAPPA] and [RFC1498]).
   A number of scaling benefits would be realized by separating the
   current IP address into separate spaces for Endpoint Identifiers
   (EIDs) and Routing Locators (RLOCs); among them are:

   1. Reduction of routing table size in the "default-free zone" (DFZ).
       Use of a separate numbering space for RLOCs will allow them to be
       assigned topologically (in today's Internet, RLOCs would be
       assigned by providers at client network attachment points),
       greatly improving aggregation and reducing the number of
       globally-visible, routable prefixes.

   2. More cost-effective multihoming for sites that connect to
       different service providers where they can control their own
       policies for packet flow into the site without using extra
       routing table resources of core routers.

   3. Easing of renumbering burden when clients change providers.
       Because host EIDs are numbered from a separate, non-provider-
       assigned and non-topologically-bound space, they do not need to
       be renumbered when a client site changes its attachment points to
       the network.

   4. Traffic engineering capabilities that can be performed by network
       elements and do not depend on injecting additional state into the
       routing system. This will fall out of the mechanism that is used
       to implement the EID/RLOC split (see Section 4).

   5. Mobility without address changing. Existing mobility mechanisms
       will be able to work in a locator/ID separation scenario. It
       will be possible for a host (or a collection of hosts) to move to
       a different point in the network topology either retaining its
       home-based address or acquiring a new address based on the new
       network location. A new network location could be a physically
       different point in the network topology or the same physical
       point of the topology with a different provider.

   This draft describes protocol mechanisms to achieve the desired
   functional separation. For flexibility, the mechanism used for
   forwarding packets is decoupled from that used to determine EID to
   RLOC mappings. This document covers the former. For the latter, see
   [CONS], [ALT], [EMACS], [RPMD], and [NERD]. This work is in response
   to and intended to address the problem statement that came out of the

RAWS effort [RFC4984].

The Routing and Addressing problem statement can be found in [RADIR].

This draft focuses on a router-based solution. Building the solution into the network will facilitate incremental deployment of the technology on the Internet. Note that while the detailed protocol specification and examples in this document assume IP version 4 (IPv4), there is nothing in the design that precludes use of the same techniques and mechanisms for IPv6. It should be possible for IPv4 packets to use IPv6 RLOCs and for IPv6 EIDs to be mapped to IPv4 RLOCs.

Related work on host-based solutions is described in Shim6 [RFC5533] and HIP [RFC4423]. Related work on a router-based solution is described in [GSE]. This draft attempts to not compete or overlap with such solutions and the proposed protocol changes are expected to complement a host-based mechanism when Traffic Engineering functionality is desired.

Some of the design goals of this proposal include:

1. Require no hardware or software changes to end-systems (hosts).

2. Minimize required changes to Internet infrastructure.

3. Be incrementally deployable.

4. Require no router hardware changes.

5. Minimize the number of routers which have to be modified. In particular, most customer site routers and no core routers require changes.

6. Minimize router software changes in those routers which are affected.

7. Avoid or minimize packet loss when EID-to-RLOC mappings need to be performed.

There are 4 variants of LISP, which differ along a spectrum of strong to weak dependence on the topological nature and possible need for routability of EIDs. The variants are:

   LISP 1: uses EIDs that are routable through the RLOC topology for
      bootstrapping EID-to-RLOC mappings. [LISP1] This was intended as
      a prototyping mechanism for early protocol implementation. It is
      now deprecated and SHOULD NOT be deployed.

   LISP 1.5: uses EIDs that are routable for bootstrapping EID-to-RLOC
      mappings; such routing is via a separate topology.

   LISP 2: uses EIDS that are not routable and EID-to-RLOC mappings are
      implemented within the DNS. [LISP2]

   LISP 3: uses non-routable EIDs that are used as lookup keys for a
      new EID-to-RLOC mapping database. Use of Distributed Hash Tables
      [DHTs] [LISPDHT] to implement such a database would be an area to
      explore. Other examples of new mapping database services are
      [CONS], [ALT], [RPMD], [NERD], and [APT].

   This document on LISP 1.5, and LISP 3 variants, both of which rely on
   a router-based distributed cache and database for EID-to-RLOC
   mappings. The LISP 1.0 mechanism works but does not allow reduction
   of routing information in the default-free-zone of the Internet. The
   LISP 2 mechanisms are put on hold and may never come to fruition
   since it is not architecturally pure to have routing depend on
   directory and directory depend on routing. The LISP 3 mechanisms
   will be documented elsewhere but may use the control-plane options
   specified in this specification.

3. **Definition of Terms**

   Provider Independent (PI) Addresses: 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 is therefore not topologically aggregatable in the routing
      system.

   Provider Assigned (PA) Addresses: a block of IP addresses that are
      assigned to a site by each service provider to which a site
      connects. Typically, each block is sub-block of a service
      provider CIDR block and is aggregated into the larger block before
      being advertised into the global Internet. Traditionally, IP
      multihoming has been implemented by each multi-homed 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):   the IPv4 or IPv6 address of an egress
      tunnel router (ETR). It is the output of a 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):   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 an destination address
      today, for example through a DNS lookup or SIP exchange. The
      source EID is obtained via existing mechanisms used to set a
      host's "local" IP address. 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. When used in
      discussions with other Locator/ID separation proposals, a LISP EID
      will be called a "LEID". Throughout this document, any references
      to "EID" refers to an LEID.

EID-prefix: A power-of-2 block of EIDs which are allocated to a
   site by an address allocation authority. EID-prefixes are
   associated with a set of RLOC addresses which 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 smaller EID-
   prefix. A globally routed address block (whether PI or PA) is not
   an EID-prefix. However, a globally routed address block may be
   removed from global routing and reused as an EID-prefix. A site
   that receives an explicitly allocated EID-prefix may not use that
   EID-prefix as a globally routed prefix assigned to RLOCs.

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):   a router which accepts an IP packet
   with a single IP header (more precisely, an IP packet that does
   not contain a LISP header). The router treats this "inner" 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 hosts
   supplied EID).

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

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. Used synonymously with the term "Tunnel
Router". For example, "An xTR can be located at the Customer Edge
(CE) router", meaning both ITR and ETR functionality is at the CE
router.

EID-to-RLOC Cache: 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: 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. That is, the EID-prefixes for the site and locator-
set for each EID-prefix MUST be the same on all ETRs so they
consistently send Map-Reply messages with the same database
mapping contents.

Recursive Tunneling: 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 refers to dynamic
encapsulating tunnels and never are they statically configured.

Reencapsulating Tunnels: when a packet has no more than one LISP IP
header (two IP headers total) and when it needs to be diverted to
new RLOC, an ETR can decapsulate the packet (remove the LISP
header) and prepends a new tunnel header, with new RLOC, on to the
packet. 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
refers to dynamic encapsulating tunnels and never are they

statically configured.

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

Address Family Identifier (AFI):   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 [RFC1700] for details. An
   AFI value of 0 used in this specification indicates an unspecified
   encoded address where the the length of the address is 0 bytes
   following the 16-bit AFI value of 0.

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.

Data Probe: 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. 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.

Proxy ITR (PITR):   also known as a PTR is defined and described in
   [INTERWORK], a PITR acts like an ITR but does so on behalf of non-
   LISP sites which send packets to destinations at LISP sites.

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

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 limits off-path data insertion.

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

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, 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.

   This design introduces "Tunnel Routers", which prepends LISP headers
   on host-originated packets and strip 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 egress tunnel
   router strips the new header. The ITR performs EID-to-RLOC lookups
   to determine the routing path to the the ETR, which has the RLOC as
   one of its IP addresses.

   Some basic rules governing LISP are:

   o End-systems (hosts) only send to addresses which are EIDs. They
      don't know addresses are EIDs versus RLOCs but assume packets get
      to LISP routers, which in turn, deliver packets to the destination
      the end-system has specified.

   o EIDs are always IP addresses assigned to hosts.

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

   o RLOCs are always IP addresses assigned to routers; preferably,
      topologically-oriented addresses from provider CIDR blocks.

   o 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 SSH, TELNET, or 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.

   o EIDs are not expected to be usable for global end-to-end
      communication in the absence of an EID-to-RLOC mapping operation.
      They are expected to be used locally for intra-site communication.

   o EID prefixes are likely to be hierarchically assigned in a manner
      which is optimized for administrative convenience and to
      facilitate scaling of the EID-to-RLOC mapping database. The
      hierarchy is based on a address allocation hierarchy which is not
      dependent on the network topology.

   o EIDs may also be structured (subnetted) in a manner suitable for
      local routing within an autonomous system.

   An additional LISP header may be prepended to packets by a transit
   router (i.e. TE-ITR) when re-routing of the path for a packet is
   desired. An obvious instance of 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 ingress tunnel router and the RLOC it
   uses for the new prepended header would be either a TE-ETR within the
   ISP (along 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).

   This specification mandates that no more than two LISP headers get
   prepended to a packet. This avoids excessive packet overhead as well
   as possible encapsulation loops. It is believed two headers is
   sufficient, where the first prepended header is used at a site for
   Location/Identity separation and 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 egress tunnel router might be the last-hop
   router directly-connected to the destination host. Another example,
   perhaps for a VPN service out-sourced 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:

   o Source host "host1.abc.com" is sending a packet to
      "host2.xyz.com", exactly what host1 would do if the site was not
      using LISP.

   o Each site is multi-homed, 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.

   o 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 LISP site.

   o Map-Requests can be sent on the underlying routing system topology
      (LISP 1.0) or over an alternative topology [ALT].

   o Map-Replies are sent on the underlying routing system topology.

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

   1. host1.abc.com wants to open a TCP connection to host2.xyz.com.
       It does a DNS lookup on host2.xyz.com. An A/AAAA record is
       returned. This address is the destination EID. The locally-
       assigned address of host1.abc.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 EID destination 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 [ALT] or
       [CONS] for possible solutions.

   3. The ITR will send a LISP Map-Request. Map-Requests SHOULD be
       rate-limited.

   4. In LISP 1.0, the Map-Request packet is routed through the
       underlying routing system. In LISP 1.5, the Map-Request packet
       is routed on an alternate logical topology. 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 put in the ITR's EID-to-
RLOC mapping cache (this is the on-demand cache, the cache where
entries time out due to inactivity).

7. Subsequent packets from host1.abc.com to host2.xyz.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,
the packet may be sent to a different ETR than the one which
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), strips the LISP
header and forwards the packets to the attached destination host.

In order to eliminate 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
ITR and the ETR may also influence the decision the other makes in
selecting an RLOC. See Section 6 for more details.

5. Tunneling Details

   This section describes the LISP Data Message which defines the
   tunneling header used to encapsulate IPv4 and IPv6 packets which
   contain EID addresses. Even though the following formats illustrate
   IPv4-in-IPv4 and IPv6-in-IPv6 encapsulations, the other 2
   combinations are supported as well.

   Since additional tunnel headers are prepended, the packet becomes
   larger and in theory can exceed the MTU of any link traversed from
   the ITR to the ETR. It is recommended, in IPv4 that packets do not
   get fragmented as they are encapsulated by the ITR. Instead, the
   packet is dropped and an ICMP Too Big message is returned to the
   source.

   Based on informal surveys of large ISP traffic patterns, it appears
   that most transit paths can accommodate a path MTU of at least 4470
   bytes. The exceptions, in terms of data rate, number of hosts
   affected, or any other metric are expected to be vanishingly small.

   To address MTU concerns, mainly raised on the RRG mailing list, the
   LISP deployment process will include collecting data during its pilot
   phase to either verify or refute the assumption about minimum
   available MTU. If the assumption proves true and transit networks
   with links limited to 1500 byte MTUs are corner cases, it would seem
   more cost-effective to either upgrade or modify the equipment in
   those transit networks to support larger MTUs or to use existing
   mechanisms for accommodating packets that are too large.

   For this reason, there is currently no plan for LISP to add any new
   additional, complex mechanism for implementing fragmentation and
   reassembly in the face of limited-MTU transit links. If analysis
   during LISP pilot deployment reveals that the assumption of
   essentially ubiquitous, 4470+ byte transit path MTUs, is incorrect,
   then LISP can be modified prior to protocol standardization to add
   support for one of the proposed fragmentation and reassembly schemes.
   Note that two simple existing schemes are detailed in Section 5.4.

5.1. LISP IPv4-in-IPv4 Header Format

```
         0                   1                   2                   3
         0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     /  |Version|  IHL  |Type of Service|          Total Length         |
    /   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |   |         Identification        |Flags|      Fragment Offset    |
    |   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   OH  |  Time to Live | Protocol = 17 |         Header Checksum        |
    |   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |   |                 Source Routing Locator                        |
     \  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      \ |               Destination Routing Locator                     |
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     /  |        Source Port = xxxx     |       Dest Port = 4341         |
   UDP  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     \  |           UDP Length          |        UDP Checksum            |
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   L    |N|L|E|V|I|flags|            Nonce/Map-Version                   |
   I  \ +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   S  / |                 Instance ID/Locator Status Bits               |
   P    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     /  |Version|  IHL  |Type of Service|          Total Length         |
    /   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |   |         Identification        |Flags|      Fragment Offset    |
    |   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   IH  |  Time to Live |    Protocol   |         Header Checksum         |
    |   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |   |                       Source EID                              |
     \  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      \ |                     Destination EID                           |
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

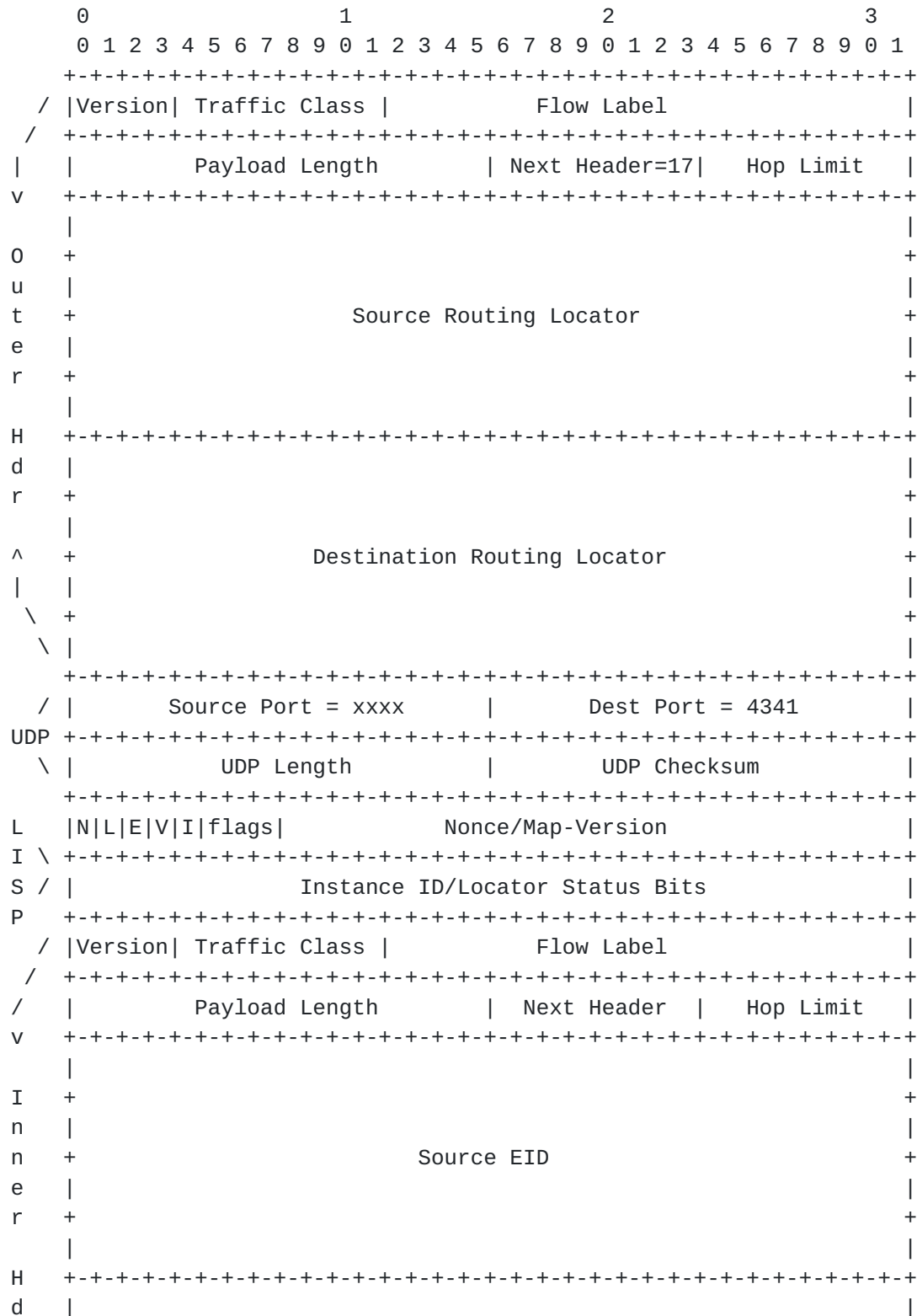## 5.2. LISP IPv6-in-IPv6 Header Format

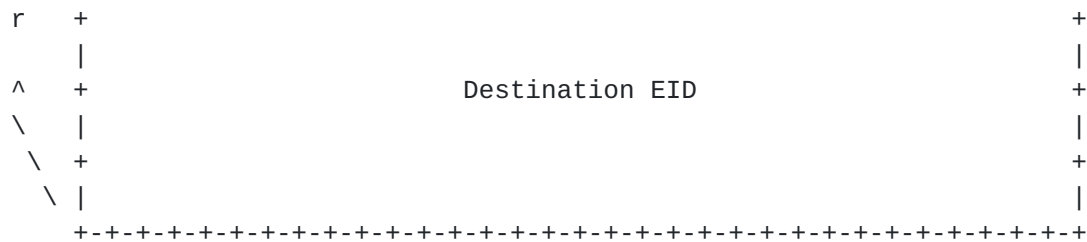```
        0   1   2   3
        0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
       +-+
     / |Version| Traffic Class | Flow Label |
    / +-+
   | | Payload Length | Next Header=17| Hop Limit |
   v   +-+
       | |
   O   +   +
   u | |
   t   +   Source Routing Locator   +
   e | |
   r   +   +
       | |
   H   +-+
   d | |
   r   +   +
       | |
   ^   +   Destination Routing Locator   +
   | | |
    \ +   +
     \ | |
       +-+
     / | Source Port = xxxx | Dest Port = 4341 |
   UDP +-+
     \ | UDP Length | UDP Checksum |
       +-+
   L |N|L|E|V|I|flags| Nonce/Map-Version |
   I \ +-+
   S / | Instance ID/Locator Status Bits |
   P   +-+
     / |Version| Traffic Class | Flow Label |
    / +-+
   / | Payload Length | Next Header | Hop Limit |
   v   +-+
       | |
   I   +   +
   n | |
   n   +   Source EID   +
   e | |
   r   +   +
       | |
   H   +-+
   d | |
```

```
r   +                                                            +
    |                                                            |
^   +                     Destination EID                        +
 \  |                                                            |
  \ +                                                            +
   \ |                                                            |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

5.3. Tunnel Header Field Descriptions

   Inner Header: is the inner header, preserved from the datagram
      received from the originating host. The source and destination IP
      addresses are EIDs.

   Outer Header: is the outer 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 DF bit of the Flags field is set to 0 when the method in
      Section 5.4.1 is used and set to 1 when the method in
      Section 5.4.2 is used.

   UDP Header: contains a ITR selected source port when encapsulating a
      packet. See Section 6.4 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: this field SHOULD be transmitted as zero by an ITR for
      either IPv4 [RFC0768] or IPv6 encapsulation [UDP-TUNNELS]. 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: for an IPv4 encapsulated packet, the inner header Total
      Length plus the UDP and LISP header lengths are used. For an IPv6
      encapsulated packet, the inner header Payload Length plus the size
      of the IPv6 header (40 bytes) plus the size of the UDP and LISP

headers are used.   The UDP header length is 8 bytes.

N: this 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 contains
   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: this 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.

```
  x 1 x x 0 x x x
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |N|L|E|V|I|flags|            Nonce/Map-Version                  |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                      Locator Status Bits                      |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

E: this is the echo-nonce-request bit. When this bit is set to 1,
   the N bit MUST be 1. This bit SHOULD be ignored and has no
   meaning when the N bit is set to 0. See Section 6.3.1 for
   details.

V: this is the Map-Version present bit. When this bit is set to 1,
   the N bit MUST be 0. Refer to Section 6.5.3 for more details.
   This bit indicates that the first 4 bytes of the LISP header is
   encoded as:

```
  0 x 0 1 x x x x
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |N|L|E|V|I|flags|  Source Map-Version   |   Dest Map-Version    |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                  Instance ID/Locator Status Bits             |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

I: this 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
   last 4 bytes of the LISP header would look like:

```
   x x x x 1 x x x
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |N|L|E|V|I|flags|              Nonce/Map-Version                |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |                 Instance ID                 |     LSBs        |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

flags: this 3-bit field is reserved for future flag use. It is set
   to 0 on transmit and ignored on receipt.

LISP Nonce: is a 24-bit value that is randomly generated by an ITR
   when the N-bit is set to 1. 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: in the LISP header are 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 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 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 other ITRs
   at the same site. When a site has multiple EID-prefixes which
   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.

When doing Recursive Tunneling or ITR/PTR encapsulation:

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

o 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 caveat, see below).

When doing Re-encapsulated Tunneling:

   o The new outer header Time to Live field SHOULD be copied from the
      stripped outer header Time to Live field.

   o The new outer header Type of Service field SHOULD be copied from
      the stripped OH header Type of Service field (with one caveat, see
      below).

   Copying the 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.

   The 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
   Congestion Experienced (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

   In the event that the MTU issues mentioned above prove to be more
   serious than expected, this section proposes 2 simple mechanisms to
   deal with large packets. One is stateless using IP fragmentation and
   the other is stateful using Path MTU Discovery [RFC1191].

   It is left to the implementor to decide if the stateless or stateful
   mechanism SHOULD be implemented. Both or neither can be decided as
   well since it is a local decision in the ITR regarding how to deal
   with MTU issues. Sites can interoperate with differing mechanisms.

   Both stateless and stateful mechanisms also apply to Reencapsulating
   and Recursive Tunneling. So any actions below referring to an ITR
   also apply to an 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 an architectural constant S for the maximum size of a
    packet, in bytes, an ITR would receive from a source inside of
    its site.

2. Define L to be the maximum size, in bytes, a packet of size S
    would be after the ITR prepends the LISP header, UDP header, and
    outer network layer header of size H.

3. Calculate: S + H = L.

When an ITR receives a packet from a site-facing interface and adds H
bytes worth of encapsulation to yield a packet size greater than L
bytes, 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. This will ensure that the new, encapsulated
packets are of size (S/2 + H), which is always below the effective
tunnel MTU.

When an ETR receives encapsulated fragments, it treats them as two
individually encapsulated packets. It strips the LISP headers 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.

This behavior is performed by the ITR when the source host originates
a packet with the DF field of the IP header is 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 sends 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
    mapping cache entry. The effective MTU is what the core network
    can deliver along the path between ITR and ETR.

2. When an IPv6 encapsulated packet or an IPv4 encapsulated packet
    with 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 mapping 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 mapping 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 [LCAF] for details for a possible address encoding. The
LCAF encoding is an area for further study.

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
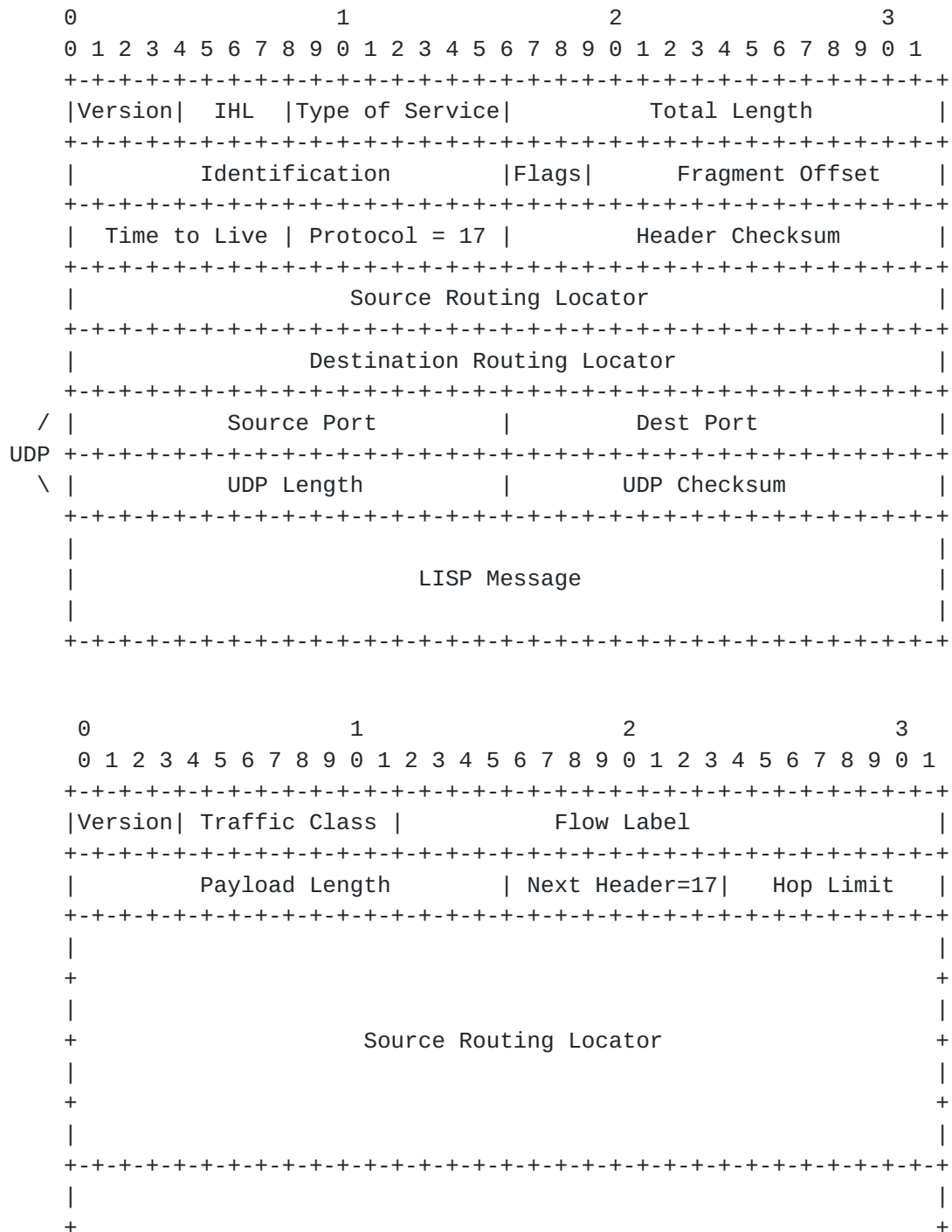copied to the Instance ID field of the LISP header and the I-bit is
set to 1.

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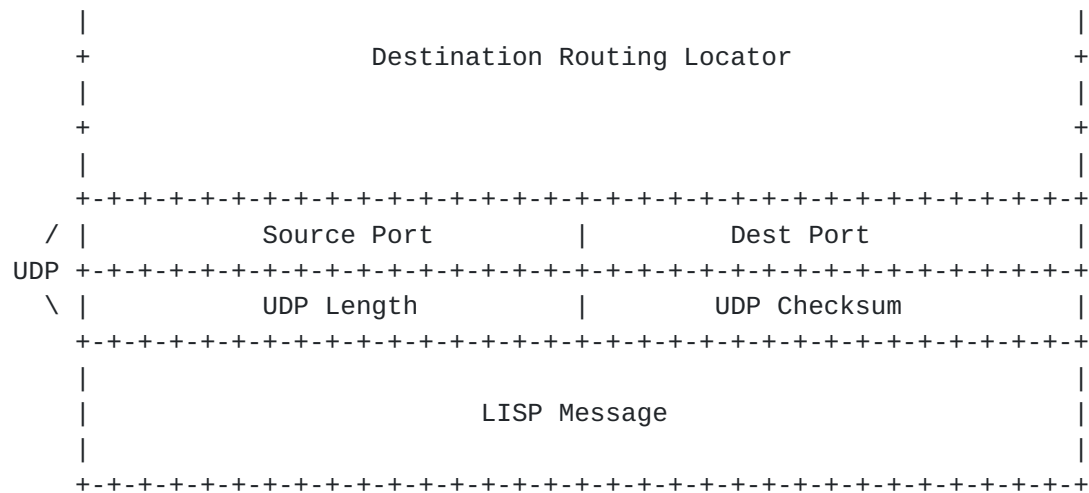to use for the inner destination EID lookup.

Some examples of the 24-bit value to copy or map into the Instance ID
field could be a 802.1Q VLAN tag or a configured VRF-ID value.

6. **EID-to-RLOC Mapping**

6.1.   **LISP IPv4 and IPv6 Control Plane Packet Formats**

   The following new UDP packet types are used to retrieve EID-to-RLOC
   mappings:

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |Version|  IHL  |Type of Service|          Total Length         |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |         Identification        |Flags|     Fragment Offset     |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    | Time to Live | Protocol = 17 |         Header Checksum        |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                    Source Routing Locator                     |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                  Destination Routing Locator                  |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  / |           Source Port         |           Dest Port           |
UDP +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  \ |           UDP Length          |          UDP Checksum         |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                                                               |
    |                         LISP Message                          |
    |                                                               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+


     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |Version| Traffic Class |           Flow Label                  |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |         Payload Length        | Next Header=17|   Hop Limit   |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                                                               |
    +                                                               +
    |                                                               |
    +                     Source Routing Locator                    +
    |                                                               |
    +                                                               +
    |                                                               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                                                               |
    +                                                               +
```

```
        |                                                         |
        +                 Destination Routing Locator            +
        |                                                         |
        +                                                         +
        |                                                         |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    /  |          Source Port          |         Dest Port        |
UDP +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    \  |          UDP Length           |        UDP Checksum      |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                                                         |
       |                        LISP Message                     |
       |                                                         |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

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.

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 ECM control messages. It MUST be checked
on receipt and if the checksum fails, the packet MUST be dropped.

LISP-CONS [CONS] use TCP to send LISP control messages. The format
of control messages includes the UDP header so the checksum and
length fields can be used to protect and delimit message boundaries.

This main LISP specification is the authoritative source for message
format definitions for the Map-Request and Map-Reply messages.

6.1.1. **LISP Packet Type Allocations**

   This section will be the authoritative source for allocating LISP
   Type values. 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 Encapsulated Control Message: 8   b'1000'

6.1.2. **Map-Request Message Format**


```
        0                   1                   2                   3
        0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |Type=1 |A|M|P|S|     Reserved      |   IRC   | Record Count  |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                         Nonce . . .                         |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                     . . . Nonce                             |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |         Source-EID-AFI        |   Source EID Address  ...   |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |         ITR-RLOC-AFI 1        |    ITR-RLOC Address 1  ...   |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                              ...                            |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |         ITR-RLOC-AFI n        |    ITR-RLOC Address n  ...   |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   /  |   Reserved    | EID mask-len  |        EID-prefix-AFI        |
  Rec +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   \  |                       EID-prefix  ...                        |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                   Map-Reply Record  ...                     |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                     Mapping Protocol Data                   |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```


   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.

M: When set, it indicates 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 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 SMR bit. See Section 6.5.2 for details.

Reserved: Set to 0 on transmission and ignored on receipt.

IRC: This 5-bit field is the ITR-RLOC Count which encodes the number
   of (ITR-RLOC-AFI, ITR-RLOC Address) fields present in this
   message. 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, where IRC value 0 means
   an ITR-RLOC address count of 1, an IRC value of 1 means an ITR-
   RLOC address count of 2, and so on up to an IRC value of 31, which
   means an ITR-RLOC address count of 32.

Record Count: 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: An 8-byte 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 depends critically 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: Address family of the "Source EID Address" field.

   Source EID Address: This is the EID of the source host which
      originated the packet which is invoking this Map-Request. When
      Map-Requests are used for refreshing a map-cache entry or for
      RLOC-probing, the value 0 is used.

   ITR-RLOC-AFI: Address family of the "ITR-RLOC Address" field that
      follows this field.

   ITR-RLOC Address: 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.

   EID mask-len: Mask length for EID prefix.

   EID-prefix-AFI: Address family of EID-prefix according to [RFC5226]

   EID-prefix: 4 bytes if an IPv4 address-family, 16 bytes if 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
      the "Record" field 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 which will receive this Map-Request to
      cache the data if it chooses to do so.

   Mapping Protocol Data: See [CONS] or [ALT] for details. This field
      is optional and present when the UDP length indicates there is
      enough space in the packet to include it.

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 destination-EID from the
   packet which had a mapping cache lookup failure. For the latter 2
   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 if
   the Map-Request is using an IPv4 versus IPv6 header, respectively.

In all cases, the UDP source port number for the Map-Request message
is a randomly allocated 16-bit value and the UDP destination port
number is set to the well-known destination port number 4342. A
successful Map-Reply updates the cached 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
[LISP-MS].

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. 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 in the "piggybacked" mapping data.

6.1.4. Map-Reply Message Format

```
      0                   1                   2                   3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |Type=2 |P|E|         Reserved              | Record Count  |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                         Nonce . . .                           |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                         . . . Nonce                           |
 +-> +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |   |                         Record  TTL                           |
 |   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 R   | Locator Count | EID mask-len  | ACT |A|      Reserved         |
 e   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 c   | Rsvd  |  Map-Version Number   |        EID-AFI                |
 o   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 r   |                          EID-prefix                           |
 d   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |  /|   Priority    |    Weight     |  M Priority   |   M Weight    |
 | L +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 | o |        Unused Flags     |L|p|R|          Loc-AFI              |
 | c +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |  \|                          Locator                             |
 +-> +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                     Mapping Protocol Data                     |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

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: Indicates that the ETR which 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.

Reserved: Set to 0 on transmission and ignored on receipt.

Record Count: 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: 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.

Record TTL: 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: The number of Locator entries. A locator entry
   comprises what is labeled above as 'Loc'. The locator count can
   be 0 indicating there are no locators for the EID-prefix.

EID mask-len: Mask length for EID prefix.

ACT: This 3-bit field describes negative Map-Reply actions. 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 PTR when a destination EID
   matches a negative mapping 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) Drop: The packet is dropped silently.

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

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

A: The Authoritative bit, when sent by a UDP-based message is always
   set to 1 by an ETR. See [CONS] for TCP-based Map-Replies. When a
   Map-Server is proxy Map-Replying [LISP-MS] 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 6.5.3 for more
   details.

 EID-AFI: Address family of EID-prefix according to [RFC5226].

 EID-prefix: 4 bytes if an IPv4 address-family, 16 bytes if 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 percentage of total unicast packets that match the
   mapping entry. If a non-zero weight value is used for any RLOC,
   then all RLOCs MUST use a non-zero weight value and then the sum
   of all weight values MUST equal 100. If a zero value is used for
   any RLOC weight, then all weights MUST be zero and the receiver of
   the Map-Reply will decide how to load-split traffic. See
   Section 6.4 for a suggested hash algorithm to distribute load
   across locators with same priority and equal weight values. When
   a single RLOC exists in a mapping entry, the weight value MUST be
   set to 100 and ignored on receipt.

 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.

 M Weight: when priorities are the same for multiple RLOCs, the
   weight indicates how to balance building multicast distribution
   trees across multiple ITRs. The weight is encoded as a percentage
   of total number of trees build to the source site identified by
   the EID-prefix. If a non-zero weight value is used for any RLOC,
   then all RLOCs MUST use a non-zero weight value and then the sum
   of all weight values MUST equal 100. If a zero value is used for
   any RLOC weight, then all weights MUST be zero and the receiver of
   the Map-Reply will decide how to distribute multicast state across
   ITRs.

Unused Flags: 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 [LISP-MS] 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: when this bit is set, the locator is known to be reachable from
   the Map-Reply sender's perspective.

Locator: an IPv4 or IPv6 address (as encoded by the 'Loc-AFI' field)
   assigned to an ETR or router acting as a proxy replier for the
   EID-prefix. 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.

Mapping Protocol Data: See [CONS] or [ALT] for details. This field
   is optional and present when the UDP length indicates there is
   enough space in the packet to include it.

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

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

Note that a Map-Reply may contain different EID-prefix granularity
(prefix + length) than the Map-Request which 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 but not by removing the less-specific prefix rather by
adding the more-specific prefixes which 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 longest 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, only
the more specifics (note in the second example above 10.0.0.0/8 was
not returned for requesting EID 10.1.5.5) entries 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 specifics exist. When a 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.

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. This type of a Map-
Reply is called a Negative Map-Reply. Negative Map-Replies convey

special actions by the sender to the ITR or PTR which 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 PTR
when a destination is for a LISP site versus a non-LISP site. And
the other is to source quench Map-Requests which 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 the 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 which is
invoking the reply. The source address of the Map-Reply is one of
the local locator addresses listed in the locator-set of any mapping
record in the message and SHOULD be chosen to allow 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 which 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 more difficult for an ETR to
misbehave. The second solution is to have ITRs and PTRs 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
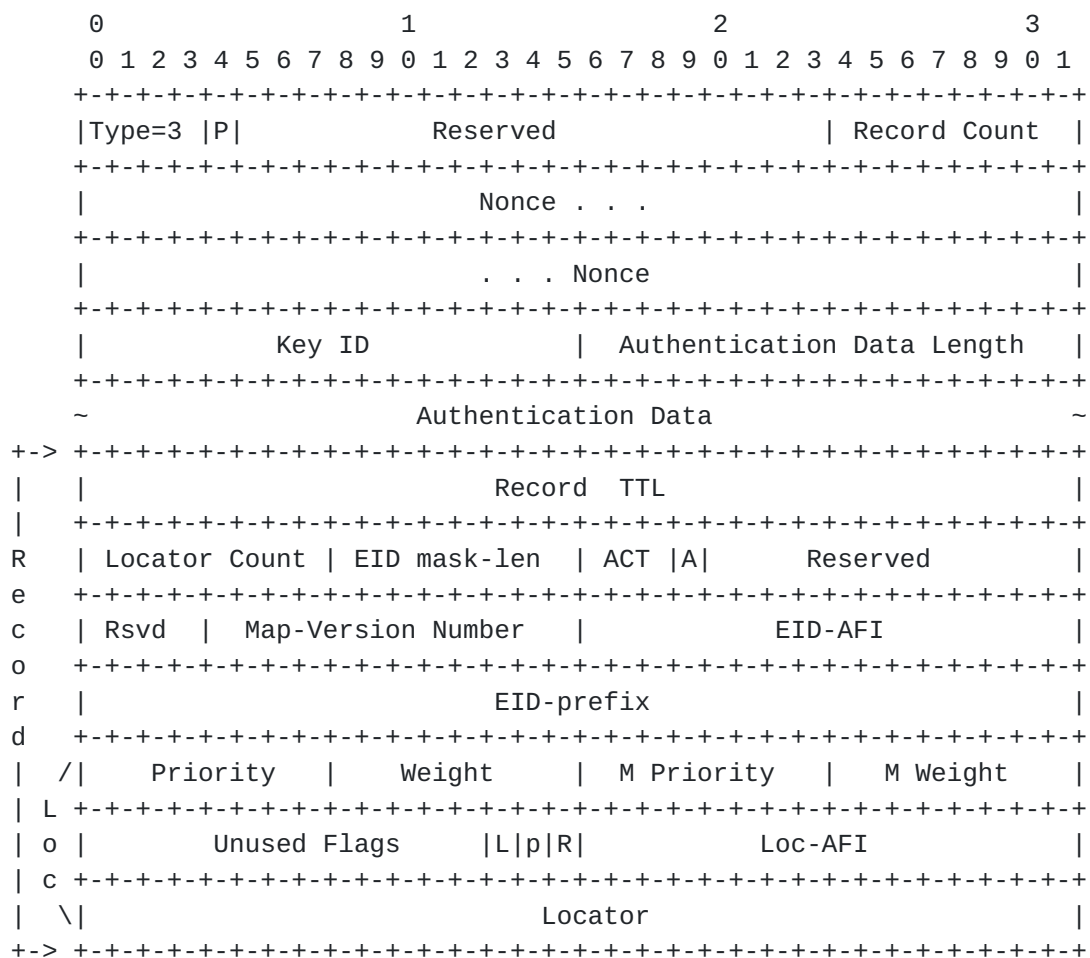independently or at the same time.

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 [LISP-MS]. 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:

```
      0                   1                   2                   3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |Type=3 |P|            Reserved              | Record Count  |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                         Nonce . . .                           |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                         . . . Nonce                           |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |            Key ID             |  Authentication Data Length   |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     ~                     Authentication Data                       ~
 +-> +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |   |                          Record  TTL                          |
 |   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 R   | Locator Count | EID mask-len  | ACT |A|       Reserved        |
 e   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 c   | Rsvd  |  Map-Version Number   |            EID-AFI            |
 o   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 r   |                          EID-prefix                           |
 d   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 | /|    Priority   |    Weight     |  M Priority   |   M Weight    |
 | L +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 | o |        Unused Flags       |L|p|R|           Loc-AFI            |
 | c +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 | \|                             Locator                           |
 +-> +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   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 for the Map-Server to proxy Map-Reply.
      The Map-Server will send non-authoritative Map-Replies on behalf
      of the ETR. Details on this usage will be provided in a future
      version of this draft.

   Reserved: Set to 0 on transmission and ignored on receipt.

   Record Count: 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-byte Nonce field is set to 0 in Map-Register messages.

   Key ID: A configured ID to find the configured Message
      Authentication Code (MAC) algorithm and key value used for the
      authentication function.

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

   Authentication Data: The message digest used from the output of the
      Message Authentication Code (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-128-256 [RFC4634]
      is recommended.

   The definition of the rest of the Map-Register can be found in the
   Map-Reply section. However, the record TTL field is not used and set
   to 0.

6.1.7. Encapsulated Control Message Format

   An Encapsulated Control Message is used to encapsulate control
   packets sent between xTRs and the mapping database system described
   in [LISP-MS].

```
        0                   1                   2                   3
        0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    / |                        IPv4 or IPv6 Header                    |
   OH |                      (uses RLOC addresses)                    |
    \ |                                                               |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    / |         Source Port = xxxx      |       Dest Port = 4342      |
   UDP +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    \ |           UDP Length            |        UDP Checksum         |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   LH  |Type=8 |                     Reserved                         |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    / |                        IPv4 or IPv6 Header                    |
   IH  |                  (uses RLOC or EID addresses)                |
    \ |                                                               |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    / |         Source Port = xxxx      |       Dest Port = yyyy      |
   UDP +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    \ |           UDP Length            |        UDP Checksum         |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   LCM |                    LISP Control Message                      |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
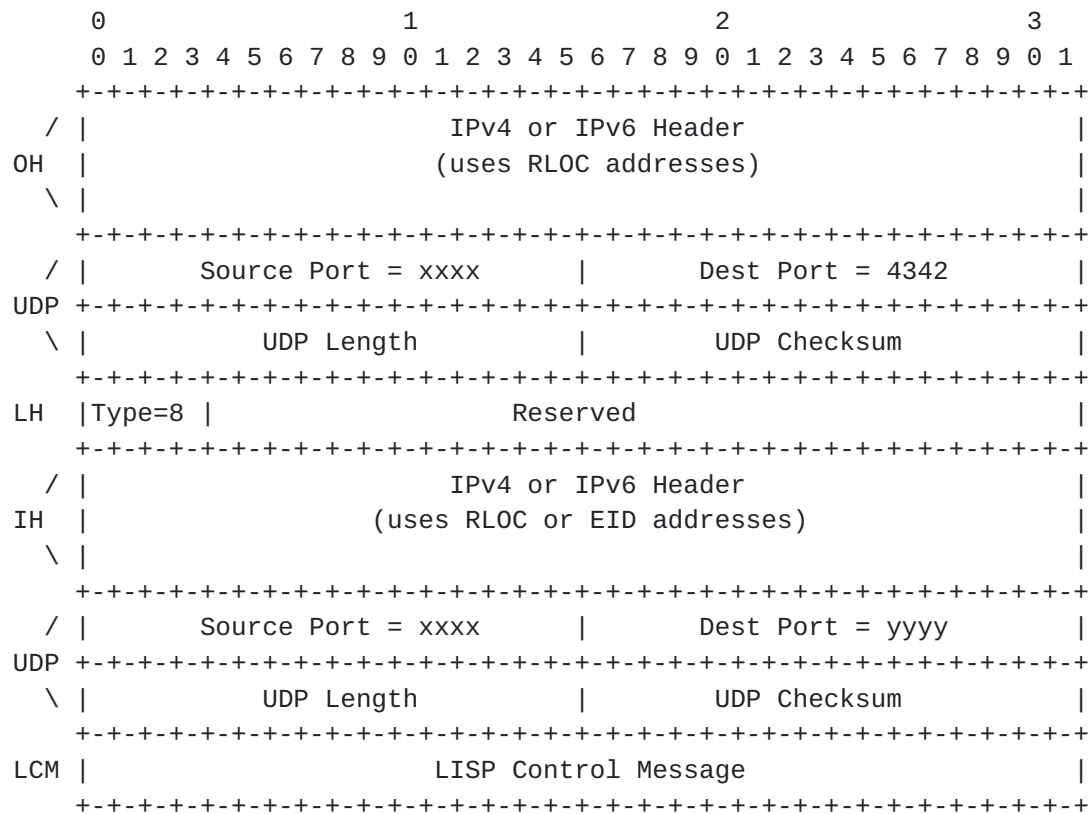
Packet header descriptions:

   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.

   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 depends on the
      control packet being encapsulated. When the control packet is a
      Map-Request or Map-Register, the source port is randomly assigned
      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 and PIM
   Join-Prune messages [MLISP] are allowed to be encapsulated.
   Encapsulating other types of LISP control messages are 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 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 enumerates different scenarios for choosing RLOCs and
the controls that are available:

o Server-side returns one RLOC. Client-side can only use one RLOC.
   Server-side has complete control of the selection.

o Server-side returns a list of RLOC where a subset of the list has
   the same best priority. Client can only use the subset list
   according to the weighting assigned by the server-side. In this
   case, the server-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.

o Server-side sets 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 are unreachable.

o Either side (more likely on 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 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.

o 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 which 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. Neither
the information contained in a Map-Reply or that stored in the
mapping database system provide reachability information for RLOCs.
Such reachability needs to be determined separately, 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 Loc-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 or ICMP Host Unreachable
    message for an RLOC it is using. This indicates that the RLOC is
    likely down.

3. An ITR which participates in the global routing system can
    determine that an RLOC is down if no BGP 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 [INTERWORK] and LISP-encapsulated data is sent to a
    non-LISP-capable site.

5. An ITR may receive a Map-Reply from a 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 Loc-
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:

o Under normal circumstances, each ITR will advertise a default
   route into the site IGP.

o 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
Loc-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 Loc-Status-Bits field for the packets they
encapsulate.

When an ETR decapsulates a packet, it will check for any change in
the Loc-Status-Bits field. When a bit goes from 1 to 0, the ETR will
refrain from encapsulating packets to an RLOC that is indicated as
down. It will only resume using that RLOC if the corresponding Loc-
Status-Bit returns to a value of 1. Loc-Status-Bits are associated

with a locator-set per EID-prefix. Therefore, when a locator becomes
unreachable, the Loc-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 or Host Unreachable messages as a
method to determine unreachability, they will refrain from using
Locators which are described in Locator lists of Map-Replies.
However, using this approach is unreliable because many network
operators turn off generation of ICMP Unreachable messages.

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

Also, BGP-enabled ITRs can unilaterally examine the BGP 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
Loc-Status-Bits indicate 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 an
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 mechanisms to
determine reachability.

6.3.1. Echo Nonce Algorithm

When data flows bidirectionally between locators from different
sites, a simple 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 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 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 echo-nonce-request state. The time the ITR waits to process the
echoed nonce before it determines the path is unreachable is variable
and 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 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 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 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 echo-nonce-request state, it can echo the ETR's
nonce in the next set of packets that it encapsulates and then
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
which 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
regard the locator unreachable erroneously. An ITR SHOULD only set
the E-bit in a 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 next section for an example of control-plane probing.

6.3.2. RLOC Probing Algorithm

RLOC Probing is a method that an ITR or PTR 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 are used for RLOC Probing.

RLOC probing is done in the control-plane on a timer basis where an
ITR or PTR 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 on the
ALT like 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 PTR. The ITR or PTR may include a mapping data
record for its own database mapping information.

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 from the destination address of the Map-Request
and the destination address of the Map-Reply is set from the source
address of the Map-Request. The Map-Reply SHOULD contain mapping
data for the EID-prefix contained in the Map-Request. This provides
the opportunity for the ITR or PTR, which 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 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 are very small.

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

6.4. 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 can be used or the
    traditional 5-tuple hash which includes the source and
    destination addresses, source and destination TCP, UDP, or SCTP
    port numbers and the IP protocol number field or IPv6 next-
    protocol fields of a packet 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 be 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 random value
allows core routers which 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 which 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.

6.5. 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 who
   requested its mappings. For scalability reasons, we want to maintain
   this approach but need to provide a way for ETRs change their
   mappings and inform the sites that are currently communicating with
   the ETR site using such mappings.

   When a locator record is added to the end of a locator-set, it is
   easy to update mappings. We assume new mappings will maintain the
   same locator ordering as the old mapping but 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 loc-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 removed from a locator-set, ITRs that have
   the mapping cached will not use the removed locator because the xTRs
   will set the loc-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 loc-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 loc-status-bit settings can
   be efficiently packed.

   We propose here three approaches for locator-set compaction, one
   operational 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.

6.5.1. 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.

6.5.2. Solicit-Map-Request (SMR)

Soliciting a Map-Request is a selective way for xTRs, 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 xTRs don't keep track of remote ITRs that have cached their
mappings, they can not tell exactly who needs the new mapping
entries. So an xTR will solicit Map-Requests from sites it is
currently sending encapsulated data to, and only from those sites.
The xTRs can locally decide the algorithm for how often and to how
many sites it sends SMR messages.

An SMR message is simply a bit set in a Map-Request message. An ITR
or PTR will send a Map-Request when they receive an SMR message.
Both the SMR sender and the Map-Request responder MUST rate-limited
these messages.

The following procedure shows how a 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 xTR which receives the SMR message will schedule sending
    a Map-Request message to the source locator address of the SMR
    message. A newly allocated random nonce is selected and the EID-

      prefix used is the one copied from the SMR message.

   3. The remote xTR MUST rate-limit the Map-Request until it gets a
       Map-Reply while continuing to use the cached mapping.

   4. The ETRs at the site with the changed mapping will reply to the
       Map-Request with a Map-Reply message provided the Map-Request
       nonce matches the nonce from the SMR. The Map-Reply messages
       SHOULD be rate limited. 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 mapping cache entry for the remote site so
       the loc-status-bits are reflective of the new mapping for packets
       going to the remote site. The ETR then stops sending SMR
       messages.

   For security reasons an ITR MUST NOT process unsolicited Map-Replies.
   The nonce MUST be carried from SMR packet, into the resultant Map-
   Request, and then into Map-Reply to reduce spoofing attacks.

   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 more secure to reach an authoritative ETR,
   it will deliver the Map-Request to the authoritative source of the
   mapping data.

6.5.3. 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 can stop to a removed locator and start 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 the Destination Map-Version Number is less than the current
   version for its mapping, the SMR procedure described in Section 6.5.2
   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 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 xTR 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 can assure that all ETRs
for a site registering to it will be Map-Version number synchronized.

See [VERSIONING] 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. By
   doing tunnel header prepending [RFC1955] and stripping instead of re-
   writing addresses, existing hardware can support the forwarding model
   with little or no modification. Where modifications are required,
   they should be limited to re-programming existing hardware rather
   than requiring expensive design changes to hard-coded algorithms in
   silicon.

   A few implementation techniques can be used to incrementally
   implement LISP:

   o 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 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 value is not necessary. No changes to existing,
      deployed hardware should be needed to support this.

   o On an ITR, prepending a new IP header is as simple as adding more
      bytes to a MAC rewrite string and prepending the string as part of
      the outgoing encapsulation procedure. Many routers that support
      GRE tunneling [RFC2784] or 6to4 tunneling [RFC3056] can already
      support this action.

   o When a received packet's outer destination address contains an EID
      which is not intended to be forwarded on the routable topology
      (i.e. LISP 1.5), the source address of a data packet or the
      router interface with which the source is associated (the
      interface from which it was received) can be associated with a VRF
      (Virtual Routing/Forwarding), in which a different (i.e. non-
      congruent) topology can be used to find EID-to-RLOC mappings.

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.
   There are two basic deployment trade-offs 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:

   o Are the tunnel routers spread out so that the caches are spread
      across all the memories of each router?

   o Should management "touch points" be minimized by choosing few
      tunnel routers, just enough for redundancy?

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

   When deciding on flat, recursive, or re-encapsulation tunneling, the
   following issues should be considered:

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

   o 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 at the
      benefit of steering traffic to resource available parts of the
      network.

   o The technique of re-encapsulation ensures that packets only
      require one tunnel header. So if a packet needs to be rerouted,
      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 describe 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 then 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 state.

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 as well.

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 offers the opposite benefit of the first-hop/last-hop tunnel
router scenario: the number of mapping entries and network management
touch points are reduced, allowing better scaling.

One disadvantage is that less of the network's 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.

8.3. ISP Provider-Edge (PE) Tunnel Routers

   Use of ISP PE routers as tunnel endpoint routers gives an ISP control
   over the location of the egress tunnel endpoints. That is, the ISP
   can decide if 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 or more 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 the RLOCs used.

   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 [INTERWORK].

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 ITR to
   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 matter as they are today. The ITR performs a TTL
   decrement and test for 0 before encapsulating. So the ITR hop is
   seen by the traceroute source has an EID address (the address of
   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 destination of the ICMP messages are to the ITR RLOC
   address, 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 as well as retaining 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 like 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 ICMP Time Exceeded
message payload. Therefore, only the ITR needs to pay special
attention for 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
bytes 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, you

cannot get all 3 segments of the traceroute. Segment 2 of the
traceroute can not 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-RLOC mappings for sites are expected to be
   handled by configuration, outside of the LISP protocol.

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
   [RFC3344bis] and Mobile IPv6 [RFC3775] [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 which 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 which 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 foreign agent or not. As a mobile
   endpoint changes networks, up to three LISP mapping changes may be
   required:

o 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 a EID-RLOC
   mapping for the home agent.

o The home agent might not have the EID-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.

o 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-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-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:

o 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.

o 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-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 a 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 which decreases the number of new EID-
RLOC mappings needed when a node moves, or maintains the validity of
an EID-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 the LISP Mobility Architecture specification [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,
   determines 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
   needs 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 Routing Locator translation
   is done at control-time.

   Another approach is to have the ITR not encapsulate a multicast
   packet and allow the the host built packet 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 is described in specification
   [MLISP].

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 Return- Routability 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 goes a long way
toward providing decent authentication.

LISP does not rely on a PKI infrastructure or a more heavy weight
authentication system. These systems challenge the scalability of
LISP which was a primary design goal.

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.

To deal with map-cache exhaustion attempts in an ITR/PTR, 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 PTRs.

13. IANA Considerations

This specification has already allocated UDP port numbers 4341 and 4342 assigned from the IANA registry.

14. **Prototype Plans and Status**

   The operator community has requested that the IETF take a practical
   approach to solving the scaling problems associated with global
   routing state growth. This document offers a simple solution which
   is intended for use in a pilot program to gain experience in working
   on this problem.

   The authors hope that publishing this specification will allow the
   rapid implementation of multiple vendor prototypes and deployment on
   a small scale. Doing this will help the community:

   o Decide whether a new EID-to-RLOC mapping database infrastructure
      is needed or if a simple, UDP-based, data-triggered approach is
      flexible and robust enough.

   o Experiment with provider-independent assignment of EIDs while at
      the same time decreasing the size of DFZ routing tables through
      the use of topologically-aligned, provider-based RLOCs.

   o Determine whether multiple levels of tunneling can be used by ISPs
      to achieve their Traffic Engineering goals while simultaneously
      removing the more specific routes currently injected into the
      global routing system for this purpose.

   o Experiment with mobility to determine if both acceptable
      convergence and session continuity properties can be scalably
      implemented to support both individual device roaming and site
      service provider changes.

   Here is a rough set of milestones:

   1. Interoperable implementations have been available since the
       beginning of 2009.

   2. Continue pilot deployment using LISP-ALT as the database mapping
       mechanism.

   3. Continue prototyping and studying other database lookup schemes,
       be it DNS, DHTs, CONS, ALT, NERD, or other mechanisms.

   4. Implement the LISP Multicast draft [MLISP].

   5. Implement the LISP Mobile Node draft [LISP-MN].

   6. Research more on how policy affects what gets returned in a Map-
       Reply from an ETR.

7. Continue to experiment with mixed locator-sets to understand how
    LISP can help the IPv4 to IPv6 transition.

8. Add more robustness to locator reachability between LISP sites.

9. Continue the deployment of Proxy-ETRs (PETRs) for uses like uRPF
    avoidance, IPv6 connectivity, and LISP-MN.

As of this writing the following accomplishments have been achieved:

1.   A unit- and system-tested software switching implementation has
     been completed on cisco NX-OS for this draft for both IPv4 and
     IPv6 EIDs using a mixed locator-set of IPv4 and IPv6 locators.

2.   A unit- and system-tested software switching implementation on
     cisco NX-OS has been completed for draft [ALT].

3.   A unit- and system-tested software switching implementation on
     cisco NX-OS has been completed for draft [INTERWORK]. Support
     for IPv4 translation is provided and PTR support for IPv4 and
     IPv6 is provided.

4.   The cisco NX-OS implementation supports an experimental
     mechanism for slow mobility.

5.   There are 5 LISP implementations that exist and the first 4
     below have gone through interoperability testing at IETF
     Hiroshima, based on the draft-ietf-lisp-05.txt spec:

     1. cisco NX-OS

     2. OpenLISP

     3. LISP-Click

     4. ZLisp

     5. cisco IOS

6.   Dave Meyer, Vince Fuller, Darrel Lewis, Gregg Schudel, Andrew
     Partan and the rest of the lisp-beta team continue to test all
     the features described above on a dual-stack infrastructure.

7.   Darrel Lewis and Dave Meyer have deployed both LISP translation
     and LISP PTR support in the pilot network. Point your browser
     to http://www.lisp4.net to see translation happening in action
     so your non-LISP site can access a web server in a LISP site.

8.   Soon http://www.lisp6.net will work where your IPv6 LISP site
      can talk to a IPv6 web server in a LISP site by using mixed
      address-family based locators.

9.   An public domain implementation of LISP is available. See
      [OPENLISP] for details.

10.   We have deployed Map-Resolvers and Map-Servers on the LISP pilot
      network to gather experience with [LISP-MS]. The first layer of
      the architecture are the xTRs which use Map-Servers for EID-
      prefix registration and Map-Resolvers for EID-to-RLOC mapping
      resolution. The second layer are the Map-Resolvers and Map-
      Servers which connect to the ALT BGP peering infrastructure.
      And the third layer are ALT-routers which aggregate EID-prefixes
      and forward Map-Requests.

11.   A cisco IOS implementation is available which currently supports
      IPv4 and IPv6 encapsulation and decapsulation features.

12.   A LISP router based LIG implementation is supported, deployed,
      and used daily to debug and test the LISP pilot network. See
      [LIG] for details.

13.   A Linux implementation of LIG has been made available and
      supported by Dave Meyer. It can be run on any Linux system
      which resides in either a LISP site or non-LISP site. See [LIG]
      for details. Public domain code can be downloaded from
      http://github.com/davidmeyer/lig/tree/master.

14.   An experimental implementation has been written for three
      locator reachability algorithms. Two are the Echo-Noncing and
      RLOC-Probing algorithms which are documented in this
      specification. The third is called TCP-counts which will be
      documented in future drafts.

15.   The LISP pilot network has been converted from using MD5 HMAC
      authentication for Map-Register messages to SHA-1 HMAC
      authentication. ETRs send with SHA-1 but Map-Servers can
      received from either for compatibility purposes.

16.   The LISP pilot network is in its 3rd generation. Current
      experiments are being performed to test EID-prefix aggregation
      at multiple service boundaries as well as deploying models for
      the existence of multiple Mapping Service Providers (MSPs).

      If interested in writing a LISP implementation, testing any of the
      LISP implementations, or want to be part of the LISP pilot program,
      please contact lisp@ietf.org.

15. References

15.1. Normative References

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

   [RFC1191] Mogul, J. and S. Deering, "Path MTU discovery", RFC 1191,
              November 1990.

   [RFC1498] Saltzer, J., "On the Naming and Binding of Network
              Destinations", RFC 1498, August 1993.

   [RFC1700] Reynolds, J. and J. Postel, "Assigned Numbers", RFC 1700,
              October 1994.

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

   [RFC1955] Hinden, R., "New Scheme for Internet Routing and
              Addressing (ENCAPS) for IPNG", RFC 1955, June 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.

   [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.

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

   [RFC3775] Johnson, D., Perkins, C., and J. Arkko, "Mobility Support
              in IPv6", RFC 3775, June 2004.

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

   [RFC4423] Moskowitz, R. and P. Nikander, "Host Identity Protocol
              (HIP) Architecture", RFC 4423, May 2006.

   [RFC4634] Eastlake, D. and T. Hansen, "US Secure Hash Algorithms
              (SHA and HMAC-SHA)", RFC 4634, July 2006.

   [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.

   [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.

   [RFC5533] Nordmark, E. and M. Bagnulo, "Shim6: Level 3 Multihoming
              Shim Protocol for IPv6", RFC 5533, June 2009.

   [UDP-TUNNELS]
              Eubanks, M. and P. Chimento, "UDP Checksums for Tunneled
              Packets"", draft-eubanks-chimento-6man-00.txt (work in
              progress), February 2009.

15.2. Informative References

   [AFI]   IANA, "Address Family Indicators (AFIs)", ADDRESS FAMILY
              NUMBERS http://www.iana.org/numbers.html, Febuary 2007.

   [ALT]   Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "LISP
              Alternative Topology (LISP-ALT)",
              draft-ietf-lisp-alt-04.txt (work in progress), April 2010.

   [APT]   Jen, D., Meisel, M., Massey, D., Wang, L., Zhang, B., and
              L. Zhang, "APT: A Practical Transit Mapping Service",
              draft-jen-apt-01.txt (work in progress), November 2007.

   [CHIAPPA] Chiappa, J., "Endpoints and Endpoint names: A Proposed
              Enhancement to the Internet Architecture", Internet-
              Draft http://www.chiappa.net/~jnc/tech/endpoints.txt,
              1999.

   [CONS]   Farinacci, D., Fuller, V., and D. Meyer, "LISP-CONS: A
              Content distribution Overlay Network Service for LISP",
              draft-meyer-lisp-cons-03.txt (work in progress),
              November 2007.

[DHTs]   Ratnasamy, S., Shenker, S., and I. Stoica, "Routing
            Algorithms for DHTs: Some Open Questions", PDF
            file http://www.cs.rice.edu/Conferences/IPTPS02/174.pdf.

[EMACS]   Brim, S., Farinacci, D., Meyer, D., and J. Curran, "EID
            Mappings Multicast Across Cooperating Systems for LISP",
            draft-curran-lisp-emacs-00.txt (work in progress),
            November 2007.

[GSE]   "GSE - An Alternate Addressing Architecture for IPv6",
            draft-ietf-ipngwg-gseaddr-00.txt (work in progress), 1997.

[INTERWORK]
            Lewis, D., Meyer, D., Farinacci, D., and V. Fuller,
            "Interworking LISP with IPv4 and IPv6",
            draft-ietf-lisp-interworking-01.txt (work in progress),
            March 2010.

[LCAF]   Farinacci, D., Meyer, D., and J. Snijders, "LISP Canonical
            Address Format", draft-farinacci-lisp-lcaf-01.txt (work in
            progress), April 2010.

[LIG]   Farinacci, D. and D. Meyer, "LISP Internet Groper (LIG)",
            draft-ietf-lisp-lig-00.txt (work in progress), April 2010.

[LISA96]   Lear, E., Katinsky, J., Coffin, J., and D. Tharp,
            "Renumbering: Threat or Menace?", Usenix , September 1996.

[LISP-MAIN]
            Farinacci, D., Fuller, V., Meyer, D., and D. Lewis,
            "Locator/ID Separation Protocol (LISP)",
            draft-farinacci-lisp-12.txt (work in progress),
            March 2009.

[LISP-MN]   Farinacci, D., Fuller, V., Lewis, D., and D. Meyer, "LISP
            Mobility Architecture", draft-meyer-lisp-mn-00.txt (work
            in progress), July 2009.

[LISP-MS]   Farinacci, D. and V. Fuller, "LISP Map Server",
            draft-ietf-lisp-ms-05.txt (work in progress), April 2010.

[LISP1]   Farinacci, D., Oran, D., Fuller, V., and J. Schiller,
            "Locator/ID Separation Protocol (LISP1) [Routable ID
            Version]",
            Slide-set http://www.dinof.net/~dino/ietf/lisp1.ppt,
            October 2006.

[LISP2]   Farinacci, D., Oran, D., Fuller, V., and J. Schiller,

"Locator/ID Separation Protocol (LISP2) [DNS-based Version]",
Slide-set http://www.dinof.net/~dino/ietf/lisp2.ppt,
November 2006.

[LISPDHT] Mathy, L., Iannone, L., and O. Bonaventure, "LISP-DHT:
Towards a DHT to map identifiers onto locators",
draft-mathy-lisp-dht-00.txt (work in progress),
February 2008.

[LOC-ID-ARCH]
Meyer, D. and D. Lewis, "Architectural Implications of
Locator/ID Separation",
draft-meyer-loc-id-implications-01.txt (work in progress),
Januaryr 2009.

[MLISP]   Farinacci, D., Meyer, D., Zwiebel, J., and S. Venaas,
"LISP for Multicast Environments",
draft-ietf-lisp-multicast-03.txt (work in progress),
April 2010.

[NERD]   Lear, E., "NERD: A Not-so-novel EID to RLOC Database",
draft-lear-lisp-nerd-08.txt (work in progress),
March 2010.

[OPENLISP]
Iannone, L. and O. Bonaventure, "OpenLISP Implementation
Report", draft-iannone-openlisp-implementation-01.txt
(work in progress), July 2008.

[RADIR]   Narten, T., "Routing and Addressing Problem Statement",
draft-narten-radir-problem-statement-00.txt (work in
progress), July 2007.

[RFC3344bis]
Perkins, C., "IP Mobility Support for IPv4, revised",
draft-ietf-mip4-rfc3344bis-05 (work in progress),
July 2007.

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

[RPMD]   Handley, M., Huici, F., and A. Greenhalgh, "RPMD: Protocol
for Routing Protocol Meta-data Dissemination",
draft-handley-p2ppush-unpublished-2007726.txt (work in
progress), July 2007.

[VERSIONING]
          Iannone, L., Saucez, D., and O. Bonaventure, "LISP Mapping
          Versioning", draft-iannone-lisp-mapping-versioning-01.txt
          (work in progress), March 2010.

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 review 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 discussion 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, 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, and Dimitri
   Papadimitriou.

   In particular, we would like to thank Dave Meyer for his clever
   suggestion for the name "LISP". ;-)

   This work originated in the Routing Research Group (RRG) of the IRTF.
   The individual submission [LISP-MAIN] was converted into this IETF
   LISP working group draft.

Appendix B. Document Change Log

B.1. Changes to draft-ietf-lisp-07.txt

   o Posted April 2010.

   o Added I-bit to data header so LSB field can also be used as an
      Instance ID field. When this occurs, the LSB field is reduced to
      8-bits (from 32-bits).

   o Added V-bit to the data header so the 24-bit nonce field can also
      be used for source and destination version numbers.

   o Added Map-Version 12-bit value to the EID-record to be used in all
      of Map-Request, Map-Reply, and Map-Register messages.

   o Added multiple ITR-RLOC fields to the Map-Request packet so an ETR
      can decide what address to select for the destination of a Map-
      Reply.

   o Added L-bit (Local RLOC bit) and p-bit (Probe-Reply RLOC bit) to
      the Locator-Set record of an EID-record for a Map-Reply message.
      The L-bit indicates which RLOCs in the locator-set are local to
      the sender of the message. The P-bit indicates which RLOC is the
      source of a RLOC-probe Reply (Map-Reply) message.

   o Add reference to the LISP Canonical Address Format [LCAF] draft.

   o Made editorial and clarification changes based on comments from
      Dhirendra Trivedi.

   o Added wordsmithing comments from Joel Halpern on DF=1 setting.

   o Add John Zwiebel clarification to Echo Nonce Algorithm section
      6.3.1.

   o Add John Zwiebel comment about expanding on proxy-map-reply bit
      for Map-Register messages.

   o Add NAT section per Ron Bonica comments.

   o Fix IDnits issues per Ron Bonica.

   o Added section on Virtualization and Segmentation to explain the
      use if the Instance ID field in the data header.

   o There are too many P-bits, keep their scope to the packet format
      description and refer to them by name every where else in the

spec.

o Scanned all occurrences of "should", "should not", "must" and
   "must not" and uppercased them.

o John Zwiebel offered text for section 4.1 to modernize the
   example. Thanks Z!

o Make it more clear in the definition of "EID-to-RLOC Database"
   that all ETRs need to have the same database mapping. This
   reflects a comment from John Scudder.

o Add a definition "Route-returnability" to the Definition of Terms
   section.

o In section 9.2, add text to describe what the signature of
   traceroute packets can look like.

o Removed references to Data Probe for introductory example. Data-
   probes are still part of the LISP design but not encouraged.

o Added the definition for "LISP site" to the Definition of Terms"
   section.

B.2. Changes to draft-ietf-lisp-06.txt

Editorial based changes:

o Posted December 2009.

o Fix typo for flags in LISP data header. Changed from "4" to "5".

o Add text to indicate that Map-Register messages must contain a
   computed UDP checksum.

o Add definitions for PITR and PETR.

o Indicate an AFI value of 0 is an unspecified address.

o Indicate that the TTL field of a Map-Register is not used and set
   to 0 by the sender. This change makes this spec consistent with
   [LISP-MS].

o Change "... yield a packet size of L bytes" to "... yield a packet
   size greater than L bytes".

o Clarify section 6.1.5 on what addresses and ports are used in Map-
   Reply messages.

o Clarify that LSBs that go beyond the number of locators do not to
   be SMRed when the locator addresses are greater lexicographically
   than the locator in the existing locator-set.

o Add Gregg, Srini, and Amit to acknowledgment section.

o Clarify in the definition of a LISP header what is following the
   UDP header.

o Clarify "verifying Map-Request" text in section 6.1.3.

o Add Xu Xiaohu to the acknowledgment section for introducing the
   problem of overlapping EID-prefixes among multiple sites in an RRG
   email message.

Design based changes:

o Use stronger language to have the outer IPv4 header set DF=1 so we
   can avoid fragment reassembly in an ETR or PETR. This will also
   make IPv4 and IPv6 encapsulation have consistent behavior.

o Map-Requests should not be sent in ECM with the Probe bit is set.
   These type of Map-Requests are used as RLOC-probes and are sent
   directly to locator addresses in the underlying network.

o Add text in section 6.1.5 about returning all EID-prefixes in a
   Map-Reply sent by an ETR when there are overlapping EID-prefixes
   configure.

o Add text in a new subsection of section 6.1.5 about dealing with
   Map-Replies with coarse EID-prefixes.

B.3. **Changes to draft-ietf-lisp-05.txt**

o Posted September 2009.

o Added this Document Change Log appendix.

o Added section indicating that encapsulated Map-Requests must use
   destination UDP port 4342.

o Don't use AH in Map-Registers. Put key-id, auth-length, and auth-
   data in Map-Register payload.

o Added Jari to acknowledgment section.

o State the source-EID is set to 0 when using Map-Requests to
   refresh or RLOC-probe.

o Make more clear what source-RLOC should be for a Map-Request.

o The LISP-CONS authors thought that the Type definitions for CONS
   should be removed from this specification.

o Removed nonce from Map-Register message, it wasn't used so no need
   for it.

o Clarify what to do for unspecified Action bits for negative Map-
   Replies. Since No Action is a drop, make value 0 Drop.

B.4. Changes to draft-ietf-lisp-04.txt

o Posted September 2009.

o How do deal with record count greater than 1 for a Map-Request.
   Damien and Joel comment. Joel suggests: 1) Specify that senders
   compliant with the current document will always set the count to
   1, and note that the count is included for future extensibility.
   2) Specify what a receiver compliant with the draft should do if
   it receives a request with a count greater than 1. Presumably, it
   should send some error back?

o Add Fred Templin in acknowledgment section.

o Add Margaret and Sam to the acknowledgment section for their great
   comments.

o Say more about LAGs in the UDP section per Sam Hartman's comment.

o Sam wants to use MAY instead of SHOULD for ignoring checksums on
   ETR. From the mailing list: "You'd need to word it as an ITR MAY
   send a zero checksum, an ETR MUST accept a 0 checksum and MAY
   ignore the checksum completely. And of course we'd need to
   confirm that can actually be implemented. In particular, hardware
   that verifies UDP checksums on receive needs to be checked to make
   sure it permits 0 checksums."

o Margaret wants a reference to
   http://www.ietf.org/id/draft-eubanks-chimento-6man-00.txt.

o Fix description in Map-Request section. Where we describe Map-
   Reply Record, change "R-bit" to "M-bit".

o Add the mobility bit to Map-Replies. So PTRs don't probe so often
   for MNs but often enough to get mapping updates.

   o Indicate SHA1 can be used as well for Map-Registers.

   o More Fred comments on MTU handling.

   o Isidor comment about spec'ing better periodic Map-Registers. Will
      be fixed in draft-ietf-lisp-ms-02.txt.

   o Margaret's comment on gleaning: "The current specification does
      not make it clear how long gleaned map entries should be retained
      in the cache, nor does it make it clear how/ when they will be
      validated. The LISP spec should, at the very least, include a
      (short) default lifetime for gleaned entries, require that they be
      validated within a short period of time, and state that a new
      gleaned entry should never overwrite an entry that was obtained
      from the mapping system. The security implications of storing
      "gleaned" entries should also be explored in detail."

   o Add section on RLOC-probing per working group feedback.

   o Change "loc-reach-bits" to "loc-status-bits" per comment from
      Noel.

   o Remove SMR-bit from data-plane. Dino prefers to have it in the
      control plane only.

   o Change LISP header to allow a "Research Bit" so the Nonce and LSB
      fields can be turned off and used for another future purpose. For
      Luigi et al versioning convergence.

   o Add a N-bit to the data header suggested by Noel. Then the nonce
      field could be used when N is not 1.

   o Clarify that when E-bit is 0, the nonce field can be an echoed
      nonce or a random nonce. Comment from Jesper.

   o Indicate when doing data-gleaning that a verifying Map-Request is
      sent to the source-EID of the gleaned data packet so we can avoid
      map-cache corruption by a 3rd party. Comment from Pedro.

   o Indicate that a verifying Map-Request, for accepting mapping data,
      should be sent over the the ALT (or to the EID).

   o Reference IPsec RFC 4302. Comment from Sam and Brian Weis.

   o Put E-bit in Map-Reply to tell ITRs that the ETR supports echo-
      noncing. Comment by Pedro and Dino.

o Jesper made a comment to loosen the language about requiring the
   copy of inner TTL to outer TTL since the text to get mixed-AF
   traceroute to work would violate the "MUST" clause. Changed from
   MUST to SHOULD in section 5.3.

B.5. Changes to draft-ietf-lisp-03.txt

o Posted July 2009.

o Removed loc-reach-bits longword from control packets per Damien
   comment.

o Clarifications in MTU text from Roque.

o Added text to indicate that the locator-set be sorted by locator
   address from Isidor.

o Clarification text from John Zwiebel in Echo-Nonce section.

B.6. Changes to draft-ietf-lisp-02.txt

o Posted July 2009.

o Encapsulation packet format change to add E-bit and make loc-
   reach-bits 32-bits in length.

o Added Echo-Nonce Algorithm section.

o Clarification how ECN bits are copied.

o Moved S-bit in Map-Request.

o Added P-bit in Map-Request and Map-Reply messages to anticipate
   RLOC-Probe Algorithm.

o Added to Mobility section to reference draft-meyer-lisp-mn-00.txt.

B.7. Changes to draft-ietf-lisp-01.txt

o Posted 2 days after draft-ietf-lisp-00.txt in May 2009.

o Defined LEID to be a "LISP EID".

o Indicate encapsulation use IPv4 DF=0.

o Added negative Map-Reply messages with drop, native-forward, and
   send-map-request actions.

o Added Proxy-Map-Reply bit to Map-Register.

B.8. Changes to draft-ietf-lisp-00.txt

o Posted May 2009.

o Rename of draft-farinacci-lisp-12.txt.

o Acknowledgment to RRG.

Authors' Addresses

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

   Email: dino@cisco.com

   Vince Fuller
   cisco Systems
   Tasman Drive
   San Jose, CA 95134
   USA

   Email: vaf@cisco.com

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

   Email: dmm@cisco.com

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

   Email: darlewis@cisco.com