

LISP Working Group
Internet-Draft
Intended status: Experimental
Expires: April 30, 2015

S. Barkai
ConteXtream Inc.
D. Farinacci
lispers.net
D. Meyer
Brocade
F. Maino
V. Ermagan
Cisco Systems
A. Rodriguez-Natal
A. Cabellos-Aparicio
Technical University of Catalonia
October 27, 2014

LISP Based FlowMapping for Scaling NFV
draft-barkai-lisp-nfv-05

Abstract

This draft describes an RFC 6830 Locator ID Separation Protocol (LISP) based distributed flow-mapping-fabric for dynamic scaling of virtualized network functions (NFV). Network functions such as subscriber-management, content-optimization, security and quality of service, are typically delivered using proprietary hardware appliances embedded into the network as turn-key service-nodes or service-blades within routers. Next generation network functions are being implemented as pure software instances running on standard servers - unbundled virtualized components of capacity and functionality. LISP-SDN based flow-mapping, dynamically assembles these components to whole solutions by steering the right traffic in the right sequence to the right virtual function instance.

Requirements Language

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]

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on April 30, 2015.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Terminology	3
3. Connectivity Model	5
4. Flow-Mapping Elements	7
5. Day-in-life of a Mapped Flow	8
5.1. XTR Flow Edge	9
5.2. Map Resolvers-Servers	11
5.3. XTRs-Mappers Scaling	11
6. Message Formats	11
7. QOS and Echo Measurements	14
8. Security Considerations	14
9. IANA Considerations	14
10. Acknowledgements	14
11. Normative References	14
Authors' Addresses	15

1. Introduction

This draft describes an RFC 6830 Locator ID Separation Protocol (LISP) based distributed flow-mapping-fabric for dynamic scaling of virtualized network functions (NFV). [RFC6830] Network functions such as subscriber-management, content-optimization, security and quality of service, are typically delivered using proprietary hardware

appliances embedded into the network as turn-key service-nodes or service-blades within routers.

This monolithic service delivery method increases the complexity of service roll-out and capacity planning, limits providers' choices, and slows down revenue generating service innovation. Next generation network functions are being implemented as pure software instances running on standard servers - unbundled ("googlized") virtualized components of capacity and functionality. Such a component based model opens up service provider networks to the savings of elasticity and open architecture driven innovation. However this model also presents the network with the new challenges of assembling components, developed by 3rd parties, into whole solutions, by forwarding the right traffic to the right function-block at the right sequence.

While this is possible, to some extent, by traditional virtual networking - virtual bridges(vBridges) and virtual-routing-forwarding (VRF) - these mechanisms are relatively static and require complex and intensive configuration of network interfaces, while elastic components are not network topology bound. Software-defined-networks, (SDN) flow based models are much more dynamically programmable but are also very centralized and hence have limited scale and resiliency. By enhancing SDN models with RFC6830 overlay model, as [I-D.rodriqueznatal-lisp-sdn] suggests, we offer a best fit to dynamic assembly of virtualized network functions in the service-providers data-centers and distribution-centers.

2. Terminology

The following terms are used to describe a LISP based implementation of Software-Defined Flow-Mapping-Fabric for NFV:

- o LISP-SDN - is an enhancement to the basic SDN model of (1) hop-to-hop (2) push-down flow-commands (3) by concentrated-controller.. to a LISP based architecture of (1) distributed-overlay e.g. SDN over IP (2) based on a pull-publish-subscribe actions from xTR-edges up.. (3) to a global mapping service. A mapping service scaled by and connected over the IP underlay network. LISP-SDN protocol operation details are covered in [I-D.rodriqueznatal-lisp-sdn].
- o Virtualized Network Function (VNF) - is a process instance with an EID and RLOC that performs a defined set of inline network functions. a VNF can be software on a virtual-machine (VM) performing a function like multimedia signaling, mobility management, content caching or streaming, security, filtering, optimization, etc. A VNF class type and VNF instance capacity,

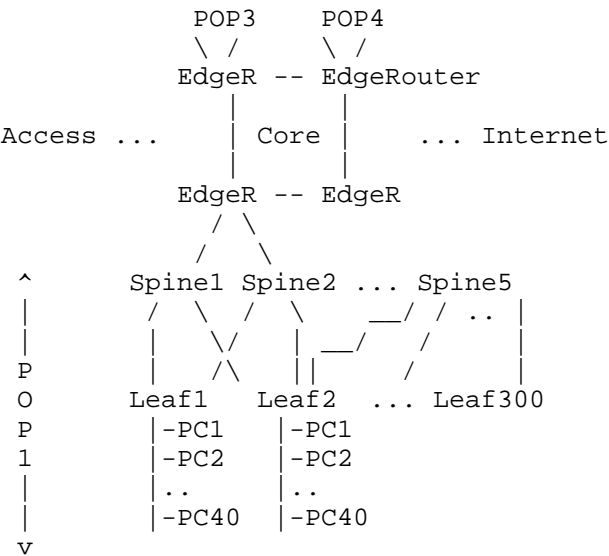
load, and location are attributes that can be resolved by the LISP-SDN mapping service.

- o Client-Flow - is a sequence of packets that corresponds to a specific communication thread or network conversation between a client application and a network service. Client-flows are typically processed by various in-network functions either as the end service side to the network conversation, or as middle-box functionality.
- o SDN-xTR - is a LISP xTR that behaves as defined in [I-D.rodriqueznatal-lisp-sdn]. It classifies traffic into application flows, maps, encapsulates, and decapsulates flows in order to emerge a flow-mapping solution - along with a collection of the SDN-xTR elements, and the LISP-SDN mapping service.
- o SDN-Overlay - is the network formed by the collection of inter-connected SDN-xTR
- o SDN-Underlay - is the IPvN network connecting SDN-xTRs
- o SDN-Outerlay (interim name)- is the collection of networks and interfaces aggregated by the various SDN-xTRs connecting VNFs and Client-flows coming from access networks or the Internet.
- o Flow-Rule - is a set of pattern tuples that match any part of a packet header and is used to classify packets into flows as well as trigger forwarding actions such as encapsulation / decapsulation, network address translation (NAT), etc. We differentiate between exact-match rules (many) which include an exact set of tuple bits, and best match rules (fewer) which contain both tuple bits and wild-cards "*".
- o Virtual IP (VIP) - is an IP address or EID that identifies a function rather than a specific destination. For example all the encapsulated client-flow traffic sent from a base-station eNodeBs over a transport network, can have as destination a VIP which represents in a given LISP-SDN solution, the function mobile-gateway or PGW, and not any specific destination.
- o Flow-Affinity - is the association between a client-flow and a VNF instance. VNF logic will typically create long-lived (minutes) in memory states in order to perform its functions. Therefore once an affinity is established it is best to keep it for as long as possible in order not to stress or break the VNF application.

3. Connectivity Model

The basic connectivity model used to assemble VNFs into whole solution is the flow-mapping-fabric. Unlike topological forwarding which is based on source-subnet >> routed hop by hop >> destination-subnet, a flow-mapping-fabric maps, forwards and "patches" flows by identity directly to the end systems. The identities used for the flow-mapping-fabric are those associated with the client-flows e.g. Subscriber ID, phone number, TCP port, etc. and those associated with the VNF e.g. the type, location, physical address, etc. the flow-mapping-fabric is implemented as a LISP-SDN overlay, over in-place IP underlay, assembling outerlay flows into solutions. Bellow are basic assumptions regarding the Underlay, Outerlay, and Overlay in the solution:

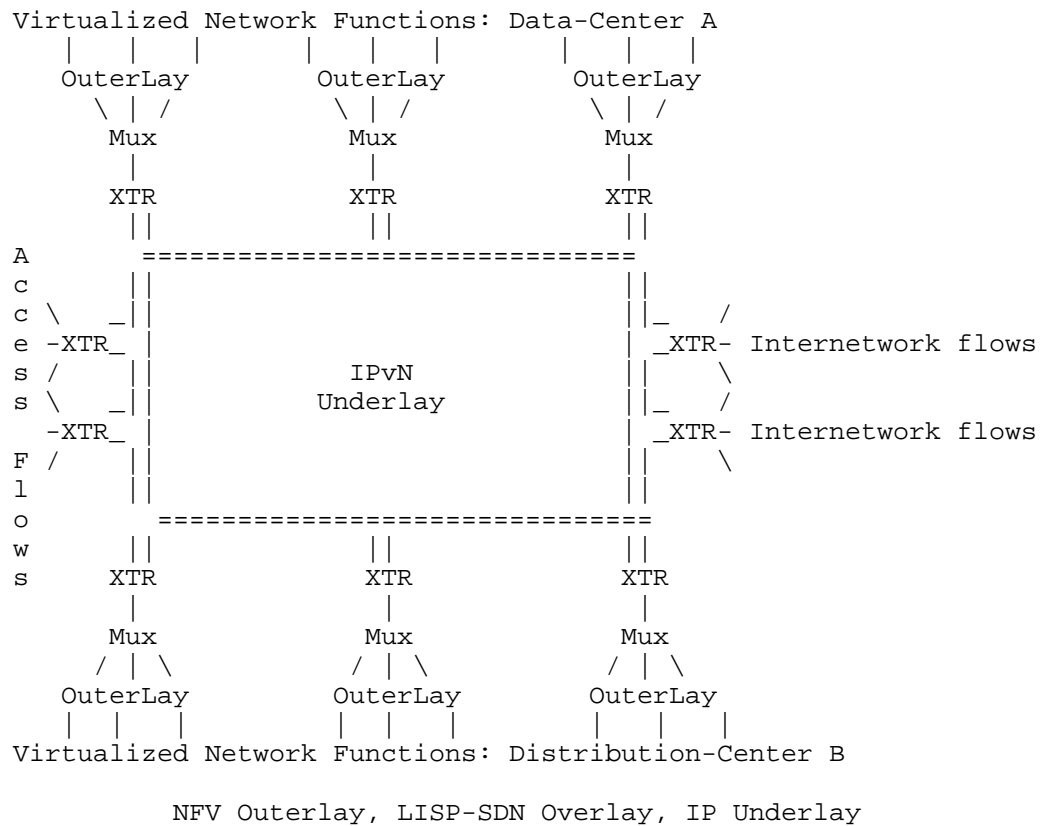
- o The underlying physical network is assumed to be topology based and implemented using standard bridging and routing. Conventional design principles are applied in order to achieve both capacity and availability of connectivity. Typical examples of underlays include spine-leaf switching for clustering server racks, and, core-edge routing inter-connecting server clusters across points of presence. Edge networks are also used to connect to access networks and Internet.
- o The flow-mapping-fabric maps outerlay client-flows to VNFs. This enables assembly, scaling, balanced high-utilization, massive concurrency, and hence, performance of NFVs. By mapping each client-flow to the correct functional instance the system engages as many VNF components as are available, scaled within and across data-centers. Applied recursively client-flow mapping can chain a sequence of VNF components to make up an end-to-end service.
- o The overlay network is based on location-identity-separation and forms a virtualization indirection ring around spines and cores. The overlay edges aggregate outerlay client-flows and VNFs. Outerlay flows are classified, mapped, and encapsulated over the edge through the underlay interfaces and are transported to the right identity's locations.



Core-Edge Spine-Leaf Underlays

v <<	FunctionA	FunctionB	..	FunctionN
v				
Recursion	Instancel..i	Instancel..j	Instancel..k	
v				
v				
SubsFlow1	o o o o - - - + o o o - - - o o o o			
SubsFlow2	o + o o - - - o o o o - - - o o o o			
.	
.	
.	
SubsFlowM	o o o o - - - o o o o - - - + o o o			

Flow-Mapping-Fabric

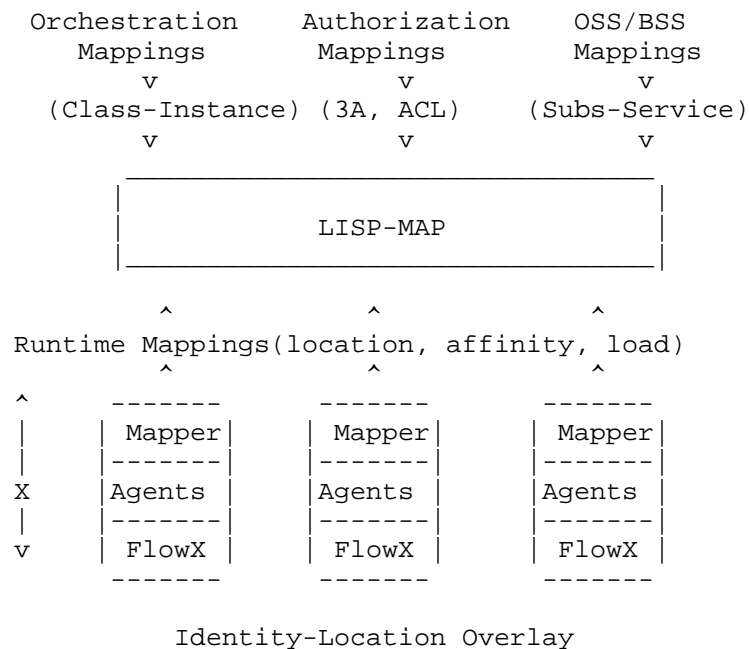


4. Flow-Mapping Elements

In order to implement NFV Flow-Mapping-Fabric using LISP-SDN We use the following components and capabilities:

1. Flow-Switching: is a component within an SDN-xTR and contains a set of n-tuple flow-rules matched against each packet in order to separate it to (LOCALLY defined) sequences representing flows. Flows are either Encapsulated into the Overlay, decapsulated to the Outerlay, or forwarded to SDN-xTR Control Agents.
2. Control-Agents: are software processes running in SDN-xTRs and are invoked for each flow where an exact match was not present in the Flow-Switching. The default "catch-all" Flow-Handler maps IP flows to locations and gateways based on RFC 6830. Protocol and application specific handlers can be loaded into the SDN-xTR for handling specific mapping and AFFINITY requirements of network functions. Examples of such protocols and applications can be SIP, GTP, S1X etc.

3. Global-Mapping: is how GLOBALLY significant key-value mappings is translated to LOCALLY defines flow masks and encapsulation actions. Examples of such mappings include: Map a functional instance ID to a function class ID; map subscriber-application ID to virtual function instance ID; map instance ID to location; instance to health, load, tenant; etc.



5. Day-in-life of a Mapped Flow

Let us walk through detailed steps of the use of RFC6830 and LISP architecture in order to perform resource virtualization and flow assignment to virtual function instances.

At a high level, when a client-flow packet first arrives at a SDN-xTR on the edge of the LISP overlay, the SDN-xTR must decide on a VNF instance that is best suited to service this flow, assign this flow to the selected VNF, and encapsulate this flow to the RLOC of the selected virtual function instance.

To select the best suited VNF instance, the SDN-xTR queries the Mapping System with the extracted identity parameters, both the client and the function EIDs, and receives the list of all VNF instances that represent that Function along with their RLOC and health-load attributes. The SDN-xTR runs local algorithms on the returned set to select the best suited virtual function instance.

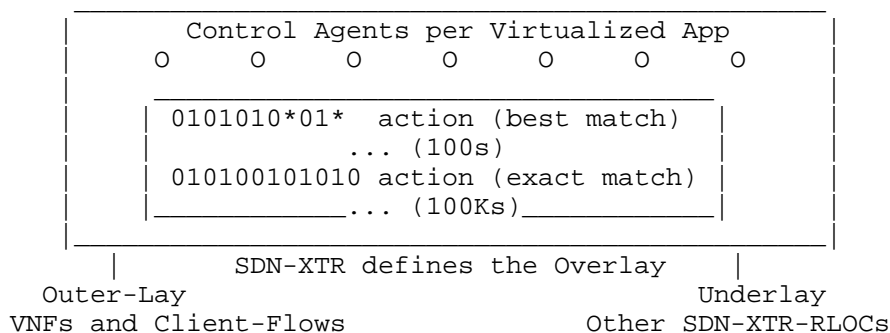
Once selected, the SDN-xTR stores (registers) the assignment of this flow to the associated VNF instance in the Mapping System. This assignment is referred to as the Affinity for this flow. The SDN-xTR also programs an exact match flow rule in its data-plane, so future packets from this flow will be mapped to the same EID-RLOC.

In the following subsections We describe this process in more detail.

5.1. XTR Flow Edge

SDN-xTR locations define the boundary of the virtual network. For the purpose of LISP-SDN flow-mapping-fabric We refer to the bellow SDN-XTR generic reference architecture. Actual vendor implementations may vary, but most likely will include similar components and structure. The SDN-XTR includes:

- o Mux-DeMux: Interfaces to the Underlay and Overlay
- o Flow-Rules: Patterns-Actions, Exact / Best Match, Encap-Decap
- o Control-Agents: Application specific flow-handlers registered in the Flow-Rules



SDN-XTR Reference Architecture

SDN-XTR Flow Switching works as follows:

1. For traffic from the Overlay of THIS xTR that has an exact match of all the source-dest-tags.. n-tuples, the packets are processed by rule actions including encapsulation to the RLOC of the xTR which aggregates the relevant function instance to which this flow is mapped to.

2. For traffic from the Underlay that has an exact match of all the source-dest-tags.. n-tuples, the packets are processed by rule actions including decapsulation and forwarding to the Outerlay of THIS xTR.
3. Traffic from the Outer-Lay or Underlay that does NOT have an exact match of all the source-dest-tags.. tuples required for normal forwarding, packets are forwarded to the control agent registered in the best-matching rule.

SDN-XTR Control Agents work as follows:

1. Mapping agent type and application scope is defined by the best match entries that point to it. Control agents will typically self-register in the flow-switch. XTR control-agents can register to an existing best-match rule, or instantiate a new one.
2. Typical rule-patterns are pattern-scoped by an agent registration, and can include: protocol or service type header indications; specific virtual IP addresses (VIP) that represent a service and not a specific destination; a specific source and wild-card destination; or vice versa.
3. Mapping agents work with the LISP-SDN mapping service in order to establish a global context and local considerations for mapping decision. The goal of the agents' decision is ultimately to provision the correct exact-match rule and actions that will offload the flow-packets to flow-switching described above.

The SDN-xTR control agents query the LISP-SDN Mapping System with the flow attributes including the destination VIP, as follows:

Mapping System Lookup: Map-Request (Client identity, Function-EID)

Two outcomes are possible based on whether an affinity already exists for this flow (flow has already been assigned to a virtual function instance):

- o Outcome A:
 - * If an affinity already exists in the Mapping System, the Mapping System returns the locator address (RLOC) associated with the Function-Instance-EID that the (Client-EID, Function-EID) is mapped to.
 - * Map-Reply: ((Client-EID, Function-EID) -> Function-Instance-RLOC)

- * In this case the Mapping System also subscribes the SDN-xTR to the Function-Instance-EID, and to the (Client-EID, Function-EID) flow in order to receive updates in case of changes on these entries. Examples of these changes are change of RLOC for the Function-Instance-EID (specially if this is a virtual application), or change of affinity for (Client-EID, Function-EID) to another Function-Instance-EID.
- * After receiving the Map-Reply from the Mapping System, the SDN-xTR programs an exact match for the flow in the xTR data-plane.
- o Outcome B:
 - * If there is no affinity previously stored, the Mapping System returns a list of Records, including one Record per each instance of the Function-EID, with their associated RLOCs and flags (weight, priority).
 - * Map-Reply: (client EID, Function-Instance-Record 1, Function-Instance-Record 2...)
 - * the SDN-xTR then selects the best suited Function-Instance-EID for this flow based on local algorithms, and registers the affinity in the Mapping System. The Mapping System stores the affinity and subscribes the SDN-xTR to the affinity and to the Function-Instance-EID in the affinity, so that SDN-xTR would receive updates if any of these changes.
 - * Map-Register ((Client-EID, Function-EID) -> Function-Instance-EID)
- o Note: An SDN-xTR must be able to query for the list of App-Instance-Records even if an affinity already exists. For this purpose a flag is required in the Map-Request to indicate whether xTR wants this info or not. We can overload the M bit in Map-Request, or allocate a new bit for this.
- o

5.2. Map Resolvers-Servers

5.3. XTRs-Mappers Scaling

6. Message Formats

This section specifies the packet formats used throughout the flow-mapping process explained above. This section is expected to be extended and moved to [I-D.rodriueznatal-lisp-sdn].

A Map-Request is used with a 2-Tuple Src/Dst LCAF to query the Mapping System for the affinity or list of virtual function instance records for this flow.

```

      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|p|s|   Reserved   |   IRC   | Record Count |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     Nonce . . .                                     |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     . . . Nonce                                     |
+-----+-----+-----+-----+-----+-----+-----+-----+
|           Source-EID-AFI           | Source EID Address ... |
+-----+-----+-----+-----+-----+-----+-----+-----+
|           ITR-RLOC-AFI 1           | ITR-RLOC Address 1 ... |
+-----+-----+-----+-----+-----+-----+-----+-----+
|   Reserved   | EID mask-len | EID-prefix-AFI = 16387 |
+-----+-----+-----+-----+-----+-----+-----+-----+
L |   Rsvd1   |   Flags   |   Type = 12   |   Rsvd2   |
C |   +-----+-----+-----+-----+-----+-----+-----+
A |   4 + n   |           |   Reserved   |
F |   +-----+-----+-----+-----+-----+-----+-----+
  | Source-ML |   Dest-ML   |           AFI = x           |
  |   +-----+-----+-----+-----+-----+-----+-----+
  |           Source-Prefix ... |
  |   +-----+-----+-----+-----+-----+-----+-----+
  |           AFI = x           |   Destination-Prefix ... |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

Where:

Source-Prefix = Client-EID

Destination-Prefix = App-EID

LISP Map-Request with 2-Tuple Src/Dst LCAF

In order to specify a 5 tuple flow, rather than just a two tuple source and destination, the combination of LCAF type 12 and LCAF type 4 must be used.

If an affinity exists in the Mapping System, meaning that the flow is already assigned to a virtual function instance, then the RLOC of that Function-Instance must be returned by the Mapping System. A Map-Reply with a 2-Tuple Src/Dst Lcaf can be used for this.

		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 S				Reserved																Record Count															
		+-----+																																			
																		Nonce . . .																			
		+-----+																																			
																		. . . Nonce																			
+----->		+-----+																																			
																		Record TTL																			
R		+-----+																																			
e		Locator Count								EID mask-len								ACT A				Reserved															
c		+-----+																																			
o		Rsvd				Map-Version Number												EID-prefix-AFI = 16387																			
r	+-->	+-----+																																			
d		Rsvd1								Flags								Type = 12								Rsvd2											
		+-----+																																			
																		4 + n																			
L		+-----+																																			
C		Source-ML								Dest-ML								AFI = x																			
A		+-----+																																			
F																		Source-Prefix ...																			
		+-----+																																			
																		AFI = x								Destination-Prefix ...											
+-->		+-----+																																			
/		Priority								Weight								M Priority								M Weight											
L		+-----+																																			
o		Unused Flags																L p R				Loc-AFI															
c		+-----+																																			
	\	Locator																																			
+----->		+-----+																																			

If no affinity exists, the Mapping System returns a list of records, including one record per each Function-Instance for the flow's Function-EID. A LISP Map-Reply can be used for this purpose with a 2-Tuple Src/Dst LCAF as the EID prefix in each Record.

If it is desired to return tuples of (Function-Instance-EID -> RLOC) per each record, a new LCAF, introduced as below, could be used.

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								
+++++																																							
AFI = 16387										Rsvd1										Flags																			
+++++																																							
Type = 14										Rsvd2										4 + n																			
+++++																																							
EID-ML										RSVD3										EID-AFI = x																			
+++++																																							
EID-Prefix ...																																							
+++++																																							
RLOC-AFI = x															Locator Address ...																								
+++++																																							

EID-RLOC LCAF:

In which, for the purpose of NFV, EID prefix will be used to specify Function-Instance-EID, and Locator address is the RLOC associated with that Function-Instance-EID. This LCAF can be used in place of the Loc-AFI in the Map-Reply Message above to include a list of (Function-Instance-EID,RLOC) for every (Client-EID, Function-EID) in the Map-Reply.

Finally to store the affinity of the flow in the Mapping System a Map-Register can be used where EID AFI is filled with a LCAF type 12 (2-Tuple Src/Dst LCAF), and Loc-AFI is filled with the AFI of the Function-Instance-EID, and the Locator is filled with the Function-Instance-EID. This way, a query on the flow 2-Tuple returns the Function-Instance-EID that the flow is assigned to.

7. QOS and Echo Measurements

8. Security Considerations

there are no security considerations related with this memo.

9. IANA Considerations

there are no IANA considerations related with this memo.

10. Acknowledgements

11. Normative References

- [I-D.rodriqueznatal-lisp-sdn]
Rodriguez-Natal, A., Cabellos-Aparicio, A.,
Barkai, S., Ermagan, V., Lewis, D., Maino, F.,
and D. Farinacci, "Software Defined Networking extensions
for the Locator/ID Separation Protocol", draft-
rodriqueznatal-lisp-sdn-00 (work in progress), February
2014.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The
Locator/ID Separation Protocol (LISP)", RFC 6830, January
2013.

Authors' Addresses

Sharon Barkai
ConteXtream Inc.
California
USA

Email: sbarkai@gmail.com

Dino Farinacci
lisppers.net
California
USA

Email: farinacci@gmail.com

David Meyer
Brocade
California
USA

Email: dmm@1-4-5.net

Fabio Maino
Cisco Systems
California
USA

Email: fmaino@cisco.com

Vina Ermagan
Cisco Systems
California
USA

Email: vermagan@cisco.com

Alberto Rodriguez-Natal
Technical University of Catalonia
Barcelona
Spain

Email: arnatal@ac.upc.edu

Albert Cabellos-Aparicio
Technical University of Catalonia
Barcelona
Spain

Email: acabello@ac.upc.edu

Internet Engineering Task Force
Internet-Draft
Intended status: Experimental
Expires: January 21, 2015

D. Farinacci
lispers.net
July 20, 2014

LISP Data-Plane Confidentiality
draft-farinacci-lisp-crypto-01

Abstract

This document describes a mechanism for encrypting LISP encapsulated traffic. The design describes how key exchange is achieved using existing LISP control-plane mechanisms as well as how to secure the LISP data-plane from third-party surveillance attacks.

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on January 21, 2015.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Overview	3
3. Diffie-Hellman Key Exchange	3
4. Encoding and Transmitting Key Material	4
5. Data-Plane Operation	6
6. Dynamic Rekeying	6
7. Future Work	7
8. Security Considerations	7
8.1. SAAG Support	7
8.2. LISP-Crypto Security Threats	8
9. IANA Considerations	8
10. References	8
10.1. Normative References	8
10.2. Informative References	9
Appendix A. Acknowledgments	9
Appendix B. Document Change Log	10
B.1. Changes to draft-farinacci-lisp-crypto-01.txt	10
B.2. Changes to draft-farinacci-lisp-crypto-00.txt	10
Author's Address	10

1. Introduction

The Locator/ID Separation Protocol [RFC6830] defines a set of functions for routers to exchange information used to map from non-routable Endpoint Identifiers (EIDs) to routable Routing Locators (RLOCs). LISP ITRs and PITRs encapsulate packets to ETRs and RTRs. Packets that arrive at the ITR or PITR are typically not modified. Which means no protection or privacy of the data is added. If the source host encrypts the data stream then the encapsulated packets can be encrypted but would be redundant. However, when plaintext packets are sent by hosts, this design can encrypt the user payload to maintain privacy on the path between the encapsulator (the ITR or PITR) to a decapsulator (ETR or RTR).

This draft has the following requirements for the solution space:

- o Do not require a separate Public Key Infrastructure (PKI) that is out of scope of the LISP control-plane architecture.
- o The budget for key exchange MUST be one round-trip time. That is, only a two packet exchange can occur.
- o Use symmetric keying so faster cryptography can be performed in the LISP data plane.
- o Avoid a third-party trust anchor if possible.

- o Provide for rekeying when secret keys are compromised.
- o At this time, encapsulated packet authentication is not a strong requirement.

2. Overview

The approach proposed in this draft is to not rely on the LISP mapping system to store security keys. This will provide for a simpler and more secure mechanism. Secret shared keys will be negotiated between the ITR and the ETR in Map-Request and Map-Reply messages. Therefore, when an ITR needs to obtain the RLOC of an ETR, it will get security material to compute a shared secret with the ETR.

The ITR can compute 3 shared-secrets per ETR the ITR is encapsulating to. And when the ITR encrypts a packet before encapsulation, it will identify the key it used for the crypto calculation so the ETR knows which key to use for decrypting the packet after decapsulation. By using key-ids in the LISP header, we can also get rekeying functionality.

3. Diffie-Hellman Key Exchange

LISP will use a Diffie-Hellman [RFC2631] key exchange sequence and computation for computing a shared secret. The Diffie-Hellman parameters will be passed in Map-Request and Map-Reply messages.

Here is a brief description how Diff-Hellman works:

ITR				ETR		
Secret	Public	Calculates	Sends	Calculates	Public	Secret
i	p,g		p,g -->			e
i	p,g,I	$g^i \bmod p = I$	I -->		p,g,I	e
i	p,g,I		<-- E	$g^e \bmod p = E$	p,g	e
i,s	p,g,I,E	$E^i \bmod p = s$		$I^e \bmod p = s$	p,g,I,E	e,s

Public-key exchange for computing a shared private key [DH]

Diffie-Hellman parameters 'p' and 'g' must be the same values used by the ITR and ETR. The ITR computes public-key 'I' and transmits 'I'

in a Map-Request packet. When the ETR receives the Map-Request, it uses parameters 'p' and 'g' to compute the ETR's public key 'E'. The ETR transmits 'E' in a Map-Reply message. At this point, the ETR has enough information to compute 's', the shared secret, by using 'I' as the base and the ETR's private key 'e' as the exponent. When the ITR receives the Map-Reply, it uses the ETR's public-key 'E' with the ITR's private key 'i' to compute the same 's' shared secret the ETR computed. The value 'p' is used as a modulus to create the width of the shared secret 's'.

4. Encoding and Transmitting Key Material

The Diffie-Hellman key material is transmitted in Map-Request and Map-Reply messages. Diffie-Hellman parameters are encoded in the LISP Security Type LCAF [LCAF].

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								
AFI = 16387										Rsvd1										Flags																			
Type = 11										Rsvd2										6 + n																			
Key Count										Rsvd3										Key Algorithm										Rsvd4	R								
Key Length										Key Material ...																													
										... Key Material																													
AFI = x										Locator Address ...																													

Diffie-Hellman parameters encoded in Key Material field

The 'Key Count' field encodes the number of {'Key-Length', 'Key-Material'} fields included in the encoded LCAF. A maximum number of keys that can be encoded are 3 keys, each identified by key-id 1, followed by key-id 2, and finally key-id 3.

The 'R' bit is not used for this use-case of the Security Type LCAF but is reserved for [LISP-DDT] security.

The 'Key Algorithm' encodes the cryptographic algorithm used. The following values are defined:

```

Null:      0
Group-ID:  1
AES:       2
3DES:      3
SHA-256:   4

```

When the 'Key Algorithm' value is 1 (Group-ID), the 'Key Material' field is encoded as:

```

      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
+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     Group ID                             |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     Public Key ...                         |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

Points to Key Material values from IANA Registry

The Group-ID values are defined in [RFC2409] and [RFC3526] which describe the Diffie Hellman parameters used for key exchange.

When the 'Key Algorithm' value is not 1 (Group-ID), the 'Key Material' field is encoded as:

```

      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
+-----+-----+-----+-----+-----+-----+-----+-----+
|  g-length  |                                     g-value ...           |
+-----+-----+-----+-----+-----+-----+-----+-----+
|  p-length  |                                     p-value ...           |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     Public Key ...                         |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

Key Length describes the length of the Key Material field

When an ITR or PITR sends a Map-Request, they will encode their own RLOC in Security Type LCAF format within the ITR-RLOCs field. When a ETR or RTR sends a Map-Reply, they will encode their RLOCs in Security Type LCAF format within the RLOC-record field of each EID-record supplied.

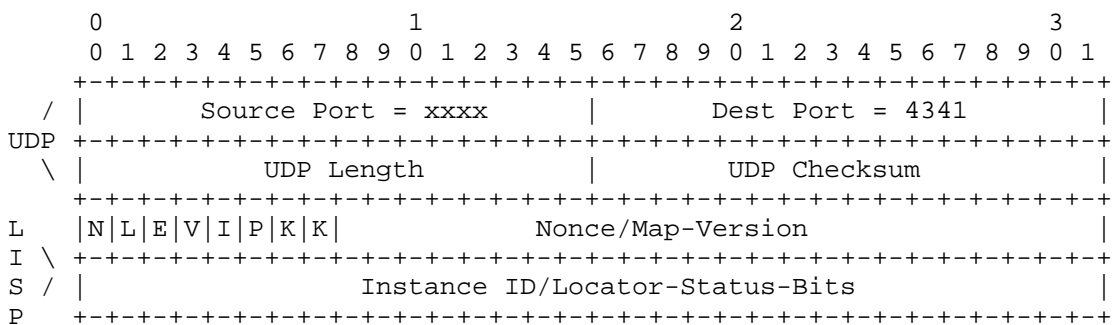
If an ITR or PITR sends a Map-Request with a Security Type LCAF included and the ETR or RTR does not want to have encapsulated traffic encrypted, they will return a Map-Reply with no RLOC records encoded with the Security Type LCAF. This signals to the ITR or PITR

that it should not encrypt traffic (it cannot encrypt traffic anyways since no ETR public-key was returned).

Likewise, if an ITR or PITR wish to include multiple key-ids in the Map-Request but the ETR or RTR wish to use some but not all of the key-ids, they return a Map-Reply only for those key-ids they wish to use.

5. Data-Plane Operation

The LISP encapsulation header [RFC6830] requires changes to encode the key-id for the key being used for encryption.



K-bits indicate when packet is encrypted and which key used

When the KK bits are 00, the encapsulated packet is not encrypted. When the value of the KK bits is 1, 2, or 3, it encodes the key-id of the secret keys computed during the Diffie-Hellman Map-Request/Map-Reply exchange.

When an ITR or PITR receives a packet to be encapsulated, they will first decide what key to use, encode the key-id into the LISP header, and use that key to encrypt all packet data that follows the LISP header. Therefore, the outer header, UDP header, and LISP header travel as plaintext.

6. Dynamic Rekeying

Since multiple keys can be encoded in both control and data messages, an ITR can encapsulate and encrypt with a specific key while it is negotiating other keys with the same ETR. Soon as an ETR or RTR returns a Map-Reply, it should be prepared to decapsulate and decrypt using the new keys computed with the new Diffie-Hellman parameters received in the Map-Request and returned in the Map-Reply.

RLOC-probing can be used to change keys by the ITR at any time. And when an initial Map-Request is sent to populate the ITR's map-cache, the Map-Requests flows across the mapping system where a single ETR from the Map-Reply RLOC-set will respond. If the ITR decides to use the other RLOCs in the RLOC-set, it MUST send a Map-Request directly to key negotiate with the ETR. This process may be used to test reachability from an ITR to an ETR initially when a map-cache entry is added for the first time, so an ITR can get both reachability status and keys negotiated with one Map-Request/Map-Reply exchange.

A rekeying event is defined to be when an ITR or PITR changes the p, g, or the public-key in a Map-Request. The ETR or RTR compares the p, g, and public-key it last received from the ITR for the key-id, and if any value has changed, it computes a new public-key of its own with the new p and g values from the Map-Request and returns it in the Map-Reply. Now a new shared secret is computed and can be used for the key-id for encryption by the ITR and decryption by the ETR. When the ITR or PITR starts this process of negotiating a new key, it must not use the corresponding key-id in encapsulated packets until it receives a Map-Reply from the ETR with the p and g values it expects (the values it sent in a Map-Request).

Note when RLOC-probing continues to maintain RLOC reachability and rekeying is not desirable, the ITR or RTR can either not include the Security Type LCAF in the Map-Request or supply the same key material as it recieved from the last Map-Reply from the ETR or RTR. This approach signals to the ETR or RTR that no rekeying event is requested.

7. Future Work

By using AES-GCM [RFC5116], or HMAC-CBC [AES-CBC], it has been suggested that encapsulated packet authentication (through encryption [RFC4106]) could be supported. There is current work in progress to investigate these techniques for the LISP data-plane. However, it will require encapsulation header changes to LISP.

For performance considerations, Elliptic-Curve Diffie Hellman (ECDH) can be used as specified in [RFC4492] to reduce CPU cycles required to compute shared secret keys.

8. Security Considerations

8.1. SAAG Support

The LISP working group will seek help from the SAAG working group for security advice. The SAAG will be involved early in the design process so they have early input and review.

8.2. LISP-Crypto Security Threats

Since ITRs and ETRs participate in key exchange over a public non-secure network, a man-in-the-middle (MITM) could circumvent the key exchange and compromise data-plane confidentiality. This can happen when the MITM is acting as a Map-Replier, provides its own public key so the ITR and the MITM generate a shared secret key among each other. If the MITM is in the data path between the ITR and ETR, it can use the shared secret key to decrypt traffic from the ITR.

Since LISP can secure Map-Replies by the authentication process specified in [LISP-SEC], the ITR can detect when a MITM has signed a Map-Reply for an EID-prefix it is not authoritative for. When an ITR determines the signature verification fails, it discards and does not reuse the key exchange parameters, avoids using the ETR for encapsulation, and issues a severe log message to the network administrator. Optionally, the ITR can send RLOC-probes to the compromised RLOC to determine if can reach the authoritative ETR. And when the ITR validates the signature of a Map-Reply, it can begin encrypting and encapsulating packets to the RLOC of ETR.

9. IANA Considerations

This draft requires the use of the registry that selects Diffie Hellman parameters. Rather than convey the key exchange parameters directly in LISP control packets, a Group-ID from the registry will be used. The Group-ID values are defined in [RFC2409] and [RFC3526].

10. References

10.1. Normative References

- [RFC2409] Harkins, D. and D. Carrel, "The Internet Key Exchange (IKE)", RFC 2409, November 1998.
- [RFC2631] Rescorla, E., "Diffie-Hellman Key Agreement Method", RFC 2631, June 1999.
- [RFC3526] Kivinen, T. and M. Kojo, "More Modular Exponential (MODP) Diffie-Hellman groups for Internet Key Exchange (IKE)", RFC 3526, May 2003.
- [RFC4106] Viega, J. and D. McGrew, "The Use of Galois/Counter Mode (GCM) in IPsec Encapsulating Security Payload (ESP)", RFC 4106, June 2005.

- [RFC4492] Blake-Wilson, S., Bolyard, N., Gupta, V., Hawk, C., and B. Moeller, "Elliptic Curve Cryptography (ECC) Cipher Suites for Transport Layer Security (TLS)", RFC 4492, May 2006.
- [RFC5116] McGrew, D., "An Interface and Algorithms for Authenticated Encryption", RFC 5116, January 2008.
- [RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, January 2013.

10.2. Informative References

- [AES-CBC] McGrew, D., Foley, J., and K. Paterson, "Authenticated Encryption with AES-CBC and HMAC-SHA", draft-mcgrew-aead-aes-cbc-hmac-sha2-03.txt (work in progress), .
- [DH] "Diffie-Hellman key exchange", Wikipedia
http://en.wikipedia.org/wiki/Diffie-Hellman_key_exchange, .
- [LCAF] Farinacci, D., Meyer, D., and J. Snijders, "LISP Canonical Address Format", draft-ietf-lisp-lcaf-04.txt (work in progress), .
- [LISP-DDT] Fuller, V., Lewis, D., Ermaagan, V., and A. Jain, "LISP Delegated Database Tree", draft-fuller-lisp-ddt-03 (work in progress), .
- [LISP-SEC] Maino, F., Ermagan, V., Cabellos, A., and D. Saucez, "LISP-Security (LISP-SEC)", draft-ietf-lisp-sec-06 (work in progress), .

Appendix A. Acknowledgments

The author would like to thank Dan Harkins, Brian Weis, Joel Halpern, Fabio Maino, Ed Lopez, and Roger Jorgensen for their interest, suggestions, and discussions about LISP data-plane security.

In addition, the support and suggestions from the SAAG working group were helpful and appreciative.

Appendix B. Document Change Log

B.1. Changes to draft-farinacci-lisp-crypto-01.txt

- o Posted July 2014.
- o Add Group-ID to the encoding format of Key Material in a Security Type LCAF and modify the IANA Considerations so this draft can use key exchange parameters from the IANA registry.
- o Indicate that the R-bit in the Security Type LCAF is not used by lisp-crypto.
- o Add text to indicate that ETRs/RTRs can negotiate less number of keys from which the ITR/PITR sent in a Map-Request.
- o Add text explaining how LISP-SEC solves the problem when a man-in-the-middle becomes part of the Map-Request/Map-Reply key exchange process.
- o Add text indicating that when RLOC-probing is used for RLOC reachability purposes and rekeying is not desired, that the same key exchange parameters should be used so a reallocation of a public key does not happen at the ETR.
- o Add text to indicate that ECDH can be used to reduce CPU requirements for computing shared secret-keys.

B.2. Changes to draft-farinacci-lisp-crypto-00.txt

- o Initial draft posted February 2014.

Author's Address

Dino Farinacci
lispers.net
San Jose, California
USA

Phone: 408-718-2001
Email: farinacci@gmail.com

Network Working Group
Internet-Draft
Intended status: Experimental
Expires: December 15, 2014

V. Moreno
Cisco Systems
D. Farinacci
lispers.net
June 13, 2014

Signal-Free LISP Multicast
draft-farinacci-lisp-signal-free-multicast-01

Abstract

When multicast sources and receivers are active at LISP sites, the core network is required to use native multicast so packets can be delivered from sources to group members. When multicast is not available to connect the multicast sites together, a signal-free mechanism can be used to allow traffic to flow between sites. The mechanism within here uses unicast replication and encapsulation over the core network for the data-plane and uses the LISP mapping database system so encapsulators at the source LISP multicast site can find de-encapsulators at the receiver LISP multicast sites.

Requirements Language

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].

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on December 15, 2014.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Definition of Terms	4
3. Reference Model	5
4. General Procedures	6
4.1. General Receiver-site Procedures	7
4.1.1. Multicast receiver detection	7
4.1.2. Receiver-site Registration	7
4.1.3. Consolidation of the replication-list	9
4.2. General Source-site Procedures	9
4.2.1. Multicast Tree Building at the Source-site	9
4.2.2. Multicast Destination Resolution	9
4.3. General LISP Notification Procedures	10
5. Source Specific Multicast Trees	10
5.1. Source directly connected to Source-ITRs	11
5.2. Source not directly connected to Source-ITRs	11
6. PIM Any Source Multicast Trees	11
7. Security Considerations	11
8. IANA Considerations	12
9. Acknowledgements	12
10. References	12
10.1. Normative References	12
10.2. Informative References	12
Appendix A. Document Change Log	13
A.1. Changes to draft-farinacci-lisp-signal-free-multicast-01	13
A.2. Changes to draft-farinacci-lisp-signal-free-multicast-00	13
Authors' Addresses	14

1. Introduction

When multicast sources and receivers are active at LISP sites, and the core network between the sites does not provide multicast support, a signal-free mechanism can be used to create an overlay that will allow multicast traffic to flow between sites and connect the multicast trees at the different sites.

The signal-free mechanism here proposed does not extend PIM over the overlay as proposed in [RFC6831], nor does the mechanism utilize direct signaling between the Receiver-ETRs and Sender-ITRs as described in [I-D.farinacci-lisp-mr-signaling]. The signal-free mechanism proposed reduces the amount of signaling required between sites to a minimum and is centered around the registration of Receiver-sites for a particular multicast-group or multicast-channel with the LISP Mapping System.

Registrations from the different receiver-sites will be merged at the Mapping System to assemble a multicast-replication-list inclusive of all RLOCs that lead to receivers for a particular multicast-group or multicast-channel. The replication-list for each specific multicast-entry is maintained as a LISP database mapping entry in the Mapping Database.

When the ITR at the source-site receives multicast traffic from sources at its site, the ITR can query the mapping system by issuing Map-Request messages for the (S,G) source and destination addresses in the packets received. The Mapping System will return the RLOC replication-list to the ITR, which the ITR will cache as per standard LISP procedure. Since the core is assumed to not support multicast, the ITR will replicate the multicast traffic for each RLOC on the replication-list and will unicast encapsulate the traffic to each RLOC. The combined function of replicating and encapsulating the traffic to the RLOCs in the replication-list is referred to as "rep-encapsulation" in this document.

The document describes the General Procedures and information encoding that are required at the Receiver-sites and Source-sites to achieve signal-free multicast interconnectivity. The General Procedures for Mapping System Notifications to different sites are also described. A section dedicated to the specific case of SSM trees discusses the implications to the General Procedures for SSM multicast trees over different topological scenarios. At this stage ASM trees are not supported with LISP Signal-Free multicast.

2. Definition of Terms

LISP related terms, notably Map-Request, Map-Reply, Ingress Tunnel Router (ITR), Egress Tunnel Router (ETR), Map-Server (MS) and Map-Resolver (MR) are defined in the LISP specification [RFC6830].

Extensions to the definitions in [RFC6830] for their application to multicast routing are documented in [RFC6831].

Terms defining interactions with the LISP Mapping System are defined in [RFC6833].

The following terms are consistent with the definitions in [RFC6830] and [RFC6831]. The terms are specific cases of the general terms and are here defined to facilitate the descriptions and discussions within this particular document.

Source: Multicast source end-point. Host originating multicast packets.

Receiver: Multicast group member end-point. Host joins multicast group as a receiver of multicast packets sent to the group.

Receiver-site: LISP site where multicast receivers are located.

Source-site: LISP site where multicast sources are located.

RP-site: LISP site where an ASM PIM Rendezvous Point is located. The RP-site and the Source-site may be the same in some situations.

Receiver-ETR: LISP xTR at the Receiver-site. This is a multicast ETR.

Source-ITR: LISP xTR at the Source-site. This is a multicast ITR.

RP-xTR: LISP xTR at the RP-site. This is typically a multicast ITR.

Replication-list: Mapping-entry containing the list of RLOCs that have registered Receivers for a particular multicast-entry.

Multicast-entry: A tuple identifying a multicast tree. Multicast-entries are in the form of (S-prefix, G-prefix).

Rep-encapsulation: The process of replicating and then encapsulating traffic to multiple RLOCs.

3. Reference Model

The reference model that will be used for the discussion of the Signal-Free multicast tree interconnection is illustrated in Figure 1.

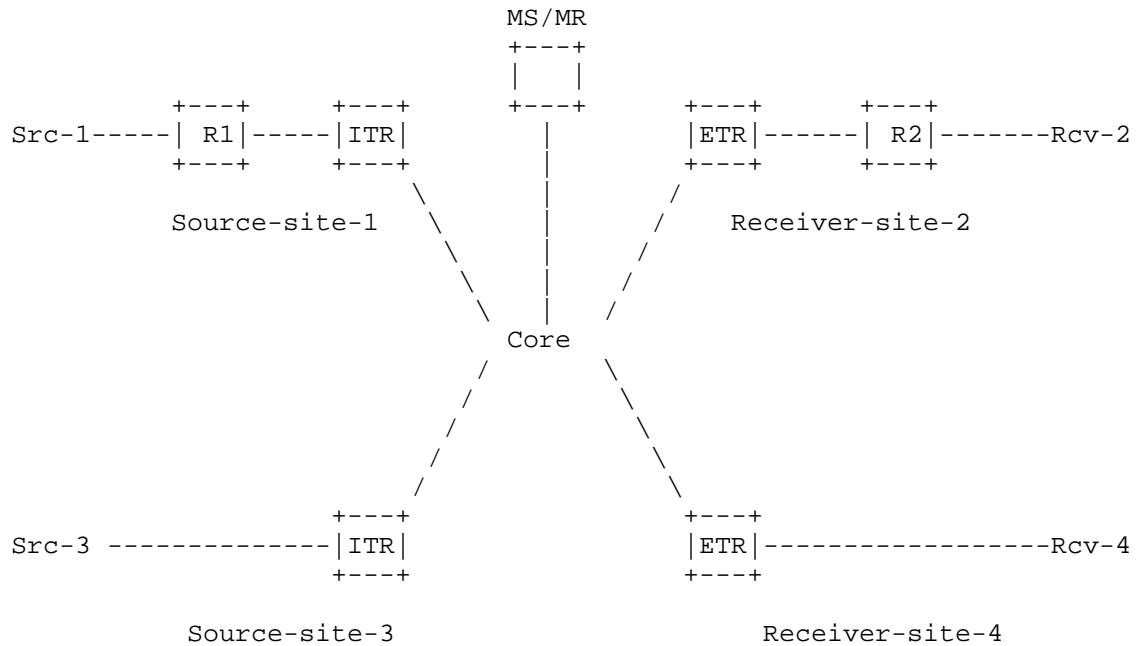


Figure 1: LISP Multicast Generic Reference Model

Sites 1 and 3 are Source-sites.

Source-site-3 presents a Source (Src-3) that is directly connected to the Source-ITR

Source-site-1 presents a Source (Src-1) that is one hop or more away from the Source-ITR

Receiver-site-2 and 4 are receiver sites with not-directly connected and directly connected Receiver end-points respectively

R1 is a router in Source-site-1.

R2 is a PIM router at the Receiver-site.

The Map-Servers and Resolvers are reachable in the RLOC space in the Core, only one is shown for illustration purposes, but these can be many or even part of a DDT tree.

The procedures for interconnecting multicast Trees over an overlay can be broken down into three functional areas:

- o Receiver-site procedures
- o Source-site procedures
- o LISP notification procedures

The receiver site procedures will be common for most tree types and topologies.

The procedures at the source site can vary depending on the type of trees being interconnected as well as based on the topological relation between sources and source-site xTRs. For ASM trees, a special case of the Source-site is the RP-site for which a variation of the Source-site procedures may be necessary if ASM trees are to be supported in future specifications of LISP Signal-Free multicast.

The LISP notification procedures between sites are normalized for the different possible scenarios. Certain scenarios may benefit from a simplified notification mechanism or no notification requirement at all.

4. General Procedures

The interconnection of multicast trees across different LISP sites involves the following procedures to build the necessary multicast distribution trees across sites.

1. The presence of multicast Receiver end-points is detected by the Receiver-ETRs at the Receiver-sites.
2. Receiver-ETRs register their RLOCs as part of the replication-list for the multicast-entry the detected Receivers subscribe to.
3. The Mapping-system merges all receiver-ETR or delivery-group RLOCs to build a comprehensive replication-list inclusive of all Receiver-sites for each multicast-entry.
4. LISP Map-Notify messages should be sent to the Source-ITR informing of any changes in the replication-list.

5. Multicast-tree building at the Source-site is initiated when the Source-ITR receives the LISP Notification.

Once the multicast distribution trees are built, the following forwarding procedures may take place:

1. The Source sends multicast packets to the multicast group destination address.
2. Multicast traffic follows the multicast tree built at the Source-site and makes its way to the Source-ITRs.
3. The Source-ITR will issue a map-request to resolve the replication-list for the multicast-entry.
4. The Mapping System responds to the Source-ITR with a map-reply containing the replication-list for the multicast group requested.
5. The Source-ITR caches the replication-list received in the map-reply for the multicast-entry.
6. Multicast traffic is rep-encapsulated. That is, the packet is replicated for each RLOC in the replication-list and then encapsulated to each one.

4.1. General Receiver-site Procedures

4.1.1. Multicast receiver detection

When the Receiver-ETRs are directly connected to the Receivers (e.g. Receiver-site-4 in Figure 1), the Receiver-ETRs will receive IGMP Reports from the Receivers indicating which group the Receivers wish to subscribe to. Based on these IGMP Reports, the receiver-ETR is made aware of the presence of Receivers as well as which group they are interested in.

When the Receiver-ETRs are several hops away from the Receivers (e.g. Receiver-site-2 in Figure 1), the Receiver-ETRs will receive PIM join messages which will allow the Receiver-ETR to know that there are multicast Receivers at the site and also learn which multicast group the Receivers are for.

4.1.2. Receiver-site Registration

Once the Receiver-ETRs detect the presence of Receivers at the Receiver-site, the Receiver-ETRs will issue Map-Register messages to

include the Receiver-ETR RLOCs in the replication-list for the multicast-entry the Receivers joined.

The Map-Register message will use the multicast-entry (Source, Group) tuple as its EID record type with the Receiver-ETR RLOCs conforming the locator set.

The EID in the Map-Register message must be encoded using the Multicast Information LCAF type defined in [I-D.ietf-lisp-lcaf]. The R, L and J bits in the Multicast-info LCAF frame are not used and should be set to zero.

The RLOC in the Map-Register message must be encoded using the Replication List Entry (RLE) LCAF type defined in [I-D.ietf-lisp-lcaf] with the Level Value fields for all entries set to 128 (decimal).

The encoding described above must be used consistently for Map-Register messages, entries in the Mapping Database, Map-reply messages as well as the map-cache at the Source-ITRs.

The Map-Register messages [RFC6830] sent by the receiver-ETRs should have the following bits set as here specified:

1. merge-request-bit set to 1. The Map-Register messages must be sent with "Merge Semantics". The Map-Server will receive registrations from a multitude of Receiver-ETRs. The Map-Server will merge the registrations for common EIDs and maintain a consolidated replication-list for each multicast-entry.
2. want-map-notify-bit (M) set to 0. This tells the Mapping System that the receiver-ETR does not expect to receive Map-Notify messages as it does not need to be notified of all changes to the replication-list.
3. proxy-reply-bit (P) set to 1. The merged replication-list is kept in the Map-Servers. By setting the proxy-reply bit, the receiver-ETRs instruct the Mapping-system to proxy reply to map-requests issued for the multicast entries.

Map-Register messages for a particular multicast-entry should be sent for every receiver detected, even if previous receivers have been detected for the particular multicast-entry. This allows the replication-list to remain up to date.

4.1.3. Consolidation of the replication-list

The Map-Server will receive registrations from a multitude of Receiver-ETRs. The Map-Server will merge the registrations for common EIDs and consolidate a replication-list for each multicast-entry.

4.2. General Source-site Procedures

Source-ITRs must register the unicast EIDs of any Sources or Rendezvous Points that may be present on the Source-site. In other words, it is assumed that the Sources and RPs are LISP EIDs.

The registration of the unicast EIDs for the Sources or Rendezvous Points allows the map-server to know where to send Map-Notify messages to. Therefore, the Source-ITR must register the unicast S-prefix EID with the want-map-notify-bit set in order to receive Map-Notify messages whenever there is a change in the replication-list.

4.2.1. Multicast Tree Building at the Source-site

When the source site receives the Map-Notify messages from the mapping system as described in Section 4.3, it will initiate the process of building a multicast distribution tree that will allow the multicast packets from the Source to reach the Source-ITR.

The Source-ITR will issue a PIM join for the multicast-entry for which it received the Map-Notify message. The join will be issued in the direction of the source or in the direction of the RP for the SSM and ASM cases respectively.

4.2.2. Multicast Destination Resolution

On reception of multicast packets, the source-ITR must obtain the replication-list for the (S,G) addresses in the packets.

In order to obtain the replication-list, the Source-ITR must issue a Map-Request message in which the EID is the (S, G) multicast tuple which is encoded using the Multicast Info LCAF type defined in [I-D.ietf-lisp-lcaf].

The Mapping System (most likely the Map-Server) will Map-reply with the merged replication-list maintained in the Mapping System. The Map-reply message must follow the format defined in [RFC6830], its EID must be encoded using the Multicast Info LCAF type and the corresponding RLOC-records must be encoded using the RLE LCAF type. Both LCAF types defined in [I-D.ietf-lisp-lcaf].

4.3. General LISP Notification Procedures

The Map-Server will issue LISP Map-Notify messages to inform the Source-site of the presence of receivers for a particular multicast group over the overlay.

Updated Map-Notify messages should be issued every time a new registration is received from a Receiver-site. This guarantees that the source-sites are aware of any potential changes in the multicast-distribution-list membership.

The Map-Notify messages carry (S,G) multicast EIDs encoded using the Multicast Info LCAF type defined in [I-D.ietf-lisp-lcaf].

Map-Notify messages will be sent by the Map-Server to the RLOCs with which the unicast S-prefix EID was registered.

When both the Receiver-sites and the Source-sites register to the same Map-Server, the Map-Server has all the necessary information to send the Map-Notify messages to the Source-site.

When the Map-Servers are distributed in a DDT, the Receiver-sites may register to one Map-Server while the Source-site registers to a different Map-Server. In this scenario, the Map-Server for the receiver sites must resolve the unicast S-prefix EID in the DDT per standard LISP lookup procedures and obtain the necessary information to send the Map-Notify messages to the Source-site. The Map-Notify messages must be sent with an authentication length of 0 as they would not be authenticated.

When the Map-Servers are distributed in a DDT, different Receiver-sites may register to different Map-Servers. This is an unsupported scenario with the currently defined mechanisms.

5. Source Specific Multicast Trees

The interconnection of Source Specific Multicast (SSM) Trees across sites will follow the General Receiver-site Procedures described in Section 4.1 on the Receiver-sites.

The Source-site Procedures will vary depending on the topological location of the Source within the Source-site as described in Section 5.1 and Section 5.2 .

5.1. Source directly connected to Source-ITRs

When the Source is directly connected to the source-ITR, it is not necessary to trigger signaling to build a local multicast tree at the Source-site. Therefore Map-Notify messages may not be required to initiate building of the multicast tree at the Source-site.

Map-Notify messages are still required to ensure that any changes to the replication-list are communicated to the Source-site so that the map-cache at the Source-ITRs is kept updated.

5.2. Source not directly connected to Source-ITRs

The General LISP Notification Procedures described in Section 4.3 must be followed when the Source is not directly connected to the source-ITR. On reception of Map-Notify messages, local multicast signaling must be initiated at the Source-site per the General Source Site Procedures for Multicast Tree building described in Section 4.2.1.

In the SSM case, the IP address of the Source is known and it is also registered with the LISP mapping system. Thus, the mapping system may resolve the mapping for the Source address in order to send Map-Notify messages to the correct source-ITR.

6. PIM Any Source Multicast Trees

LISP signal-free multicast will not support ASM Trees at this time. A future revision of this specification may include procedures for PIM ASM support.

PIM ASM in shared-tree only mode could be supported in the scenario where the root of the shared tree (the PIM RP) is placed at the source site.

7. Security Considerations

[I-D.ietf-lisp-sec] defines a set of security mechanisms that provide origin authentication, integrity and anti-replay protection to LISP's EID-to-RLOC mapping data conveyed via mapping lookup process. LISP-SEC also enables verification of authorization on EID-prefix claims in Map-Reply messages.

Additional security mechanisms to protect the LISP Map-Register messages are defined in [RFC6833].

The security of the Mapping System Infrastructure depends on the particular mapping database used. The [I-D.ietf-lisp-ddt]

specification, as an example, defines a public-key based mechanism that provides origin authentication and integrity protection to the LISP DDT protocol.

Map-Replies received by the source-ITR can be signed (by the Map-Server) so the ITR knows the replication-list is from a legit source.

Data-plane encryption can be used when doing unicast rep-encapsulation as described in [I-D.farinacci-lisp-crypto]. For further study we will look how to do multicast rep-encapsulation.

8. IANA Considerations

This document has no IANA implications

9. Acknowledgements

The authors want to thank Greg Shepherd, Joel Halpern and Sharon Barkai for their insightful contribution to shaping the ideas in this document. Thanks also goes to Jimmy Kyriannis, Paul Vinciguerra, and Florin Coras for testing an implementation of this draft.

10. References

10.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC3618] Fenner, B. and D. Meyer, "Multicast Source Discovery Protocol (MSDP)", RFC 3618, October 2003.
- [RFC4601] Fenner, B., Handley, M., Holbrook, H., and I. Kouvelas, "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)", RFC 4601, August 2006.
- [RFC4607] Holbrook, H. and B. Cain, "Source-Specific Multicast for IP", RFC 4607, August 2006.

10.2. Informative References

- [I-D.farinacci-lisp-crypto] Farinacci, D., "LISP Data-Plane Confidentiality", draft-farinacci-lisp-crypto-00 (work in progress), February 2014.

- [I-D.farinacci-lisp-mr-signaling]
Farinacci, D. and M. Napierala, "LISP Control-Plane Multicast Signaling", draft-farinacci-lisp-mr-signaling-04 (work in progress), March 2014.
- [I-D.ietf-lisp-ddt]
Fuller, V., Lewis, D., Ermagan, V., and A. Jain, "LISP Delegated Database Tree", draft-ietf-lisp-ddt-01 (work in progress), March 2013.
- [I-D.ietf-lisp-lcaf]
Farinacci, D., Meyer, D., and J. Snijders, "LISP Canonical Address Format (LCAF)", draft-ietf-lisp-lcaf-05 (work in progress), May 2014.
- [I-D.ietf-lisp-sec]
Maino, F., Ermagan, V., Cabellos-Aparicio, A., and D. Saucez, "LISP-Security (LISP-SEC)", draft-ietf-lisp-sec-06 (work in progress), April 2014.
- [RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, January 2013.
- [RFC6831] Farinacci, D., Meyer, D., Zwiebel, J., and S. Venaas, "The Locator/ID Separation Protocol (LISP) for Multicast Environments", RFC 6831, January 2013.
- [RFC6833] Fuller, V. and D. Farinacci, "Locator/ID Separation Protocol (LISP) Map-Server Interface", RFC 6833, January 2013.

Appendix A. Document Change Log

A.1. Changes to draft-farinacci-lisp-signal-free-multicast-01

- o Posted June 2014.
- o Changes based on implementation experience of this draft.

A.2. Changes to draft-farinacci-lisp-signal-free-multicast-00

- o Posted initial draft February 2014.

Authors' Addresses

Victor Moreno
Cisco Systems
170 Tasman Drive
San Jose, California 95134
USA

Email: vimoreno@cisco.com

Dino Farinacci
lispers.net
San Jose, CA 95120
USA

Email: farinacci@gmail.com

Network Working Group
Internet-Draft
Intended status: Informational
Expires: April 27, 2015

L. Iannone
Telecom ParisTech
R. Jorgensen
Bredbandsfylket Troms
D. Conrad
Virtualized, LLC
G. Huston
APNIC - Asia Pacific Network Information Center
October 24, 2014

LISP EID Block Management Guidelines
draft-ietf-lisp-eid-block-mgmt-03.txt

Abstract

This document proposes a framework for the management of the LISP EID Prefix. The framework described relies on hierarchical distribution of the address space, granting temporary usage of sub-prefixes of such space to requesting organizations.

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on April 27, 2015.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect

to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Requirements Notation	2
2. Introduction	2
3. Definition of Terms	3
4. EID Prefix Registration Policy	3
5. EID Prefixes Registration Requirements	4
6. EID Prefix Request Template	4
7. Policy Validity Period	6
8. Security Considerations	6
9. Acknowledgments	6
10. IANA Considerations	7
11. References	7
11.1. Normative References	7
11.2. Informative References	7
Appendix A. LISP Terms	8
Appendix B. Document Change Log	11
Authors' Addresses	11

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

The Locator/ID Separation Protocol (LISP - [RFC6830]) and related mechanisms ([RFC6831], [RFC6832], [RFC6833], [RFC6834], [RFC6835], [RFC6836], [RFC6837]) separates the IP addressing space into two logical spaces, the End-point Identifier (EID) space and the Routing LOCator (RLOC) space. The first space is used to identify communication end-points, while the second is used to locate EIDs in the Internet routing infrastructure topology.

The document [I-D.ietf-lisp-eid-block] requested an IPv6 address block reservation exclusively for use as EID prefixes in the LISP experiment. The rationale, intent, size, and usage of the EID address block are described in [I-D.ietf-lisp-eid-block].

This document proposes a management framework for the registration of EID prefixes from that block, allowing the requesting organisation

exclusive use of those EID prefixes limited to the duration of the LISP experiment.

3. Definition of Terms

This document does not introduce any new terms related to the set of LISP Specifications ([RFC6830], [RFC6831], [RFC6832], [RFC6833], [RFC6834], [RFC6835], [RFC6836], [RFC6837]). To help the reading of this document the terminology introduced by LISP is summarized in Appendix A.

4. EID Prefix Registration Policy

The request registration of EID prefixes MUST be done under the following policies:

1. EID prefixes are made available in the reserved space on a temporary basis and for experimental uses. The requester of an experimental prefix MUST provide a short description of the intended use or experiment that will be carried out (see Section 6). If the prefix will be used for activities not documented in the original description, the renewal of the registration may be denied.
2. EID prefix registrations SHOULD be renewed on a regular basis to ensure their use by active participants in the experiment. The registration period is proposed to be 12 months. Registration renewal SHOULD NOT cause a change in the registered EID prefix. The conditions of registration renewal should no different to the conditions of registration.
3. When an EID prefix registration is removed from the registry, then the reuse of the EID prefix in a subsequent registration on behalf of a different end user should be avoided where possible. If the considerations of overall usage of the EID block prefix requires reuse of a previously registered EID prefix, then a minimum delay of at least one week between removal and subsequent registration SHOULD be applied by the registry operator.
4. All registrations of EID prefixes cease at the time of the expiration of the reserved experimental LISP EID Block. The further disposition of these prefixes and the associated registry entries is to be specified in the announcement of the cessation of this experiment.

5. EID Prefixes Registration Requirements

All EID prefix registrations MUST respect the following requirements:

1. All EID prefix registrations MUST use a globally unique EID prefix.
2. If there is more than one registry operator, all operators MUST use the same registry management policies and practices.
3. The EID Prefix registration information as specified in Section 6, MUST be collected upon initial registration and renewal, and made publicly available through interfaces allowing both retrieval of specific registration details (search) and enumeration of the entire registry contents (e.g., [I-D.ietf-weirds-rdap-sec], whois, http, or similar access methods).
4. The registry operator MUST permit the delegation of EID prefixes in the reverse DNS space to holders of registered EID prefixes.
5. Anyone can obtain an entry in the EID prefix registry, on the understanding that the prefix so registered is for the exclusive use in the LISP experimental network, and that their registration details (as specified in Section 6) are openly published in the EID prefix registry.

6. EID Prefix Request Template

The following is a basic request template for prefix registration so to ensure a uniform process. Such a template is inspired by the IANA Private Enterprise Number online request form (<http://pen.iana.org/pen/PenApplication.page>).

Note that all details in this registration become part of the registry, and will be published in the LISP EID Prefix Registry.

The EID Prefix Request template MUST at minimum contain:

1. Organization (In case of individuals requesting an EID prefix this section can be left empty)
 - (a) Organization Name
 - (b) Organization Address
 - (c) Organization Phone

2. Contact Person (Mandatory)

- (a) Name
- (b) Address
- (c) Phone
- (d) Fax (optional)
- (e) Email

3. EID Prefix Request (Mandatory)

- (a) Prefix Size
- (b) Prefix Size Rationale
- (c) Lease Period
 - + Note Well: All EID Prefix registrations will be valid until the earlier date of 12 months from the date of registration or 31 December 2017.
 - + All registrations may be renewed by the applicant for further 12 month periods, ending on 31 December 2017.
 - + According to the 3+3 year experimentation plan, defined in [I-D.ietf-lisp-eid-block], all registrations MUST end by 31 December 2017, unless the IETF community decides to grant a permanent LISP EID address block. In the latter case, registrations following the present document policy MUST end by 31 December 2020 and a new policy (to be decided - see Section 7) will apply starting 1 January 2021.

4. Experiment Description

- (a) Experiment and Deployment Description
- (b) Interoperability with existing LISP deployments
- (c) Interoperability with Legacy Internet

5. Reverse DNS Servers (Optional)

- (a) Name server name:
- (b) Name server address:

(c) Name server name:

(d) Name server address:

(Repeat if necessary)

7. Policy Validity Period

Policy outlined in the present document is tied to the existence of the experimental LISP EID block requested in [I-D.ietf-lisp-eid-block] and valid until 31 December 2017.

If the IETF decides to transform the block in a permanent allocation, the LISP EID block reserved usage period will be extended for three years (until 31 December 2020) so to give time to the IETF to define, following the policies outlined in [RFC5226], the final size of the EID block and create a transition plan, while the policy in the present document will still apply.

Note that, as stated in [I-D.ietf-lisp-eid-block], the transition of the EID block into a permanent allocation, has the potential to pose policy issues (as recognized in [RFC2860], section 4.3) and hence discussion with the IANA, the RIR communities, and the IETF community will be necessary to determine appropriate policy for permanent EID prefix management, which will be effective starting 1 January 2021.

8. Security Considerations

This document does not introduce new security threats in the LISP architecture nor in the Legacy Internet architecture.

For accountability reasons, and in line with the security considerations in [RFC7020], each registration request MUST contain accurate information on the requesting entity (company, institution, individual, etc.) and valid and accurate contact information of a referral person (see Section 6).

9. Acknowledgments

Thanks to J. Curran, A. Severin, B. Haberman, T. Manderson, D. Lewis, D. Farinacci, M. Binderberger, for their helpful comments.

The work of Luigi Iannone has been partially supported by the ANR-13-INFR-0009 LISP-Lab Project (www.lisp-lab.org) and the EIT KIC ICT-Labs SOFNETS Project.

10. IANA Considerations

This document provides only management guidelines for the reserved LISP EID prefix requested in [I-D.ietf-lisp-eid-block].

There is an operational requirement for an EID registration service that ensures uniqueness of EIDs according to the requirements described in Section 5. Furthermore, there is an operational requirement for EID registration service that allows a lookup of the contact information of the entity that registered the EID.

IANA is to ensure both of these services are provided in a globally uniform fashion for the duration of the experiment.

11. References

11.1. Normative References

- [I-D.ietf-lisp-eid-block]
Iannone, L., Lewis, D., Meyer, D., and V. Fuller, "LISP EID Block", draft-ietf-lisp-eid-block-09 (work in progress), July 2014.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC4632] Fuller, V. and T. Li, "Classless Inter-domain Routing (CIDR): The Internet Address Assignment and Aggregation Plan", BCP 122, RFC 4632, August 2006.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

11.2. Informative References

- [I-D.ietf-weirds-rdap-sec]
Hollenbeck, S. and N. Kong, "Security Services for the Registration Data Access Protocol", draft-ietf-weirds-rdap-sec-06 (work in progress), February 2014.
- [RFC2860] Carpenter, B., Baker, F., and M. Roberts, "Memorandum of Understanding Concerning the Technical Work of the Internet Assigned Numbers Authority", RFC 2860, June 2000.
- [RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, January 2013.

- [RFC6831] Farinacci, D., Meyer, D., Zwiebel, J., and S. Venaas, "The Locator/ID Separation Protocol (LISP) for Multicast Environments", RFC 6831, January 2013.
- [RFC6832] Lewis, D., Meyer, D., Farinacci, D., and V. Fuller, "Interworking between Locator/ID Separation Protocol (LISP) and Non-LISP Sites", RFC 6832, January 2013.
- [RFC6833] Fuller, V. and D. Farinacci, "Locator/ID Separation Protocol (LISP) Map-Server Interface", RFC 6833, January 2013.
- [RFC6834] Iannone, L., Saucez, D., and O. Bonaventure, "Locator/ID Separation Protocol (LISP) Map-Versioning", RFC 6834, January 2013.
- [RFC6835] Farinacci, D. and D. Meyer, "The Locator/ID Separation Protocol Internet Groper (LIG)", RFC 6835, January 2013.
- [RFC6836] Fuller, V., Farinacci, D., Meyer, D., and D. Lewis, "Locator/ID Separation Protocol Alternative Logical Topology (LISP+ALT)", RFC 6836, January 2013.
- [RFC6837] Lear, E., "NERD: A Not-so-novel Endpoint ID (EID) to Routing Locator (RLOC) Database", RFC 6837, January 2013.
- [RFC7020] Housley, R., Curran, J., Huston, G., and D. Conrad, "The Internet Numbers Registry System", RFC 7020, August 2013.

Appendix A. LISP Terms

LISP operates on two name spaces and introduces several new network elements. This section provides high-level definitions of the LISP name spaces and network elements and as such, it must not be considered as an authoritative source. The reference to the authoritative document for each term is included in every term description.

Legacy Internet: The portion of the Internet that does not run LISP and does not participate in LISP+ALT or any other mapping system.

LISP site: A LISP site is a set of routers in an edge network that are under a single technical administration. LISP routers that reside in the edge network are the demarcation points to separate the edge network from the core network. See [RFC6830] for more details.

Endpoint ID (EID): An EID is a 32-bit (for IPv4) or 128-bit (for IPv6) value used in the source and destination address fields of the first (most inner) LISP header of a packet. A packet that is emitted by a system contains EIDs in its headers and LISP headers are prepended only when the packet reaches an Ingress Tunnel Router (ITR) on the data path to the destination EID. 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. See [RFC6830] for more details.

EID-prefix: A power-of-two block of EIDs that are allocated to a site by an address allocation authority. See [RFC6830] for more details.

EID-Prefix Aggregate: A set of EID-prefixes said to be aggregatable in the [RFC4632] sense. That is, an EID-Prefix aggregate is defined to be a single contiguous power-of-two EID-prefix block. A prefix and a length characterize such a block. See [RFC6830] for more details.

Routing LOCator (RLOC): A RLOC is an IPv4 or IPv6 address of an egress tunnel router (ETR). A RLOC is the output of an EID-to-RLOC mapping lookup. An EID maps to one or more RLOCs. Typically, RLOCs are numbered from topologically aggregatable blocks that are assigned to a site at each point to which it attaches to the global Internet; where the topology is defined by the connectivity of provider networks, RLOCs can be thought of as Provider Aggregatable (PA) addresses. See [RFC6830] for more details.

EID-to-RLOC Mapping: A binding between an EID-Prefix and the RLOC-set that can be used to reach the EID-Prefix. The general term "mapping" always refers to an EID-to-RLOC mapping. See [RFC6830] for more details.

Ingress Tunnel Router (ITR): An Ingress Tunnel Router (ITR) is a router that accepts receives IP packets from site end-systems on one side and sends LISP-encapsulated IP packets toward the Internet on the other side. The router treats the "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. See [RFC6830] for more details.

Egress Tunnel Router (ETR): An Egress Tunnel Router (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. An ETR router 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. See [RFC6830] for more details.

Proxy ITR (PITR): A Proxy-ITR (PITR) acts like an ITR but does so on behalf of non-LISP sites which send packets to destinations at LISP sites. See [RFC6832] for more details.

Proxy ETR (PETR): A Proxy-ETR (PETR) acts like an ETR but does so on behalf of LISP sites which send packets to destinations at non-LISP sites. See [RFC6832] for more details.

Map Server (MS): A network infrastructure component that learns EID-to-RLOC mapping entries from an authoritative source (typically an ETR). A Map Server publishes these mappings in the distributed mapping system. See [RFC6833] for more details.

Map Resolver (MR): A network infrastructure component that accepts LISP Encapsulated Map-Requests, typically from an ITR, quickly determines whether or not the destination IP address is part of the EID namespace; if it is not, a Negative Map-Reply is immediately returned. Otherwise, the Map Resolver finds the appropriate EID-to-RLOC mapping by consulting the distributed mapping database system. See [RFC6833] for more details.

The LISP Alternative Logical Topology (ALT): The virtual overlay network made up of tunnels between LISP+ALT Routers. The Border Gateway Protocol (BGP) runs between ALT Routers and is used to carry reachability information for EID-prefixes. The ALT provides a way to forward Map-Requests toward the ETR that "owns" an EID-prefix. See [RFC6836] for more details.

ALT Router: The device on which runs the ALT. The ALT is a static network built using tunnels between ALT Routers. These routers are deployed in a roughly-hierarchical mesh in which routers at each level in the topology are responsible for aggregating EID-Prefixes learned from those logically "below" them and advertising summary prefixes to those logically "above" them. Prefix learning and propagation between ALT Routers is done using BGP. When an ALT Router receives an ALT Datagram, it looks up the destination EID in its forwarding table (composed of EID-Prefix routes it learned from neighboring ALT Routers) and forwards it to the logical next-hop on the overlay network. The primary function of LISP+ALT routers is to provide a lightweight forwarding infrastructure for LISP control-plane messages (Map-Request and Map-Reply), and to transport data packets when the packet has the

same destination address in both the inner (encapsulating) destination and outer destination addresses ((i.e., a Data Probe packet). See [RFC6830] for more details.

Appendix B. Document Change Log

Version 03 Posted October 2014.

- o Re-worded the document so to avoid confusion on "allocation" and "assignement". The document now reffers to "registration". As for comments by G. Huston and M. Binderberger.

Version 02 Posted July 2014.

- o Deleted the trailing paragraph of Section 4, as for discussion in the mailing list.
- o Deleted the fees policy as of suggestion of G. Huston and discussion during 89th IETF.
- o Re-phrased the availability of the registration information requirement avoiding putting specific numbers (previously requiring 99% up time), as of suggestion of G. Huston and discussion during 89th IETF.

Version 01 Posted February 2014.

- o Dropped the reverse DNS requirement as for discussion during the 88th IETF meeting.
- o Dropped the minimum allocation requirement as for discussion during the 88th IETF meeting.
- o Changed Section 7 from "General Consideration" to "Policy Validity Period", according to J. Curran feedback. The purpose of the section is just to clearly state the period during which the policy applies.

Version 00 Posted December 2013.

- o Rename of draft-iannone-lisp-eid-block-mgmt-03.txt.

Authors' Addresses

Luigi Iannone
Telecom ParisTech

Email: ggx@gigix.net

Roger Jorgensen
Bredbandsfylket Troms

Email: rogerj@gmail.com

David Conrad
Virtualized, LLC

Email: drc@virtualized.org

Geoff Huston
APNIC - Asia Pacific Network Information Center

Email: gih@apnic.net

Network Working Group
Internet-Draft
Intended status: Informational
Expires: April 27, 2015

A. Cabellos
UPC-BarcelonaTech
D. Saucez (Ed.)
INRIA
October 24, 2014

An Architectural Introduction to the Locator/ID Separation Protocol
(LISP)
draft-ietf-lisp-introduction-07.txt

Abstract

This document describes the architecture of the Locator/ID Separation Protocol (LISP), making it easier to read the rest of the LISP specifications and providing a basis for discussion about the details of the LISP protocols. This document is used for introductory purposes, more details can be found in RFC6830, the protocol specification.

Requirements Language

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 RFC 2119 [RFC2119].

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on April 27, 2015.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Definition of Terms	4
3. LISP Architecture	4
3.1. Design Principles	4
3.2. Overview of the Architecture	4
3.3. Data-Plane	7
3.3.1. LISP Encapsulation	7
3.3.2. LISP Forwarding State	8
3.4. Control-Plane	9
3.4.1. LISP Mappings	9
3.4.2. Mapping System Interface	9
3.4.3. Mapping System	10
3.5. Interworking Mechanisms	13
4. LISP Operational Mechanisms	13
4.1. Cache Management	14
4.2. RLOC Reachability	14
4.3. ETR Synchronization	16
4.4. MTU Handling	16
5. Mobility	16
6. Multicast	17
7. Security	18
8. Use Cases	19
8.1. Traffic Engineering	19
8.2. LISP for IPv6 Co-existence	19
8.3. LISP for Virtual Private Networks	20
8.4. LISP for Virtual Machine Mobility in Data Centers	20
9. Security Considerations	21
10. IANA Considerations	21
11. Acknowledgements	21
12. References	21
12.1. Normative References	21
12.2. Informative References	22
Appendix A. A Brief History of Location/Identity Separation	24
A.1. Old LISP Models	24
Authors' Addresses	25

1. Introduction

This document introduces the Locator/ID Separation Protocol (LISP) [RFC6830] architecture, its main operational mechanisms and its design rationale. Fundamentally, LISP is built following a well-known architectural idea: decoupling the IP address overloaded semantics. Indeed and as pointed out by [Chiappa], currently IP addresses both identify the topological location of a network attachment point as well as the node's identity. However, nodes and routing have fundamentally different requirements, routing systems require that addresses are aggregatable and have topological meaning, while nodes require to be identified independently of their current location [RFC4984].

LISP creates two separate namespaces, EIDs (End-host IDentifiers) and RLOCs (Routing LOCators), both are typically syntactically identical to the current IPv4 and IPv6 addresses. EIDs are used to uniquely identify nodes irrespective of their topological location and are typically routed intra-domain. RLOCs are assigned topologically to network attachment points and are typically routed inter-domain. With LISP, the edge of the Internet (where the nodes are connected) and the core (where inter-domain routing occurs) can be logically separated and interconnected by LISP-capable routers. LISP also introduces a database, called the Mapping System, to store and retrieve mappings between identity and location. LISP-capable routers exchange packets over the Internet core by encapsulating them to the appropriate location.

By taking advantage of such separation between location and identity LISP offers Traffic Engineering, multihoming, and mobility among others benefits. Additionally, LISP's approach to solve the routing scalability problem [RFC4984] is that with LISP the Internet core is populated with RLOCs while Traffic Engineering mechanisms are pushed to the Mapping System. With this RLOCs are quasi-static (i.e., low churn) and hence, the routing system scalable [Quoitin] while EIDs can roam anywhere with no churn to the underlying routing system.

This document describes the LISP architecture, its main operational mechanisms as its design rationale. It is important to note that this document does not specify or complement the LISP protocol. The interested reader should refer to the main LISP specifications [RFC6830] and the complementary documents [RFC6831], [RFC6832], [RFC6833], [RFC6834], [RFC6835], [RFC6836], [RFC7052] for the protocol specifications along with the LISP deployment guidelines [RFC7215].

2. Definition of Terms

This document describes the LISP architecture and does not define or introduce any new term. The reader is referred to [RFC6830], [RFC6831], [RFC6832], [RFC6833], [RFC6834], [RFC6835], [RFC6836], [RFC7052], [RFC7215] for the LISP definition of terms.

3. LISP Architecture

This section presents the LISP architecture, it first details the design principles of LISP and then it proceeds to describe its main aspects: data-plane, control-plane, and inetworking mechanisms.

3.1. Design Principles

The LISP architecture is built on top of four basic design principles:

- o Locator/Identifier split: By decoupling the overloaded semantics of the current IP addresses the Internet core can be assigned identity meaningful addresses and hence, can use aggregation to scale. Devices are assigned with relatively opaque identity meaningful addresses that are independent of their topological location.
- o Overlay architecture: Overlays route packets over the current Internet, allowing deployment of new protocols without changing the current infrastructure hence, resulting into a low deployment cost.
- o Decoupled data and control-plane: Separating the data-plane from the control-plane allows them to scale independently and use different architectural approaches. This is important given that they typically have different requirements and allows for other data-planes to be added.
- o Incremental deployability: This principle ensures that the protocol interoperates with the legacy Internet while providing some of the targeted benefits to early adopters.

3.2. Overview of the Architecture

LISP splits architecturally the core from the edge of the Internet by creating two separate namespaces: Endpoint Identifiers (EIDs) and Routing LOCators (RLOCs). The edge consists of LISP sites (e.g., an Autonomous System) that use EID addresses. EIDs are typically -but not limited to- IPv4 or IPv6 addresses that uniquely identify communication end-hosts and are assigned and configured by the same

mechanisms that exist at the time of this writing. EIDs do not contain inter-domain topological information and can be thought as an analogy to Provider Independent (PI [RFC4116]) addresses. Because of this, EIDs are usually only routable at the edge with a LISP site.

With LISP, LISP sites (edge) and the core of the Internet are interconnected by means of LISP-capable routers (e.g., border routers) using tunnels. When packets originated from a LISP site are flowing towards the core network, they ingress into an encapsulated tunnel via an Ingress Tunnel Router (ITR). When packets flow from the core network to a LISP site, they egress from an encapsulated tunnel to an Egress Tunnel Router (ETR). An xTR is a router which can perform both ITR and ETR operations. In this context ITRs encapsulate packets while ETRs decapsulate them, hence LISP operates as an overlay on top of the current Internet core.

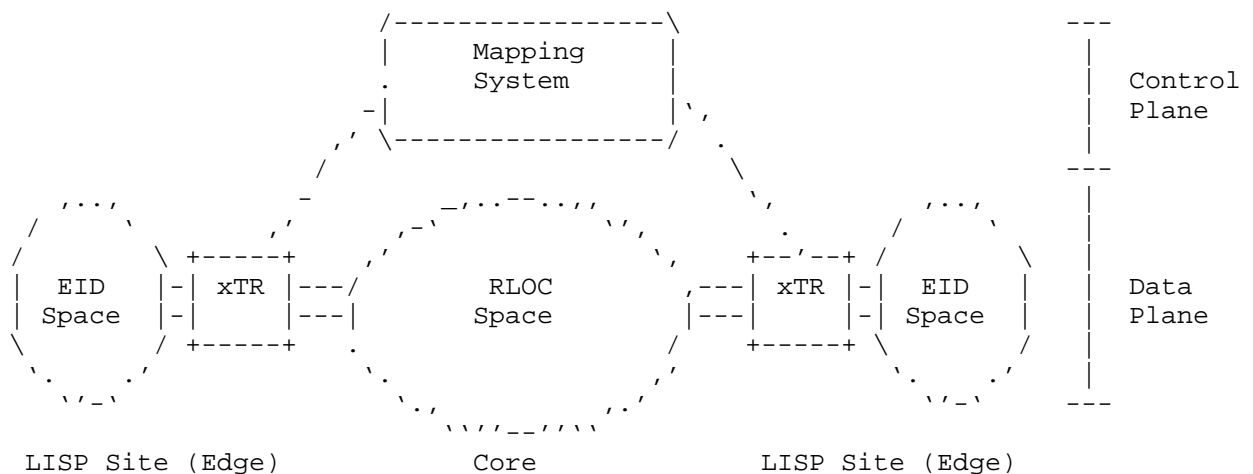


Figure 1.- A schema of the LISP Architecture

With LISP, the core uses RLOCs, an RLOC is typically -but not limited to- an IPv4 or IPv6 address assigned to an Internet-facing network interface of an ITR or ETR. Typically RLOCs are numbered from topologically aggregatable blocks assigned to a site at each point to which it attaches to the global Internet. The topology is defined by the connectivity of networks, in this context RLOCs can be thought of Provider Aggregatable addresses [RFC4116].

A typically distributed database, called the Mapping System, stores mappings between EIDs and RLOCs. Such mappings relate the identity of the devices attached to LISP sites (EIDs) to the set of RLOCs configured at the LISP-capable routers servicing the site. Furthermore, the mappings also include traffic engineering policies and can be configured to achieve multihoming and load balancing. The LISP Mapping System is conceptually similar to the DNS where it is organized as a distributed multi-organization network database. With LISP, ETRs register mappings while ITRs retrieve them.

Finally, the LISP architecture emphasizes a cost effective incremental deployment. Given that LISP represents an overlay to the current Internet architecture, endhosts as well as intra and inter-domain routers remain unchanged, and the only required changes to the existing infrastructure are to routers connecting the EID with the RLOC space. Such LISP capable routers, in most cases, only require a software upgrade. Additionally, LISP requires the deployment of an independent Mapping System, such distributed database is a new network entity.

The following describes a simplified packet flow sequence between two nodes that are attached to LISP sites. Client HostA wants to send a packet to server HostB.

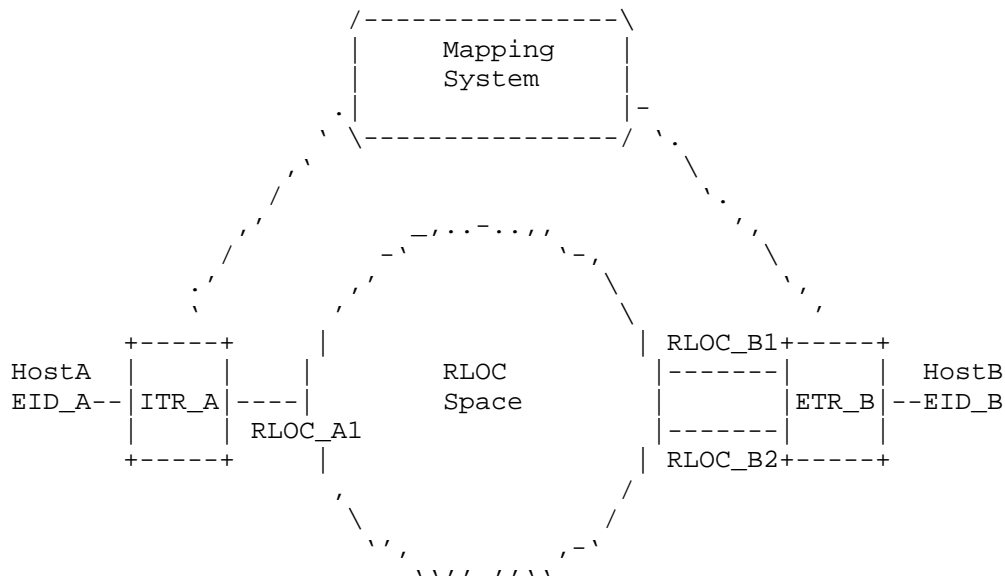


Figure 2.- Packet flow sequence in LISP

1. HostA retrieves the EID_B of HostB (typically querying the DNS) and generates an IP packet as in the Internet, the packet has source address EID_A and destination address EID_B.
2. The packet is routed towards ITR_A in the LISP site using standard intra-domain mechanisms.
3. ITR_A upon receiving the packet queries the Mapping System to retrieve the locator of ETR_B that is servicing HostB's EID_B. In order to do so it uses a LISP control message called Map-Request, the message contains EID_B as the lookup key. In turn it receives another LISP control message called Map-Reply, the message contains two locators: RLOC_B1 and RLOC_B2 along with traffic engineering policies: priority and weight per locator. ITR_A also stores the mapping in a local cache to speed-up forwarding of subsequent packets.
4. ITR_A encapsulates the packet towards RLOC_B1 (chosen according to the priorities/weights specified in the mapping). The packet contains two IP headers, the outer header has RLOC_A1 as source and RLOC_B2 as destination, the inner original header has EID_A as source and EID_B as destination. Furthermore ITR_A adds a LISP header, more details about LISP encapsulation can be found in Section 3.3.1.
5. The encapsulated packet is forwarded by the Internet core as a normal IP packet, making the EID invisible from the Internet core.
6. Upon reception of the encapsulated packet by ETR_B, it decapsulates the packet and forwards it to HostB.

3.3. Data-Plane

This section provides a high-level description of the LISP data-plane, which is specified in detail in [RFC6830]. The LISP data-plane is responsible for encapsulating and decapsulating data packets and caching the appropriate forwarding state. It includes two main entities, the ITR and the ETR, both are LISP capable routers that connect the EID with the RLOC space (ITR) and vice versa (ETR).

3.3.1. LISP Encapsulation

ITRs encapsulate data packets towards ETRs. LISP data packets are encapsulated using UDP (port 4341). A particularity of LISP is that UDP packets should include a zero checksum [RFC6935] [RFC6936] that it is not verified in reception, LISP also supports non-zero checksums that may be verified. This decision was made because the

typical transport protocols used by the applications already include a checksum, by neglecting the additional UDP encapsulation checksum xTRs can forward packets more efficiently.

LISP-encapsulated packets also include a LISP header (after the UDP header and before the original IP header). The LISP header is prepended by ITRs and striped by ETRs. It carries reachability information (see more details in Section 4.2) and the Instance ID field. The Instance ID field is used to distinguish traffic to/from different tenant address spaces at the LISP site and that may use overlapped but logically separated EID addressing.

Overall, LISP works on 4 headers, the inner header the source constructed, and the 3 headers a LISP encapsulator prepends ("outer" to "inner"):

1. Outer IP header containing RLOCs as source and destination addresses. This header is originated by ITRs and stripped by ETRs.
2. UDP header (port 4341) with zero checksum. This header is originated by ITRs and stripped by ETRs.
3. LISP header that contains various forwarding-plane features (such as reachability) and an Instance ID field. This header is originated by ITRs and stripped by ETRs.
4. Inner IP header containing EIDs as source and destination addresses. This header is created by the source end-host and is left unchanged by LISP data plane processing on the ITR and ETR.

Finally, in some scenarios Recursive and/or Re-encapsulating tunnels can be used for Traffic Engineering and re-routing. Re-encapsulating tunnels are consecutive LISP tunnels and occur when a decapsulator (an ETR action) removes a LISP header and then acts as an encapsulator (an ITR action) to prepend another one. On the other hand, Recursive tunnels are nested tunnels and are implemented by using multiple LISP encapsulations on a packet. Typically such functions are implemented by Reencapsulating Tunnel Routers (RTRs).

3.3.2. LISP Forwarding State

ITRs retrieve from the LISP Mapping System mappings between EID prefixes and RLOCs that are used to encapsulate packets. Such mappings are stored in a local cache called the Map-Cache for subsequent packets addressed to the same EID prefix. Mappings include a (Time-to-Live) TTL (set by the ETR). More details about the Map-Cache management can be found in Section 4.1.

3.4. Control-Plane

The LISP control-plane, specified in [RFC6833], provides a standard interface to register and request mappings. The LISP Mapping System is a database that stores such mappings. The following first describes the mappings, then the standard interface to the Mapping System, and finally its architecture.

3.4.1. LISP Mappings

Each mapping includes the bindings between EID prefix(es) and set of RLOCs as well as traffic engineering policies, in the form of priorities and weights for the RLOCs. Priorities allow the ETR to configure active/backup policies while weights are used to load-balance traffic among the RLOCs (on a per-flow basis).

Typical mappings in LISP bind EIDs in the form of IP prefixes with a set of RLOCs, also in the form of IPs. IPv4 and IPv6 addresses are encoded using the appropriate Address Family Identifier (AFI) [RFC3232]. However LISP can also support more general address encoding by means of the ongoing effort around the LISP Canonical Address Format (LCAF) [I-D.ietf-lisp-lcaf].

With such a general syntax for address encoding in place, LISP aims to provide flexibility to current and future applications. For instance LCAFs could support MAC addresses, geo-coordinates, ASCII names and application specific data.

3.4.2. Mapping System Interface

LISP defines a standard interface between data and control planes. The interface is specified in [RFC6833] and defines two entities:

Map-Server: A network infrastructure component that learns mappings from ETRs and publishes them into the LISP Mapping System. Typically Map-Servers are not authoritative to reply to queries and hence, they forward them to the ETR. However they can also operate in proxy-mode, where the ETRs delegate replying to queries to Map-Servers. This setup is useful when the ETR has limited resources (i.e., CPU or power).

Map-Resolver: A network infrastructure component that interfaces ITRs with the Mapping System by proxying queries and in some cases responses.

The interface defines four LISP control messages which are sent as UDP datagrams (port 4342):

Map-Register: This message is used by ETRs to register mappings in the Mapping System and it is authenticated using a shared key between the ETR and the Map-Server.

Map-Notify: When requested by the ETR, this message is sent by the Map-Server in response to a Map-Register to acknowledge the correct reception of the mapping and convey the latest Map-Server state on the EID to RLOC mapping. In some cases a Map-Notify can be sent to the previous RLOCs when an EID is registered by a new set of RLOCs.

Map-Request: This message is used by ITRs or Map-Resolvers to resolve the mapping of a given EID.

Map-Reply: This message is sent by Map-Servers or ETRs in response to a Map-Request and contains the resolved mapping. Please note that a Map-Reply may contain a negative reply if, for example, the queried EID is not part of the LISP EID space. In such cases the ITR typically forwards the traffic natively (non encapsulated) to the public Internet, this behavior is defined to support incremental deployment of LISP.

3.4.3. Mapping System

LISP architecturally decouples control and data-plane by means of a standard interface. This interface glues the data-plane, routers responsible for forwarding data-packets, with the LISP Mapping System, a database responsible for storing mappings.

With this separation in place the data and control-plane can use different architectures if needed and scale independently. Typically the data-plane is optimized to route packets according to hierarchical IP addresses. However the control-plane may have different requirements, for instance and by taking advantage of the LCAFs, the Mapping System may be used to store non-hierarchical keys (such as MAC addresses), requiring different architectural approaches for scalability. Another important difference between the LISP control and data-planes is that, and as a result of the local mapping cache available at ITR, the Mapping System does not need to operate at line-rate.

The LISP WG has explored application of the following distributed system techniques to the Mapping System architecture: graph-based databases in the form of LISP+ALT [RFC6836], hierarchical databases in the form of LISP-DDT [I-D.ietf-lisp-ddt], monolithic databases in the form of LISP-NERD [RFC6837], flat databases in the form of LISP-DHT [I-D.cheng-lisp-shdht],[I-D.mathy-lisp-dht] and, a multicast-based database [I-D.curran-lisp-emacs]. Furthermore it is worth

noting that, in some scenarios such as private deployments, the Mapping System can operate as logically centralized. In such cases it is typically composed of a single Map-Server/Map-Resolver.

The following focuses on the two mapping systems that have been implemented and deployed (LISP-ALT and LISP+DDT).

3.4.3.1. LISP+ALT

The LISP Alternative Topology (LISP+ALT) [RFC6836] was the first Mapping System proposed, developed and deployed on the LISP pilot network. It is based on a distributed BGP overlay participated by Map-Servers and Map-Resolvers. The nodes connect to their peers through static tunnels. Each Map-Server involved in the ALT topology advertises the EID-prefixes registered by the serviced ETRs, making the EID routable on the ALT topology.

When an ITR needs a mapping it sends a Map-Request to a Map-Resolver that, using the ALT topology, forwards the Map-Request towards the Map-Server responsible for the mapping. Upon reception the Map-Server forwards the request to the ETR that in turn, replies directly to the ITR using the native Internet core.

3.4.3.2. LISP-DDT

LISP-DDT [I-D.ietf-lisp-ddt] is conceptually similar to the DNS, a hierarchical directory whose internal structure mirrors the hierarchical nature of the EID address space. The DDT hierarchy is composed of DDT nodes forming a tree structure, the leafs of the tree are Map-Servers. On top of the structure there is the DDT root node [DDT-ROOT], which is a particular instance of a DDT node and that matches the entire address space. As in the case of DNS, DDT supports multiple redundant DDT nodes and/or DDT roots. Finally, Map-Resolvers are the clients of the DDT hierarchy and can query either the DDT root and/or other DDT nodes.

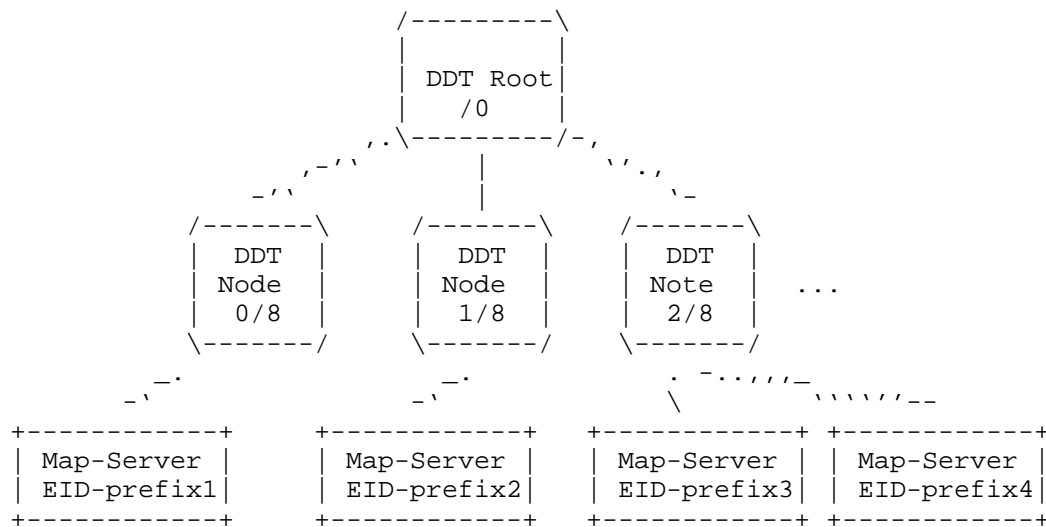


Figure 3.- A schematic representation of the DDT tree structure, please note that the prefixes and the structure depicted should be only considered as an example.

The DDT structure does not actually index EID-prefixes but eXtended EID-prefixes (XEID). An XEID-prefix is just the concatenation of the following fields (from most significant bit to less significant bit): Database-ID, Instance ID, Address Family Identifier and the actual EID-prefix. The Database-ID is provided for possible future requirements of higher levels in the hierarchy and to enable the creation of multiple and separate database trees.

In order to resolve a query LISP-DDT operates in a similar way to the DNS but only supports iterative lookups. DDT clients (usually Map-Resolvers) generate Map-Requests to the DDT root node. In response they receive a newly introduced LISP-control message: a Map-Referral. A Map-Referral provides the list of RLOCs of the set of DDT nodes matching a configured XEID delegation. That is, the information contained in the Map-Referral points to the child of the queried DDT node that has more specific information about the queried XEID-prefix. This process is repeated until the DDT client walks the tree structure (downwards) and discovers the Map-Server servicing the queried XEID. At this point the client sends a Map-Request and receives a Map-Reply containing the mappings. It is important to note that DDT clients can also cache the information contained in Map-Referrals, that is, they cache the DDT structure. This is used to reduce the mapping retrieving latency[Jakab].

The DDT Mapping System relies on manual configuration. That is Map-Resolvers are manually configured with the set of available DDT root nodes while DDT nodes are manually configured with the appropriate XEID delegations. Configuration changes in the DDT nodes are only required when the tree structure changes itself, but it doesn't depend on EID dynamics (RLOC allocation or traffic engineering policy changes).

3.5. Interworking Mechanisms

EIDs are typically identical to either IPv4 or IPv6 addresses and they are stored in the LISP Mapping System, however they are usually not announced in the Internet global routing system. As a result LISP requires an inetworking mechanism to allow LISP sites to speak with non-LISP sites and vice versa. LISP inetworking mechanisms are specified in [RFC6832].

LISP defines two entities to provide inetworking:

Proxy Ingress Tunnel Router (PITR): PITRs provide connectivity from the legacy Internet to LISP sites. PITRs announce in the global routing system blocks of EID prefixes (aggregating when possible) to attract traffic. For each incoming packet from a source not in a LISP site (a non-EID), the PITR LISP-encapsulates it towards the RLOC(s) of the appropriate LISP site. The impact of PITRs in the routing table size of the DFZ is, in the worst-case, similar to the case in which LISP is not deployed. EID-prefixes will be aggregated as much as possible both by the PITR and by the global routing system.

Proxy Egress Tunnel Router (PETR): PETRs provide connectivity from LISP sites to the legacy Internet. In some scenarios, LISP sites may be unable to send encapsulated packets with a local EID address as a source to the legacy Internet. For instance when Unicast Reverse Path Forwarding (uRPF) is used by Provider Edge routers, or when an intermediate network between a LISP site and a non-LISP site does not support the desired version of IP (IPv4 or IPv6). In both cases the PETR overcomes such limitations by encapsulating packets over the network. There is no specified provision for the distribution of PETR RLOC addresses to the ITRs.

4. LISP Operational Mechanisms

This section details the main operational mechanisms defined in LISP.

4.1. Cache Management

LISP's decoupled control and data-plane, where mappings are stored in the control-plane and used for forwarding in the data plane, requires of a local cache in ITRs to reduce signaling overhead (Map-Request/Map-Reply) and increase forwarding speed. The local cache available at the ITRs, called Map-Cache, is used by the router to LISP-encapsulate packets. The Map-Cache is indexed by (Instance ID, EID-prefix) and contains basically the set of RLOCs with the associated traffic engineering policies (priorities and weights).

The Map-Cache, as any other cache, requires cache coherence mechanisms to maintain up-to-date information. LISP defines three main mechanisms for cache coherence:

Time-To-Live (TTL): Each mapping contains a TTL set by the ETR, upon expiration of the TTL the ITR has to refresh the mapping by sending a new Map-Request. Typical values for TTL defined by LISP are 24 hours.

Solicit-Map-Request (SMR): SMR is an explicit mechanism to update mapping information. In particular a special type of Map-Request can be sent on demand by ETRs to request refreshing a mapping. Upon reception of a SMR message, the ITR must refresh the bindings by sending a Map-Request to the Mapping System.

Map-Versioning: This optional mechanism piggybacks in the LISP header of data-packets the version number of the mappings used by an xTR. This way, when an xTR receives a LISP-encapsulated packet from a remote xTR, it can check whether its own Map-Cache or the one of the remote xTR is outdated. If its Map-Cache is outdated, it sends a Map-Request for the remote EID so to obtain the newest mappings. On the contrary, if it detects that the remote xTR Map-Cache is outdated, it sends a SMR to notify it that a new mapping is available.

Finally it is worth noting that in some cases an entry in the map-cache can be proactively refreshed using the mechanisms described in the section below.

4.2. RLOC Reachability

The LISP architecture is an edge to edge pull architecture, where the network state is stored in the control-plane while the data-plane pulls it on demand. On the contrary BGP is a push architecture, where the required network state is pushed by means of BGP UPDATE messages to BGP speakers. In push architectures, reachability information is also pushed to the interested routers. However pull

architectures require explicit mechanisms to propagate reachability information. LISP defines a set of mechanisms to inform ITRs and PITRS about the reachability of the cached RLOCs:

Locator Status Bits (LSB): LSB is a passive technique, the LSB field is carried by data-packets in the LISP header and can be set by a ETRs to specify which RLOCs of the ETR site are up/down. This information can be used by the ITRs as a hint about the reachability to perform additional checks. Also note that LSB does not provide path reachability status, only hints on the status of RLOCs.

Echo-nonce: This is also a passive technique, that can only operate effectively when data flows bi-directionally between two communicating xTRs. Basically, an ITR piggybacks a random number (called nonce) in LISP data packets, if the path and the probed locator are up, the ETR will piggyback the same random number on the next data-packet, if this is not the case the ITR can set the locator as unreachable. When traffic flow is unidirectional or when the ETR receiving the traffic is not the same as the ITR that transmits it back, additional mechanisms are required.

RLOC-probing: This is an active probing algorithm where ITRs send probes to specific locators, this effectively probes both the locator and the path. In particular this is done by sending a Map-Request (with certain flags activated) on the data-plane (RLOC space) and waiting in return a Map-Reply, also sent on the data-plane. The active nature of RLOC-probing provides an effective mechanism to determine reachability and, in case of failure, switching to a different locator. Furthermore the mechanism also provides useful RTT estimates of the delay of the path that can be used by other network algorithms.

Additionally, LISP also recommends inferring reachability of locators by using information provided by the underlay, in particular:

It is worth noting that RLOC probing and Echo-nonce can work together. Specifically if a nonce is not echoed, an ITR could RLOC-probe to determine if the path is up when it cannot tell the difference between a failed bidirectional path or the return path is not used (a unidirectional path).

ICMP signaling: The LISP underlay -the current Internet- uses the ICMP protocol to signal unreachability (among other things). LISP can take advantage of this and the reception of a ICMP Network Unreachable or ICMP Host Unreachable message can be seen as a hint that a locator might be unreachable, this should lead to perform additional checks.

Underlay routing: Both BGP and IBGP carry reachability information, LISP-capable routers that have access to underlay routing information can use it to determine if a given locator or path are reachable.

4.3. ETR Synchronization

All the ETRs that are authoritative to a particular EID-prefix must announce the same mapping to the requesters, this means that ETRs must be aware of the status of the RLOCs of the remaining ETRs. This is known as ETR synchronization.

At the time of this writing LISP does not specify a mechanism to achieve ETR synchronization. Although many well-known techniques could be applied to solve this issue it is still under research, as a result operators must rely on coherent manual configuration

4.4. MTU Handling

Since LISP encapsulates packets it requires dealing with packets that exceed the MTU of the path between the ITR and the ETR. Specifically LISP defines two mechanisms:

Stateless: With this mechanism the effective MTU is assumed from the ITR's perspective. If a payload packet is too big for the effective MTU, and can be fragmented, the payload packet is fragmented on the ITR, such that reassembly is performed at the destination host.

Stateful: With this mechanism ITRs keep track of the MTU of the paths towards the destination locators by parsing the ICMP Too Big packets sent by intermediate routers. Additionally ITRs will send ICMP Too Big messages to inform the sources about the effective MTU.

In both cases if the packet cannot be fragmented (IPv4 with DF=1 or IPv6) then the ITR drops it and replies with a ICMP Too Big message to the source.

5. Mobility

The separation between locators and identifiers in LISP was initially proposed for traffic engineering purpose where LISP sites can change their attachment points to the Internet (i.e., RLOCs) without impacting endpoints or the Internet core. In this context, the border routers operate the xTR functionality and endpoints are not aware of the existence of LISP. However, this mode of operation does not allow seamless mobility of endpoints between different LISP sites as the EID address might not be routable in a visited site.

Nevertheless, LISP can be used to enable seamless IP mobility when LISP is directly implemented in the endpoint or when the endpoint roams to an attached xTR. Each endpoint is then an xTR and the EID address is the one presented to the network stack used by applications while the RLOC is the address gathered from the network when it is visited.

Whenever the device changes of RLOC, the xTR updates the RLOC of its local mapping and registers it to its Map-Server. To avoid the need of a home gateway, the ITR also indicates the RLOC change to all remote devices that have ongoing communications with the device that moved. The combination of both methods ensures the scalability of the system as signaling is strictly limited the Map-Server and to hosts with which communications are ongoing.

6. Multicast

LISP also supports transporting IP multicast packets sent from the EID space, the operational changes required to the multicast protocols are documented in [RFC6831].

In such scenarios, LISP may create multicast state both at the core and at the sites (both source and receiver). When signaling is used to create multicast state at the sites, LISP routers unicast encapsulate PIM Join/Prune messages from receiver to source sites. At the core, ETRs build a new PIM Join/Prune message addressed to the RLOC of the ITR servicing the source. An simplified sequence is shown below

1. An end-host willing to join a multicast channel sends an IGMP report. Multicast PIM routers at the LISP site propagate PIM Join/Prune messages (S-EID, G) towards the ETR.
2. The join message flows to the ETR, upon reception the ETR builds two join messages, the first one unicast LISP-encapsulates the original join message towards the RLOC of the ITR servicing the source. This message creates (S-EID, G) multicast state at the source site. The second join message contains as destination address the RLOC of the ITR servicing the source (S-RLOC, G) and creates multicast state at the core.
3. Multicast data packets originated by the source (S-EID, G) flow from the source to the ITR. The ITR LISP-encapsulates the multicast packets, the outer header includes its own RLOC as the source (S-RLOC) and the original multicast group address (G) as the destination. Please note that multicast group address are logical and are not resolved by the mapping system. Then the multicast packet is transmitted through the core towards the

receiving ETRs that decapsulates the packets and sends them using the receiver's site multicast state.

LISP can also support non-PIM mechanisms to maintain multicast state.

7. Security

LISP uses a pull architecture to learn mappings. While in a push system, the state necessary to forward packets is learned independently of the traffic itself, with a pull architecture, the system becomes reactive and data-plane events (e.g., the arrival of a packet for an unknown destination) may trigger control-plane events. This on-demand learning of mappings provides many advantages as discussed above but may also affect the way security is enforced.

Usually, the data-plane is implemented in the fast path of routers to provide high performance forwarding capabilities while the control-plane features are implemented in the slow path to offer high flexibility and a performance gap of several order of magnitude can be observed between the slow and the fast paths. As a consequence, the way data-plane events are notified to the control-plane must be thought carefully so to not overload the slow path and rate limiting should be used as specified in [RFC6830].

Care must also be taken so to not overload the mapping system (i.e., the control plane infrastructure) as the operations to be performed by the mapping system may be more complex than those on the data-plane, for that reason [RFC6830] recommends to rate limit the sending of messages to the mapping system.

To improve resiliency and reduce the overall number of messages exchanged, LISP offers the possibility to leak information, such as reachability of locators, directly into data plane packets. In environments that are not fully trusted, control informations gleaned from data-plane packets should be verified before using them.

Mappings are the centrepiece of LISP and all precautions must be taken to avoid them to be manipulated or misused by malicious entities. Using trustable Map-Servers that strictly respect [RFC6833] and the lightweight authentication mechanism proposed by LISP-Sec [I-D.ietf-lisp-sec] reduces the risk of attacks to the mapping integrity. In more critical environments, secure measures may be needed.

As with any other tunneling mechanism, middleboxes on the path between an ITR (or PITR) and an ETR (or PETR) must implement mechanisms to strip the LISP encapsulation to correctly inspect the content of LISP encapsulated packets.

Like other map-and-encap mechanisms, LISP enables triangular routing (i.e., packets of a flow cross different border routers depending on their direction). This means that intermediate boxes may have incomplete view on the traffic they inspect or manipulate.

More details about security implications of LISP are discussed in [I-D.ietf-lisp-threats].

8. Use Cases

8.1. Traffic Engineering

BGP is the standard protocol to implement inter-domain routing. With BGP, routing informations are propagated along the network and each autonomous system can implement its own routing policy that will influence the way routing information are propagated. The direct consequence is that an autonomous system cannot precisely control the way the traffic will enter the network.

As opposed to BGP, a LISP site can strictly impose via which ETRs the traffic must enter the the LISP site network even though the path followed to reach the ETR is not under the control of the LISP site. This fine control is implemented with the mappings. When a remote site is willing to send traffic to a LISP site, it retrieves the mapping associated to the destination EID via the mapping system. The mapping is sent directly by an authoritative ETR of the EID and is not altered by any intermediate network.

A mapping associates a list of RLOCs to an EID prefix. Each RLOC corresponds to an interface of an ETR (or set of ETRs) that is able to correctly forward packets to EIDs in the prefix. Each RLOC is tagged with a priority and a weight in the mapping. The priority is used to indicates which RLOCs should be preferred to send packets (the least preferred ones being provided for backup purpose). The weight permits to balance the load between the RLOCs with the same priority, proportionally to the weight value.

As mappings are directly issued by the authoritative ETR of the EID and are not altered while transmitted to the remote site, it offers highly flexible incoming inter-domain traffic engineering with even the possibility for a site to issue a different mapping for each remote site, implementing so precise routing policies.

8.2. LISP for IPv6 Co-existence

LISP encapsulations permits to transport packets using EIDs from a given address family (e.g., IPv6) with packets from other address families (e.g., IPv4). The absence of correlation between the

address family of RLOCs and EIDs makes LISP a candidate to allow, e.g., IPv6 to be deployed when all of the core network may not have IPv6 enabled.

For example, two IPv6-only data centers could be interconnected via the legacy IPv4 Internet. If their border routers are LISP capable, sending packets between the data center is done without any form of translation as the native IPv6 packets (in the EID space) will be LISP encapsulated and transmitted over the IPv4 legacy Internet by the mean of IPv4 RLOCs.

8.3. LISP for Virtual Private Networks

It is common to operate several virtual networks over the same physical infrastructure. In such virtual private networks, it is essential to distinguish which virtual network a packet belongs and tags or labels are used for that purpose. With LISP, the distinction can be made with the Instance ID field. When an ITR encapsulates a packet from a particular virtual network (e.g., known via the VRF or VLAN), it tags the encapsulated packet with the Instance ID corresponding to the virtual network of the packet. When an ETR receives a packet tagged with an Instance ID it uses the Instance ID to determine how to treat the packet.

The main advantage of using LISP for virtual networks, on top of the simplicity of managing the mappings, is that it does not impose any requirement on the underlying network, as long as it is running IP.

8.4. LISP for Virtual Machine Mobility in Data Centers

A way to enable seamless virtual machine mobility in data center is to conceive the datacenter backbone as the RLOC space and the subnet where servers are hosted as forming the EID space. A LISP router is placed at the border between the backbone and each subnet. When a virtual machine is moved to another subnet, it can keep (temporarily) the address it had before the move so to continue without a transport layer connection reset. When an xTR detects a source address received on a subnet to be an address not assigned to the subnet, it registers the address to the Mapping System.

To inform the other LISP routers that the machine moved and where, and then to avoid detours via the initial subnetwork, mechanisms such as the Solicit-Map-Request messages are used.

9. Security Considerations

This document does not specify any protocol or operational practices and hence, does not have any security considerations.

10. IANA Considerations

This memo includes no request to IANA.

11. Acknowledgements

This document was initiated by Noel Chiappa and much of the core philosophy came from him. The authors acknowledge the important contributions he has made to this work and thank him for his past efforts.

The authors would also like to thank Dino Farinacci, Fabio Maino, Luigi Iannone, Sharon Barakai, Isidoros Kouvelas, Christian Cassar, Florin Coras, Marc Binderberger, Alberto Rodriguez-Natal, Ronald Bonica, Chad Hintz, Robert Raszuk, Joel M. Halpern, Darrel Lewis, as well as every people acknowledged in [RFC6830].

12. References

12.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC3232] Reynolds, J., "Assigned Numbers: RFC 1700 is Replaced by an On-line Database", RFC 3232, January 2002.
- [RFC4116] Abley, J., Lindqvist, K., Davies, E., Black, B., and V. Gill, "IPv4 Multihoming Practices and Limitations", RFC 4116, July 2005.
- [RFC4984] Meyer, D., Zhang, L., and K. Fall, "Report from the IAB Workshop on Routing and Addressing", RFC 4984, September 2007.
- [RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, January 2013.
- [RFC6831] Farinacci, D., Meyer, D., Zwiebel, J., and S. Venaas, "The Locator/ID Separation Protocol (LISP) for Multicast Environments", RFC 6831, January 2013.

- [RFC6832] Lewis, D., Meyer, D., Farinacci, D., and V. Fuller, "Interworking between Locator/ID Separation Protocol (LISP) and Non-LISP Sites", RFC 6832, January 2013.
- [RFC6833] Fuller, V. and D. Farinacci, "Locator/ID Separation Protocol (LISP) Map-Server Interface", RFC 6833, January 2013.
- [RFC6834] Iannone, L., Saucez, D., and O. Bonaventure, "Locator/ID Separation Protocol (LISP) Map-Versioning", RFC 6834, January 2013.
- [RFC6835] Farinacci, D. and D. Meyer, "The Locator/ID Separation Protocol Internet Groper (LIG)", RFC 6835, January 2013.
- [RFC6836] Fuller, V., Farinacci, D., Meyer, D., and D. Lewis, "Locator/ID Separation Protocol Alternative Logical Topology (LISP+ALT)", RFC 6836, January 2013.
- [RFC6837] Lear, E., "NERD: A Not-so-novel Endpoint ID (EID) to Routing Locator (RLOC) Database", RFC 6837, January 2013.
- [RFC6935] Eubanks, M., Chimento, P., and M. Westerlund, "IPv6 and UDP Checksums for Tunneled Packets", RFC 6935, April 2013.
- [RFC6936] Fairhurst, G. and M. Westerlund, "Applicability Statement for the Use of IPv6 UDP Datagrams with Zero Checksums", RFC 6936, April 2013.
- [RFC7052] Schudel, G., Jain, A., and V. Moreno, "Locator/ID Separation Protocol (LISP) MIB", RFC 7052, October 2013.
- [RFC7215] Jakab, L., Cabellos-Aparicio, A., Coras, F., Domingo-Pascual, J., and D. Lewis, "Locator/Identifier Separation Protocol (LISP) Network Element Deployment Considerations", RFC 7215, April 2014.

12.2. Informative References

- [Chiappa] Chiappa, J., "Endpoints and Endpoint names: A Propose Enhancement to the Internet Architecture, <http://mercury.lcs.mit.edu/~jnc/tech/endpoints.txt>", 1999.
- [DDT-ROOT] LISP DDT ROOT, , "<http://ddt-root.org/>", August 2013.
- [DFZ] Huston, Geoff., "Growth of the BGP Table - 1994 to Present <http://bgp.potaroo.net/>", August 2013.

- [I-D.cheng-lisp-shdht]
Cheng, L. and J. Wang, "LISP Single-Hop DHT Mapping Overlay", draft-cheng-lisp-shdht-04 (work in progress), July 2013.
- [I-D.curran-lisp-emacs]
Brim, S., Farinacci, D., Meyer, D., and J. Curran, "EID Mappings Multicast Across Cooperating Systems for LISP", draft-curran-lisp-emacs-00 (work in progress), November 2007.
- [I-D.ietf-lisp-ddt]
Fuller, V., Lewis, D., Ermagan, V., and A. Jain, "LISP Delegated Database Tree", draft-ietf-lisp-ddt-02 (work in progress), October 2014.
- [I-D.ietf-lisp-lcaf]
Farinacci, D., Meyer, D., and J. Snijders, "LISP Canonical Address Format (LCAF)", draft-ietf-lisp-lcaf-06 (work in progress), October 2014.
- [I-D.ietf-lisp-sec]
Maino, F., Ermagan, V., Cabellos-Aparicio, A., and D. Saucez, "LISP-Security (LISP-SEC)", draft-ietf-lisp-sec-07 (work in progress), October 2014.
- [I-D.ietf-lisp-threats]
Saucez, D., Iannone, L., and O. Bonaventure, "LISP Threats Analysis", draft-ietf-lisp-threats-10 (work in progress), July 2014.
- [I-D.mathy-lisp-dht]
Mathy, L., Iannone, L., and O. Bonaventure, "LISP-DHT: Towards a DHT to map identifiers onto locators" draft-mathy-lisp-dht-00 (work in progress)", April 2008.
- [Jakab]
Jakab, L., Cabellos, A., Saucez, D., and O. Bonaventure, "LISP-TREE: A DNS Hierarchy to Support the LISP Mapping System, IEEE Journal on Selected Areas in Communications, vol. 28, no. 8, pp. 1332-1343", October 2010.
- [Quoitin]
Quoitin, B., Iannone, L., Launois, C., and O. Bonaventure, "Evaluating the Benefits of the Locator/Identifier Separation" in Proceedings of 2Nd ACM/IEEE International Workshop on Mobility in the Evolving Internet Architecture", 2007.

Appendix A. A Brief History of Location/Identity Separation

The LISP system for separation of location and identity resulted from the discussions of this topic at the Amsterdam IAB Routing and Addressing Workshop, which took place in October 2006 [RFC4984].

A small group of like-minded personnel from various scattered locations within Cisco, spontaneously formed immediately after that workshop, to work on an idea that came out of informal discussions at the workshop and on various mailing lists. The first Internet-Draft on LISP appeared in January, 2007.

Trial implementations started at that time, with initial trial deployments underway since June 2007; the results of early experience have been fed back into the design in a continuous, ongoing process over several years. LISP at this point represents a moderately mature system, having undergone a long organic series of changes and updates.

LISP transitioned from an IRTF activity to an IETF WG in March 2009, and after numerous revisions, the basic specifications moved to becoming RFCs at the start of 2013 (although work to expand and improve it, and find new uses for it, continues, and undoubtedly will for a long time to come).

A.1. Old LISP Models

LISP, as initially conceived, had a number of potential operating modes, named 'models'. Although they are not used anymore, one occasionally sees mention of them, so they are briefly described here.

LISP 1: EIDs all appear in the normal routing and forwarding tables of the network (i.e. they are 'routable'); this property is used to 'bootstrap' operation, by using this to load EID->RLOC mappings. Packets were sent with the EID as the destination in the outer wrapper; when an ETR saw such a packet, it would send a Map-Reply to the source ITR, giving the full mapping.

LISP 1.5: Similar to LISP 1, but the routability of EIDs happens on a separate network.

LISP 2: EIDs are not routable; EID->RLOC mappings are available from the DNS.

LISP 3: EIDs are not routable; and have to be looked up in a new EID->RLOC mapping database (in the initial concept, a system using Distributed Hash Tables). Two variants were possible: a 'push'

system, in which all mappings were distributed to all ITRs, and a 'pull' system in which ITRs load the mappings they need, as needed.

Authors' Addresses

Albert Cabellos
UPC-BarcelonaTech
c/ Jordi Girona 1-3
Barcelona, Catalonia 08034
Spain

Email: acabello@ac.upc.edu

Damien Saucez (Ed.)
INRIA
2004 route des Lucioles BP 93
Sophia Antipolis Cedex 06902
France

Email: damien.saucez@inria.fr

Network Working Group
Internet-Draft
Intended status: Experimental
Expires: February 9, 2015

C. Cassar
I. Kouvelas
D. Lewis
Cisco Systems
August 8, 2014

LISP Reliable Transport
draft-kouvelas-lisp-reliable-transport-01.txt

Abstract

The communication between LISP ETRs and Map-Servers is based on unreliable UDP message exchange coupled with periodic message transmission in order to maintain soft state. The drawback of periodic messaging is the constant load imposed on both the ETR and the Map-Server. New use cases for LISP have increased the amount of state that needs to be communicated with requirements that are not satisfied by the current mechanism. This document introduces the use of a reliable transport for ETR to Map-Server communication in order to eliminate the periodic messaging overhead, while providing reliability, flow-control and endpoint liveness detection.

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on February 9, 2015.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of

publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Requirements Notation	3
3. Message Format	3
4. Session Establishment	5
5. Error Notifications	5
6. EID Prefix Registration	7
6.1. Reliable Mapping Registration Messages	7
6.1.1. Registration Message	7
6.1.2. Registration Acknowledgement Message	8
6.1.3. Registration Rejected Message	9
6.1.4. Registration Refresh Message	9
6.1.5. Mapping Notification Message	10
6.2. ETR Behavior	11
6.3. Map-Server Behavior	15
7. Security Considerations	16
8. IANA Considerations	16
8.1. LISP Reliable Transport Message Types	16
8.2. Transport Protocol Port Numbers	16
9. Acknowledgments	16
10. Normative References	17
Authors' Addresses	17

1. Introduction

The communication channel between LISP ETRs and Map-Servers is based on unreliable UDP message exchange [RFC6833]. Where required, reliability is pursued through periodic retransmissions that maintain soft state on the peer. Map-Register messages are retransmitted every minute by an ETR and the Map-Server times out its state if the state is not refreshed for three successive periods. When registering multiple EID-Prefixes, the ETR includes multiple mapping records in the Map-Register message. Packet size limitations provide an upper bound to the number of mapping records that can be placed in each Map-Register message. When the ETR has more EID-Prefixes to register than can be packed in a single Map-Register message, the mapping records for the EID-Prefixes are split across multiple Map-Register messages.

The drawback of the periodic registration is the constant load that it introduces on both the ETR and the Map-Server. The ETR uses resources to periodically build and transmit the Map-Register messages, and to process the resulting Map-Notify messages issued by the Map-Server. The Map-Server uses resources to process the received Map-Register messages, update the corresponding registration state, and build and transmit the matching Map-Notify messages. When the number of EID-Prefixes to be registered by an ETR is small, the resulting load imposed by periodic registrations may not be significant. The ETR will only transmit a single Map-Register message each period that contains a small number of mapping records.

In some LISP deployments, a large set of EID-Prefixes must be registered by each ETR (e.g. mobility, database redistribution). Use cases with a large set of EID-Prefixes behind an ETR will result in a much higher load. An example is LISP mobility deployments where EID-Prefixes are limited to host entries. ETRs may have thousands of hosts to register resulting in hundreds of Map-Register and Map-Notify messages per registration period.

A transport is required for the ETR to Map-Server communication that provides reliability, flow-control and endpoint liveness notifications. This document describes the use of TCP or SCTP as a LISP reliable transport. The initial application for the LISP reliable transport session is the support of scalable EID prefix registration. The reliable session mechanism is defined to be extensible so that it can support additional LISP communication requirements as they arise using a single reliable transport session between an ETR and a Map-Server. The use of the reliable transport session for EID prefix registration is an alternative and does not replace the existing UDP based mechanism.

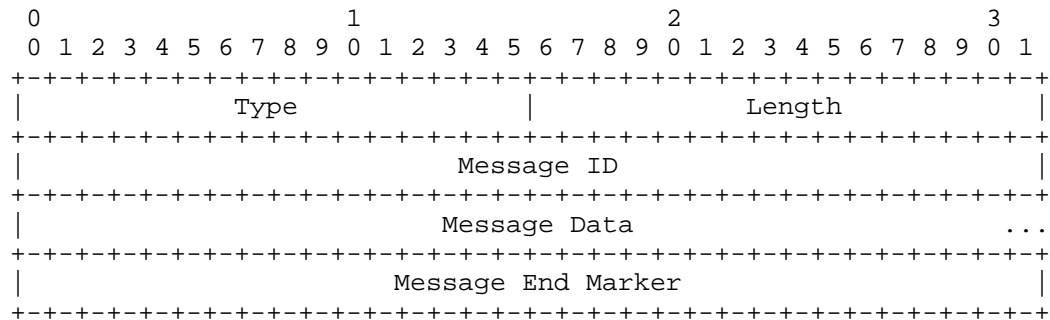
2. Requirements Notation

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

3. Message Format

A single LISP reliable transport session may carry information for multiple LISP applications. One such application is the registration of EID to RLOC mappings that operates over a session between an ETR and a Map-Server. Communication over a session is based on the exchange of messages. This document defines a base set of messages to support session establishment and management. It also defines the messages for the EID to RLOC mapping registration application.

To support protocol extensibility when new applications, or extensions to existing applications are introduced, the messages are based on a TLV format.



Reliable transport message format

- o Type: 16 bit type field identifying the message type.
- o Length: 16 bit field that provides the total size of the message in octets including the length, type and end marker fields. The length allows the receiver to locate the next message in the TCP stream. The minimum value of the length field is 8.
- o ID: A 32-bit value that identifies the message. May be used by the receiver to identify the message in replies or notification messages.
- o Data: Type specific message contents.
- o End Marker: A 32-bit message end marker that must be set to 0x9FACADE9. The End Marker is used by the receiver to validate that it has correctly parsed or skipped a message and provides a method to detect formatting errors. Note that message data may also contain this marker, and that the marker itself is not sufficient for parsing the message.

The base message format does not indicate how the peer should deal with the message in cases where the message type is not supported/understood. This is best dealt with by the application. For example, in case an error notification is returned, or an expected acknowledgement message is not received, the application might choose various courses of action; from simply logging that the feature is not supported, all the way to tearing the relationship with the peer down for the feature, or for all LISP features.

4. Session Establishment

The LISP router that performs the active open initiates the connection from a locally generated source transport port number to the well-known destination transport port assigned to LISP. The LISP router that performs the passive open listens on the well-known local transport port and does not qualify the remote transport port number. In the ETR to Map-Server reliable transport session, the ETR assumes the active role and the Map-Server passively accepts connections.

A single reliable transport session can be established between a pair of LISP peers to cover all communication needs. For example, an ETR that has EID prefix registrations for multiple EID instances and EID address families might only establish a single session with the Map-Server.

When using TCP and symmetric connection establishment LISP must perform collision detection and duplicate session elimination. To accomplish that, LISP peer ID messages will be exchanged between the peers once a session is established. If duplicate sessions are detected then the one that was initiated by the router with the higher ID is kept and the other session is torn down. TBD

5. Error Notifications

The error notification message is used to communicate base reliable transport session communication errors. LISP applications making use of the reliable transport session and having to communicate application specific errors must define their own messages to do so. An error notification is issued when the receiver of a message does not recognize the message type or cannot parse the message contents. The notification includes the offending message type and ID and as much of the offending message data as the notification sender wishes to.



Error notification message format

- o Error Code: An 8 bit field identifying the type of error that occurred. Defined errors are:
 - * Unrecognized message type.
 - * Message format error.
- o Reserved: Set to zero by the sender and ignored by the receiver.
- o Offending Message Type: 16 bit type field identifying the message type of the offending message that triggered this error notification. This is copied from the Type field of the offending message.
- o Offending Message Length: 16 bit field that provides the total size of the offending message in octets. This is copied from the Length field of the offending message.
- o Offending Message ID: A 32-bit field that is set to the Message ID field of the offending message.
- o Offending Message Data: The Data from the offending message that triggered this error notification. The sender of the notification may include as much of the original data as is deemed necessary. The length of the Offending Message Data field is not provided by the Offending Message Length field and is determined by subtracting the size of the other fields in the message from the

Length field. It is valid to not include any of the offending message data when sending an error notification.

- o End Marker: A 32-bit message end marker that must be set to 0x9FACADE9. The End Marker is used by the receiver to validate that it has correctly parsed or skipped a message and provides a method to detect formatting errors. Note that message data may also contain this marker, and that the marker itself is not sufficient for parsing the message.

An error notification cannot be the offending message in another error notification and MUST NOT trigger such a message.

6. EID Prefix Registration

EID prefix registration uses the reliable transport session between an ETR and a Map-Server to communicate the ETR local EID database EID to RLOC mappings to the Map-Server. In contrast to the UDP based periodic registration, mapping information over the reliable transport session is only sent when there is new information available for the Map-Server. The Map-Server does not maintain a timer to expire registrations communicated over the reliable transport session. Instead an explicit de-registration (a registration carrying a zero TTL) is needed to delete the state maintained by the Map-Server.

The key used to identify registration mapping records in the ETR to Map-Server communication is the EID prefix. The prefix may be specified using an LCAF encoding that includes an EID instance ID.

When the reliable transport session goes down, registration mappings learned by the Map-Server are treated as periodic UDP registrations and a timer is used to expire them after 3 minutes. During this period UDP based registrations or the re-establishment of the reliable transport session and subsequent communication of a new mapping can update the EID prefix mapping state.

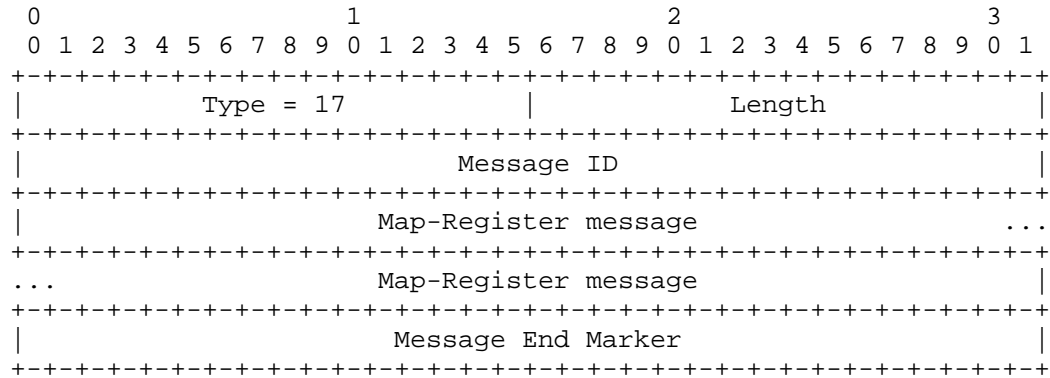
6.1. Reliable Mapping Registration Messages

This section defines the LISP reliable transport session messages used to communicate local EID database registrations between the ETR and the Map-Server.

6.1.1. Registration Message

The reliable transport Registration message is used to communicate EID to RLOC mapping registrations from the ETR to the Map-Server. The Registration message uses exactly the same format as the UDP Map-

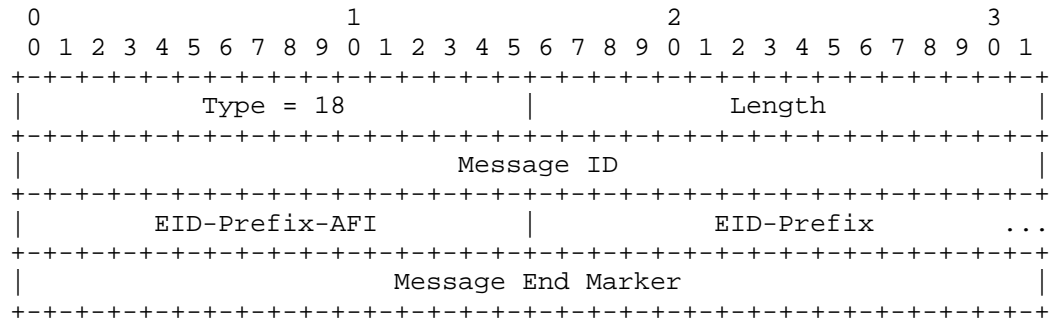
Register message but instead of the IP/UDP header, the Map-Register is placed within the value section of the reliable transport TLV. A common message format is proposed to leverage the authentication features built into the UDP Map-Register message and increase code reuse.



Registration message format

6.1.2. Registration Acknowledgement Message

The Acknowledgement message is sent from the Map-Server to the ETR to confirm successful registration of an EID prefix previously communicated by a reliable transport session Registration message. The Registration Acknowledgement message does not carry a mapping record (the map servers view of the mapping). This is accomplished by the LISP reliable transport Map Notification message.



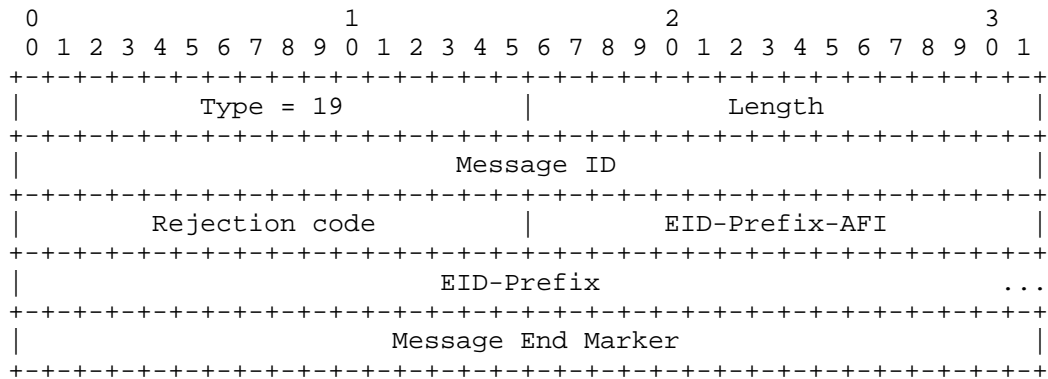
Registration Acknowledgement message format

- o EID-Prefix AFI: Address family identifier for the EID prefix in the following field.

- o EID-Prefix: The EID prefix from the received Registration.

6.1.3. Registration Rejected Message

Negative acknowledgement sent from the Map-Server to the ETR to indicate that the registration of a specific EID prefix was rejected. The ETR must keep track of the fact that the registration of the EID prefix was rejected by the Map-Server and be prepared to re-register the mapping when requested through a failed Registration Refresh request.

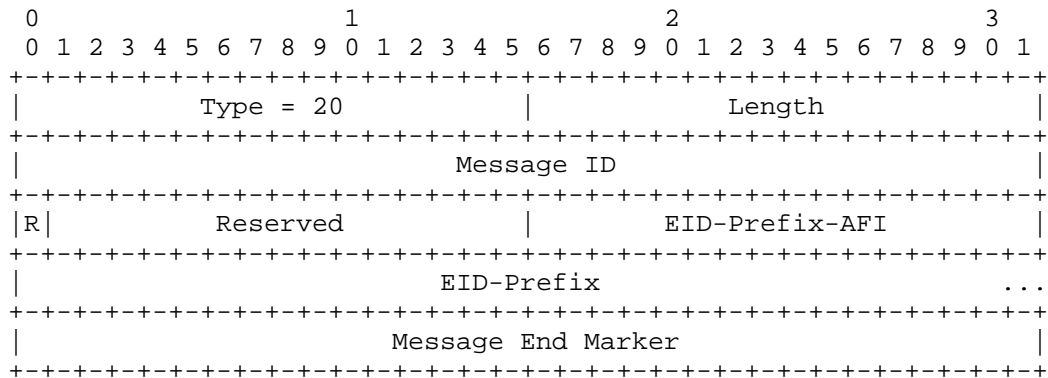


Registration Rejected message format

- o Rejection code: Code identifying the reason for which the Map-Server rejected the registration. Codes:
 - * 1 - Not a valid site EID prefix.
 - * 2 - Authentication failure.
 - * 3 - Locator set not allowed.
- o EID-Prefix AFI: Address family identifier for the EID prefix in the following field.
- o EID-Prefix: The EID prefix from the received Registration.

6.1.4. Registration Refresh Message

Sent by the Map-Server to the ETR to request the re-transmission of EID prefix database mapping Registration messages.

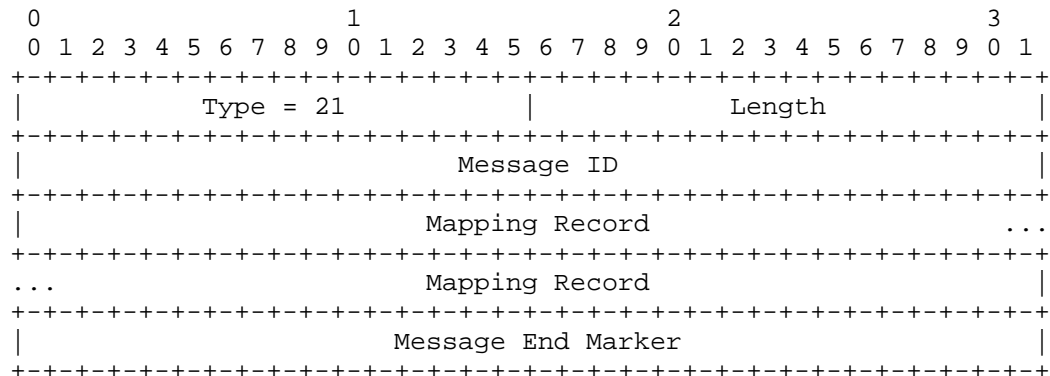


Registration Refresh message format

- o R: Request from the ETR to only refresh registrations that have been previously rejected by the Map-Server.
- o EID prefix, and its more specifics, to refresh. The prefix can be in LCAF format allowing specification of a complete refresh (unspecified prefix), refresh of all the prefixes under an EID instance or even of more specific registrations under a specific EID prefix.

6.1.5. Mapping Notification Message

Mapping Notification messages communicate the Map-Server view of the mapping for an EID prefix and no longer serve as a registration acknowledgement. Mapping Notifications do not need message level authentication as they are received over a reliable transport session to a known Map-Server. Note that reliable transport Mapping Notification messages do not reuse the UDP Map-Notify message format.



Registration message format

6.2. ETR Behavior

The ETR operates the following per EID prefix, per MS state machine that defines the reliable transport EID prefix registration behavior.

There are five states:

- o No state: The local EID database prefix does not exist.
- o Periodic: The local EID database prefix is being periodically registered through UDP Map-Register messages as specified in [1].
- o Stable: From the ETR's perspective, no registrations are due to be sent to the peer. The session to the peer is up, and the peer has either acknowledged the registration, or is expected to request a refresh in the future.
- o AckWait: A Registration message for the prefix has been transmitted to the Map-Server and the ETR is waiting for either a Registration Acknowledge or Registration Rejected reply from the Map-Server.
- o Reject: The reliable transport registration for the local EID database prefix was rejected by the Map-Server. From the ETR's perspective, no registration is due to the peer AND the peer is known to have rejected the registration.

The following events drive the state transitions:

- o DB creation: The local EID database entry for the EID prefix is created.

- o DB deletion: The local EID database entry for the EID prefix is deleted.
- o DB change: The mapping contents or authentication information for the local EID database entry changes.
- o Session up: The reliable transport session to the Map-Server is established.
- o Session down: The reliable transport session the Map-Server goes down.
- o Recv Refresh: A Registration refresh message is received from the Map-Server.
- o Recv ACK: A Registration Acknowledge message is received from the Map-Server.
- o Recv Rejected: A Registration Rejected message is received from the Map-Server.
- o Periodic timer: The timer that drives generation of periodic UDP Map-Register messages fires.

The state machine is:

Event	Prev State	
	No state	Periodic
DB creation [session down]	-> Periodic A1	N/A
DB creation [session up]	-> AckWait A2	N/A
DB deletion	N/A	-> No state A3
DB change	N/A	- A1
Session up	-	-> Stable A4
Session down	-	N/A
Recv Refresh	-	N/A
Recv Refresh [rejected]	-	N/A
Recv ACK	-	N/A
Recv Rejection	-	N/A
Timer	N/A	- A5

xTR per EID prefix per MS state machine

Event	Prev State		
	Stable	AckWait	Rejected
DB creation	N/A	N/A	N/A
DB deletion	-> No state A6	-> No state A6	-> No state
DB change	-> AckWait A2	- A2	-> AckWait A2
Session up	N/A	N/A	N/A
Session down	-> Periodic A7	-> Periodic A7	-> Periodic A7
Recv Refresh	-> AckWait A2	- A2	-> AckWait A2
Recv Refresh [rejected]	-	- A2	-> AckWait A2
Recv ACK	-	-> Stable	-> AckWait A2
Recv Rejection	-> Rejected	-> Rejected	-
Timer	N/A	N/A	N/A

xTR per EID prefix per MS state machine

Action descriptions:

- o A1: Start periodic registration timer with zero delay.
- o A2: Send Registration over reliable transport session.
- o A3: Send UDP registration with zero TTL.
- o A4: Stop periodic registration timer.

- o A7: Send UDP registration and start periodic registration timer with registration period.
- o A6: Send Registration with TTL zero over reliable transport session.
- o A7: Start periodic registration timer with registration period.

All timer start actions must be jittered.

When the reliable transport session is established the state machine moves into the Stable state without first registering the EID prefix over the reliable transport session. The subsequent refresh issued by the Map-Server will trigger the registration message to be sent. This model will allow future optimisations where the Map-Server may retain registration state from a previous instantiation of the reliable transport session with the ETR and only request the refresh of EID prefix state beyond some negotiated session progress marker.

Aa Map-Server authentication key change is treated as a DB change event and will result in triggering a new Registration message to be transmitted.

6.3. Map-Server Behavior

Received registrations create/update or delete mapping state.

A refresh for an unspecified prefix is sent when a session is first established to obtain the complete database contents from the ETR.

Refresh for rejected registrations sent (R bit set) when a new EID prefix is configured on the Map-Server.

Rejection sent to the ETR when an EID prefix that is registered is deconfigured.

Rejected Refresh (R bit set) sent when authentication for an EID prefix changes followed by a Rejection for existing registrations which fail authentication following change.

Mapping Notification message sent whenever the mapping for a registered or more specific prefix for which notifications are requested changes. ETR acknowledgement or rejection messaging for Mapping Notification is not required because the ETR decides how to process the message based on the registered mapping information. If the mapping information changes the resulting registration will trigger a new Mapping Notification message from the Map-Server.

7. Security Considerations

The LISP reliable transport session SHOULD be authenticated. On controlled RLOC networks that can guarantee that the source RLOC address of data packets cannot be spoofed, the authentication check can be a source address validation on the reliable transport packets. When the RLOC network does not provide such guarantees, reliable transport authentication SHOULD be used. Implementations SHOULD support the TCP Authentication Option (TCP-AO) [RFC5925] and SCTP Authenticated Chunks [RFC4895].

8. IANA Considerations

8.1. LISP Reliable Transport Message Types

Assignment of new LISP reliable transport message types is done according to the "IETF Review" model defined in [RFC5266].

The initial content of the registry should be as follows.

Type	Name	Reference
0-15	Reserved	This document
16	Error Notification	This document
17	Registration Message	This document
18	Registration Acknowledgement Message	This document
19	Registration Rejected Message	This document
20	Registration Refresh Message	This document
21	Mapping Notification Message	This document
22-30	Reserved for EID membership distribution	TBD
31-64999	Unassigned	
65000-65535	Reserved for Experimental Use	

8.2. Transport Protocol Port Numbers

TCP port 4342 already reserved for LISP CONS that is now obsolete. Repurpose for reliable transport over TCP. Reserve an SCTP port.

9. Acknowledgments

The authors would like to thank Noel Chiappa, Dino Farinacci, Jesper Skriver, Johnson Leong, Andre Pelletier and Les Ginsberg for their contributions to this document.

10. Normative References

- [I-D.ietf-lisp-lcaf]
Farinacci, D., Meyer, D., and J. Snijders, "LISP Canonical Address Format (LCAF)", draft-ietf-lisp-lcaf-05 (work in progress), May 2014.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC5266] Devarapalli, V. and P. Eronen, "Secure Connectivity and Mobility Using Mobile IPv4 and IKEv2 Mobility and Multihoming (MOBIKE)", BCP 136, RFC 5266, June 2008.
- [RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, January 2013.
- [RFC6833] Fuller, V. and D. Farinacci, "Locator/ID Separation Protocol (LISP) Map-Server Interface", RFC 6833, January 2013.

Authors' Addresses

Chris Cassar
Cisco Systems
10 New Square Park
Bedfont Lakes, Feltham TW14 8HA
United Kingdom

Email: ccassar@cisco.com

Isidor Kouvelas
Cisco Systems
Monumental Plaza, Building C
44 Kifissias Ave.
Maroussi, Athens 15125
Greece

Email: kouvelas@cisco.com

Darrel Lewis
Cisco Systems
Tasman Drive
San Jose, CA 95134
USA

Email: darlewis@cisco.com

Network Working Group
Internet-Draft
Intended status: Experimental
Expires: March 26, 2015

C. Cassar
I. Kouvelas
J. Leong
D. Lewis
G. Schudel
Cisco Systems
September 22, 2014

LISP RLOC Membership Distribution
draft-kouvelas-lisp-rloc-membership-00.txt

Abstract

The Locator/ID Separation Protocol (LISP) operation is based on EID to RLOC mappings that are exchanged through a mapping system. The mapping system can use the RLOCs included in mapping registrations to construct the complete set of RLOC addresses across all xTRs that are members of the LISP deployment. This set can then be made available by the mapping system to all the member xTRs. An xTR can use the RLOC set to optimise protocol operation as well as to implement new functionality. This document describes the use of the LISP reliable transport session between an xTR and a Map-Server to communicate the contents of the RLOC membership set.

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on March 26, 2015.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Requirements Notation	3
3. Membership Distribution Overview	4
4. Membership Message Format	4
4.1. Membership Subscribe	5
4.2. Membership Subscribe ACK	6
4.3. Membership Subscribe NACK	7
4.4. Membership Unsubscribe	8
4.5. Membership Element Add	9
4.6. Membership Element Delete	10
4.7. Membership Refresh Request	10
4.8. Membership Refresh Begin	11
4.9. Membership Refresh End	12
5. Membership Distribution Message Exchange	12
6. Security Considerations	14
7. IANA Considerations	14
8. Acknowledgments	14
9. References	15
9.1. Normative References	15
9.2. Informative References	15
Authors' Addresses	15

1. Introduction

The Locator/ID Separation Protocol (LISP) registration process between an xTR and a Map-Server is defined in [RFC6833]. In each registration message the xTR communicates mapping records providing the list of routing locators (RLOCs) that can be used to reach the endpoint identifier (EID) space behind the xTR. By gleaning the RLOCs from all such registrations, the map-server constructs the set of RLOCs across all the received registrations. This set represents all the RLOCs used to encapsulate traffic and is the complete RLOC membership of the LISP network (limitations described below).

The gleaned RLOC membership set is communicated to the member xTRs where it can be used to implement new functionality as well as to

optimise protocol operation. As one example in deployments where the RLOC network provides guarantees against RLOC source address spoofing the membership can be used as a decapsulation filter to prevent injection of traffic by non-members. As a second example, a possible optimisation to existing functionality can use changes to the RLOC membership set to validate the xTR map-cache contents and trigger updates for out-of-date mappings.

Distribution of the RLOC membership set is practical in VPN use cases [I-D.lewis-lisp-vpns] where the number of member xTRs and their RLOCs is bounded thus limiting both the number of membership elements that must be distributed as well as the number of members that the set must be distributed to. In a VPN use case the membership set is specific to each VPN identified through the LISP Instance ID (IID). It is reasonable to expect that all member xTRs for a specific VPN can register against a pair of redundant Map-Servers. The complete membership set will therefore be available on those Map-Servers. Alternatively, registration can be across a small set of Map-Servers that synchronise the RLOC membership set between them (outside the scope of this document). In the general case the RLOC membership knowledge is split across a distributed mapping system [I-D.ietf-lisp-ddt] and its collection and distribution would hit scale limits.

Membership gleaning at the Map-Server assumes symmetric ITR and ETR deployments. All encapsulating ITRs also have to be configured as ETRs registering against the Map-Servers. This is a common way of deploying LISP xTRs. To allow members that do not own EID space (such as exclusive ITRs and proxy routers) to be included in the membership set the registration mechanism must be extended.

Note that automatic membership gleaning at the Map-Server through registrations is just one mechanism that can be used to discover the RLOC set to be distributed. This document focuses on the membership set distribution mechanism.

The LISP extension in [I-D.kouvelas-lisp-reliable-transport] introduces a reliable transport session between the xTR and the MS. The membership set communication described in this document is based on message exchange over the reliable transport.

2. Requirements Notation

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

3. Membership Distribution Overview

The RLOC membership set distribution from the Map-Server to the xTR is initiated on demand by the xTR. Unless the xTR specifically subscribes to receive the RLOC membership set no action is taken by the Map-Server. The granularity at which a Map-Server gleans membership, and that an xTR can request its distribution, is per EID address family and instance ID. This matches the VPN EID space segmentation model allowing separate communication of the membership of different VPNs. It also allows for each EID address family to have a different xTR membership.

The Map-Server SHOULD only allow the distribution of the RLOC membership set for an EID instance and address family to xTRs that are valid members of the set being distributed. An xTR that has a reliable transport session established with the Map-Server and is registering EID prefixes with the Map-Server but not for the specific instance ID and EID address family, SHOULD NOT be sent the RLOC membership set.

The set of member RLOCs for an EID address family and instance ID is dynamic and changes as new registrations are received by the Map-Server and as registration state times out. When membership distribution is initiated by the xTR, the complete RLOC set contents is communicated. In parallel updates to the membership set begin being communicated. The membership set updates continue for the duration of the reliable transport session or until the xTR unsubscribes from the membership distribution.

4. Membership Message Format

The membership distribution exchange between the xTR and Map-Server over the reliable transport session relies on a number of new messages defined below. The use of these messages is described in the following sections. The table below lists the messages. All messages carry the EID address family and instance ID for the membership distribution. Some messages additionally carry extra fields that are listed in the table. The new messages are:

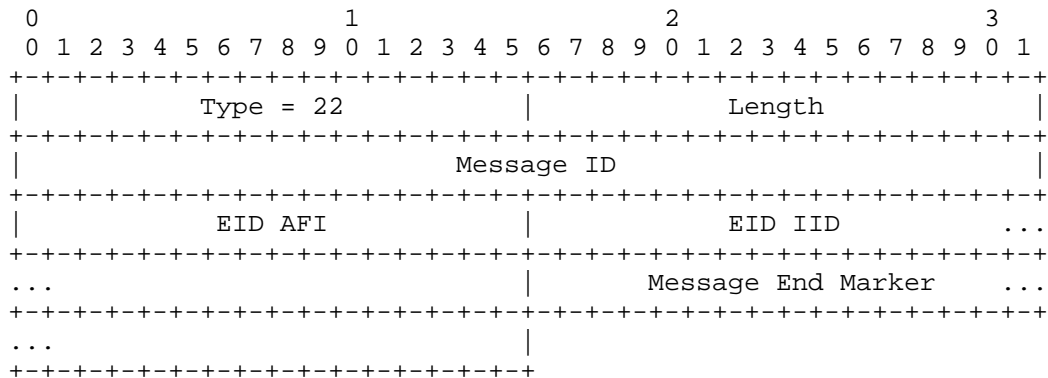
Type	Message	Direction	Additional fields
22	Subscribe	xTR -> MS	
23	Subscribe ACK	MS -> xTR	Subscribe ID
24	Subscribe NACK	MS -> xTR	Subscribe ID, Error
25	Unsubscribe	xTR -> MS	
26	Element Add	MS -> xTR	Site-ID, RLOC
27	Element Delete	MS -> xTR	Site-ID, RLOC
28	Refresh Request	xTR -> MS	
29	Refresh Begin	MS -> xTR	Request ID
30	Refresh End	MS -> xTR	Request ID

Table 1: Reliable transport membership distribution TLVs

The rest of this section provides the format of each of the messages in the table. For a description of the Type, Length, Message ID and Message End Marker fields refer to [I-D.kouvelas-lisp-reliable-transport].

4.1. Membership Subscribe

The Membership subscribe message is sent by the xTR to the Map-Server to initiate RLOC membership set distribution for a specific EID AFI and instance ID.

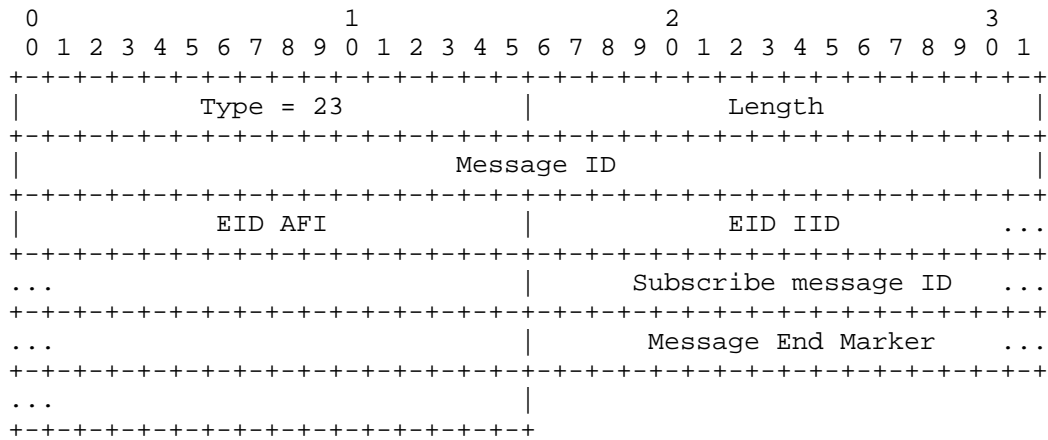


Membership subscribe message format

- o EID-AFI: EID address family for which the membership is being requested.
- o EID IID: The EID instance ID identifying the VPN for which the membership is being requested [I-D.lewis-lisp-vpns]. Although the IID is only 24 bits in size in the data encapsulation, it is being defined as a 32 bit field for consistency with the LCAF Instance ID header [I-D.ietf-lisp-lcaf].

4.2. Membership Subscribe ACK

The Membership-Subscribe-ACK message is sent by the Map-Server to the xTR to acknowledge acceptance of a Membership-Request. This message indicates that the Map-Server will be providing the requested membership to the xTR.

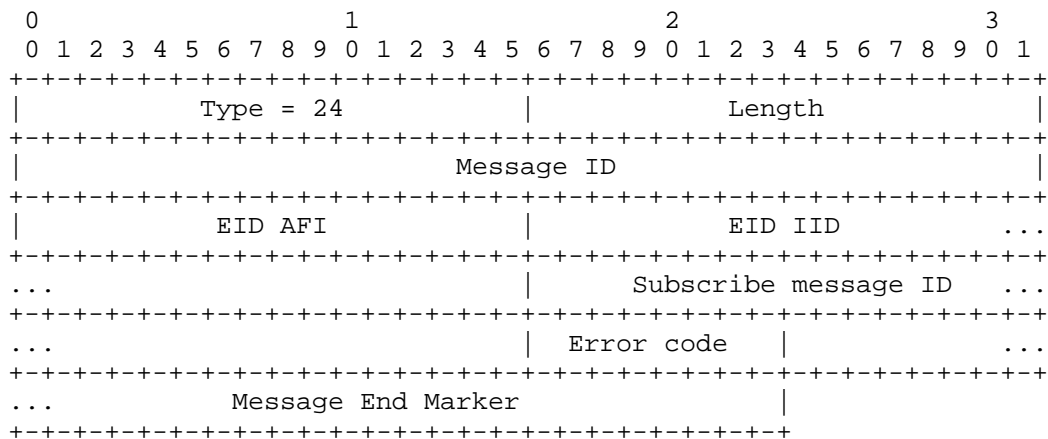


Membership-Subscribe-ACK message format

- o Subscribe message ID: The message ID carried over from the membership subscribe message.

4.3. Membership Subscribe NACK

The Membership-Subscribe-NACK message is sent by the Map-Server to the xTR to reject a membership request. This message indicates that the Map-Server will not be providing the requested membership to the xTR. The membership subscribe NACK message can be sent at any point following the receipt of a Membership-Subscribe message. The Map-Server may initially acknowledge a subscription with a Membership Subscribe ACK and later when conditions change cancel the subscription by issuing a membership subscribe NACK message.

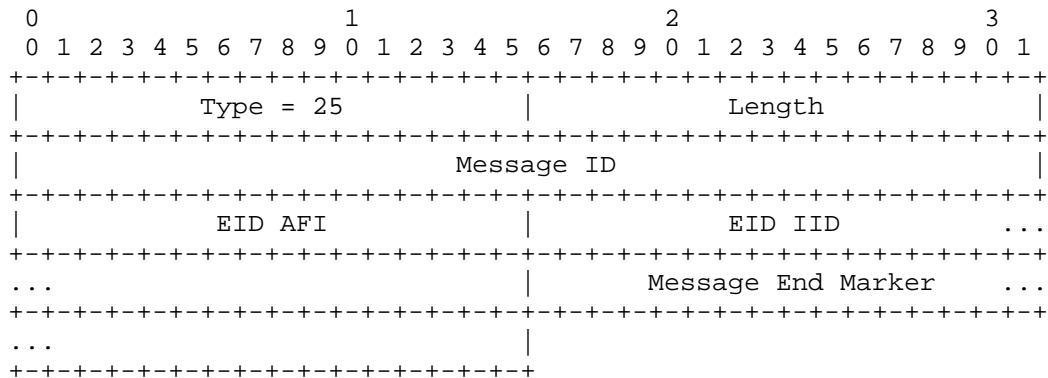


Membership subscribe NACK message format

- o Subscribe message ID: The message ID carried over from the membership subscribe message.
- o Error code: The error code provides a reason for which the registration was rejected by the Map-Server. Defined values are:
 - 1 - Not found: The EID instance and address family do not match the Map-Server configuration.
 - 2 - Not enabled: The Map-Server is not configured to allow membership distribution for the requested EID instance and address family.
 - 3 - Not authorized: The xTR that sent the request does not have a valid registration under the EID instance and address family.

4.4. Membership Unsubscribe

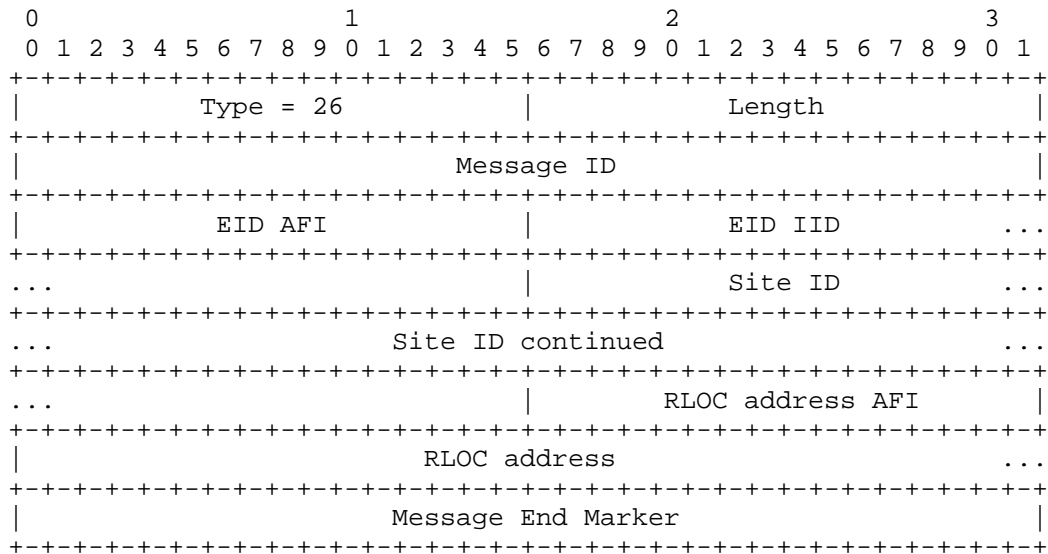
The Membership-Unsubscribe message is sent by the xTR to the Map-Server to terminate RLOC membership set distribution for a specific EID AFI and instance ID.



Membership-Unsubscribe message format

4.5. Membership Element Add

The Membership-Element-Add message is sent by the Map-Server to the xTR to communicate a single RLOC that is a member of the set for the specified EID instance and address family.



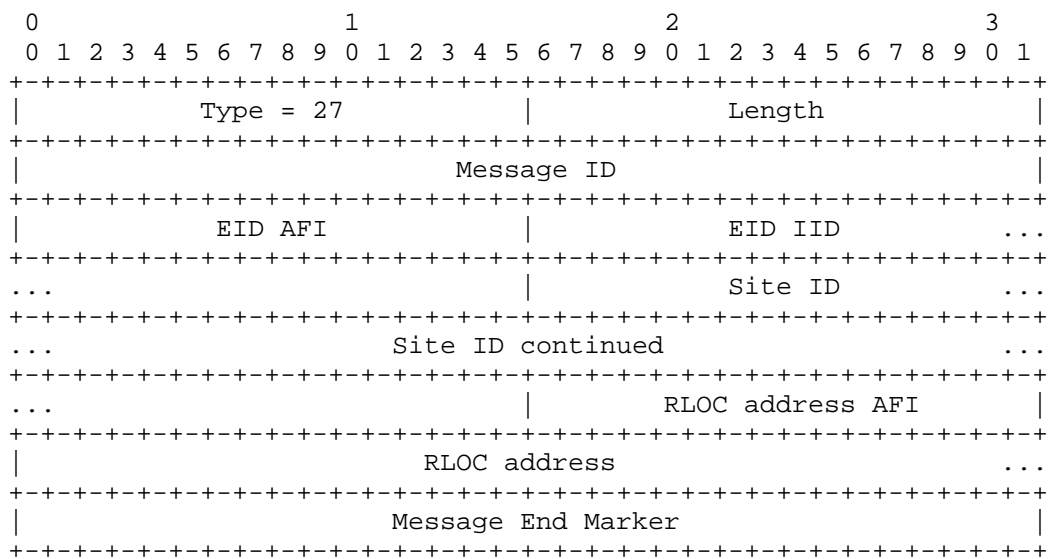
Membership-Element-Add message format

- o Site ID: The 64 bit site ID value from the mapping registration that contributed this RLOC to the membership list. The site ID can be used by the receiving xTR to derive information about the grouping of member RLOCs to remote sites.

- o RLOC address AFI: Address family identifier for the RLOC address in the following field.
- o RLOC address: The actual RLOC membership set element address being communicated. Note that the length of this field depends on the RLOC address AFI in the preceding field.

4.6. Membership Element Delete

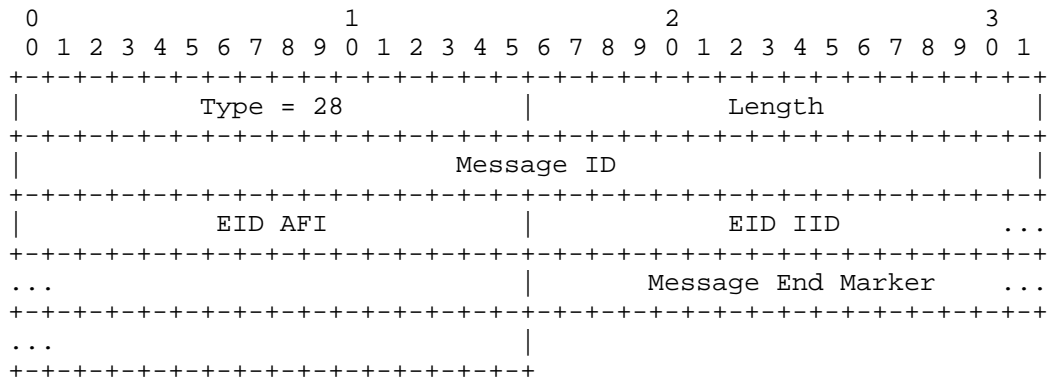
The Membership-Element-Delete message is sent by the Map-Server to the xTR to communicate a single RLOC that is no longer a member of the set for the specified EID instance and address family.



Membership-Element-Delete message format

4.7. Membership Refresh Request

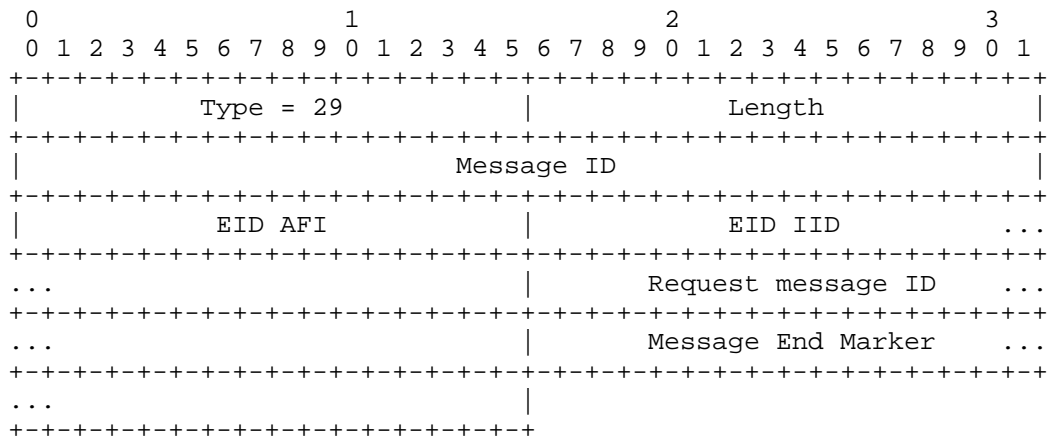
The Membership-Refresh-Request message is sent by the xTR to the Map-Server to request that the Map-Server send the complete RLOC membership set contents for the specified instance ID and AFI.



Membership-Refresh-Request message format

4.8. Membership Refresh Begin

The Membership-Refresh-Begin message is sent by the Map-Server to the xTR to acknowledge an earlier Membership-Refresh-Request message and to indicate that the following membership updates are part of the refresh.

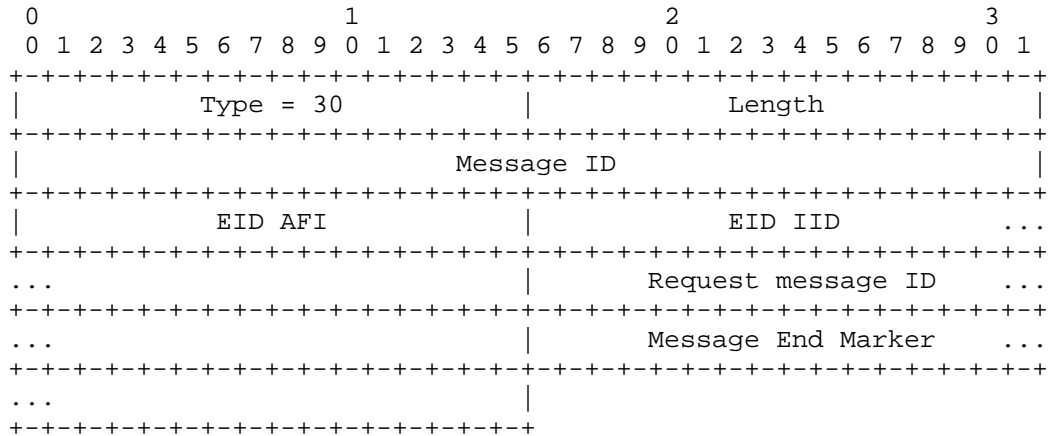


Membership-Refresh-Begin message format

- o Request message ID: The message ID carried over from the membership request message.

4.9. Membership Refresh End

The Membership-Refresh-End message is sent by the Map-Server to the xTR to indicate that the communication of the full membership refresh for the specified EID instance ID and AF is now complete.



Membership-Refresh-End message format

- o Request message ID: The message ID carried over from the membership request message.

5. Membership Distribution Message Exchange

Following the reliable transport session establishment, the EID membership communication relies on the exchange of the membership messages defined in the previous section. The description in this section presents the exchange from the perspective of a single xTR and Map-Server.

xTR	MS
----- Subscribe ----->	
<----- Subscribe ACK -----	
---- Refresh request --->	
<----- Refresh begin -----	
<----- Element add -----	
<----- Element add -----	
<----- Element add -----	
<----- Refresh end -----	
<-- Element add/delete --	
----- Unsubscribe ----->	

Typical membership distribution message exchange

The xTR starts the exchange by issuing a Membership-Subscribe-Request message to the Map-Server for a specific EID instance. Assuming the Map-Server is configured to allow membership distribution and the requesting router is authorized to receive the membership of the EID instance, the MS will reply with a Membership-Subscribe-ACK. After sending the ACK, the MS will start sending to the xTR Membership-Element-Add and Membership-Element-Delete messages corresponding to changes of the EID instance membership.

On receipt of the Membership-Subscribe-ACK message, the xTR issues a Membership-Refresh-Request message in order to receive the complete contents of the EID instance membership held by the MS. The MS responds to the Membership-Refresh-Request by issuing a Membership-Refresh-Begin message, followed by a Membership-Element-Add message for each member of the EID instance and finally completes the refresh by sending a Membership-Refresh-End message.

On receipt of Membership-Element-Add and Membership-Element-Delete messages, the xTR updates its membership database for the EID instance ID and address family by adding or deleting the entry corresponding to the communicated RLOC address. Note that the membership state on the xTR is Map-Server specific and the xTR has to maintain separate RLOC membership entries received from each Map-Server it subscribes with.

When the xTR receives the Membership-Refresh-End message it purges all the stale membership entries it may have obtained during a previous session instantiation that were not updated during the refresh.

The MS may issue Membership-Element-Add and Membership-Element-Delete messages corresponding to membership changes at any point after issuing the Membership-ACK message, even during a refresh.

The xTR may request additional full refreshes of the complete membership set at any point after having received a Membership-Subscribe-ACK message by issuing a new Membership-Refresh-Request.

When the Map-Server determines that an xTR is no longer eligible to receive membership updates, for example the EID instance and address family registration state of the xTR becomes invalid, then the Map-Server SHOULD send it a Membership-NACK message to indicate the termination of the membership communication.

6. Security Considerations

The RLOC membership distribution message communication takes place over a LISP reliable transport connection. The security mechanisms of the reliable transport apply to this solution.

7. IANA Considerations

The following message types must be assigned out of the space defined in [I-D.kouvelas-lisp-reliable-transport].

Type	Name	Reference
22	Membership Subscribe	This document
23	Membership Subscribe ACK	This document
24	Membership Subscribe NACK	This document
25	Membership Unsubscribe	This document
26	Membership Element Add	This document
27	Membership Element Delete	This document
28	Membership Refresh Request	This document
29	Membership Refresh Begin	This document
30	Membership Refresh End	This document

8. Acknowledgments

The authors would like to thank Michiel Blokzijl, Selina Heimlich, Vasileios Lakafosis, Fabio Maino, Andre Pelletier, Jesper Skriver and Chao Yu, for their contributions to this specification.

9. References

9.1. Normative References

- [I-D.kouvelas-lisp-reliable-transport]
Cassar, C., Kouvelas, I., and D. Lewis, "LISP Reliable Transport", draft-kouvelas-lisp-reliable-transport-01 (work in progress), August 2014.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, January 2013.
- [RFC6833] Fuller, V. and D. Farinacci, "Locator/ID Separation Protocol (LISP) Map-Server Interface", RFC 6833, January 2013.

9.2. Informative References

- [I-D.ietf-lisp-ddt]
Fuller, V., Lewis, D., Ermagan, V., and A. Jain, "LISP Delegated Database Tree", draft-ietf-lisp-ddt-01 (work in progress), March 2013.
- [I-D.ietf-lisp-lcaf]
Farinacci, D., Meyer, D., and J. Snijders, "LISP Canonical Address Format (LCAF)", draft-ietf-lisp-lcaf-05 (work in progress), May 2014.
- [I-D.lewis-lisp-vpns]
Lewis, D. and G. Schudel, "LISP Virtual Private Networks (VPNs)", draft-lewis-lisp-vpns-00 (work in progress), February 2014.

Authors' Addresses

Chris Cassar
Cisco Systems
10 New Square Park
Bedfont Lakes, FELTHAM TW14 8HA
UNITED KINGDOM

Email: ccassar@cisco.com

Isidor Kouvelas
Cisco Systems
Monumental Plaza, Building C
44 Kifissias Ave.
Maroussi, Athens 15125
Greece

Email: kouvelas@cisco.com

Johnson Leong
Cisco Systems
Tasman Drive
San Jose, CA 95134
USA

Email: joleong@cisco.com

Darrel Lewis
Cisco Systems
Tasman Drive
San Jose, CA 95134
USA

Email: darlewis@cisco.com

Gregg Schudel
Cisco Systems
Tasman Drive
San Jose, CA 95134
USA

Email: gschudel@cisco.com

Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: June 18, 2018

D. Lewis
Cisco
J. Lemon
Broadcom
P. Agarwal
Innovium
L. Kreeger

P. Quinn
M. Smith
N. Yadav
F. Maino, Ed.
Cisco

December 15, 2017

LISP Generic Protocol Extension
draft-lewis-lisp-gpe-04

Abstract

This draft describes extending the Locator/ID Separation Protocol (LISP), via changes to the LISP header, to support multi-protocol encapsulation.

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on June 18, 2018.

Copyright Notice

Copyright (c) 2017 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
1.1. Conventions	3
1.2. Definition of Terms	3
2. LISP Header Without Protocol Extensions	3
3. Generic Protocol Extension for LISP (LISP-GPE)	3
4. Backward Compatibility	5
4.1. Type of Service	5
4.2. VLAN Identifier (VID)	5
5. IANA Considerations	5
6. Security Considerations	6
7. Acknowledgements	6
8. References	6
8.1. Normative References	6
8.2. Informative References	7
Authors' Addresses	7

1. Introduction

LISP, as defined in [RFC6830] and extended in [I-D.ietf-lisp-rfc6830bis], defines an encapsulation format that carries IPv4 or IPv6 (henceforth referred to as IP) packets in a LISP header and outer UDP/IP transport.

The LISP header does not specify the protocol being encapsulated and therefore is currently limited to encapsulating only IP packet payloads. Other protocols, most notably VXLAN [RFC7348] (which defines a similar header format to LISP), are used to encapsulate L2 protocols such as Ethernet.

This document defines an extension for the LISP header, as defined in [I-D.ietf-lisp-rfc6830bis], to indicate the inner protocol, enabling the encapsulation of Ethernet, IP or any other desired protocol all the while ensuring compatibility with existing LISP deployments.

A flag in the LISP header, called the P-bit, is used to signal the presence of the 8-bit Next Protocol field. The Next Protocol field,

when present, uses 8 bits of the field allocated to the echo-noncing and map-versioning features. The two features are still available, albeit with a reduced length of Nonce and Map-Version.

1.1. Conventions

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 RFC 2119 [RFC2119].

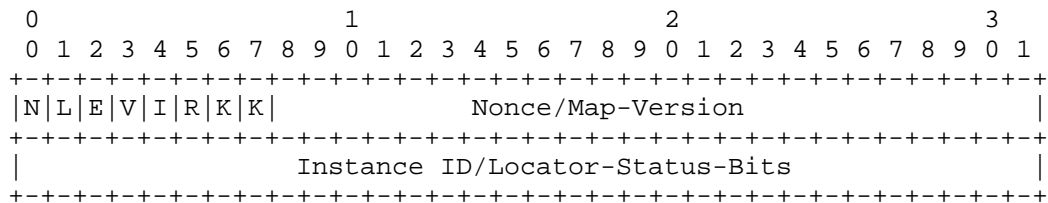
1.2. Definition of Terms

This document uses terms already defined in [I-D.ietf-lisp-rfc6830bis].

2. LISP Header Without Protocol Extensions

As described in the introduction, the LISP header has no protocol identifier that indicates the type of payload being carried. Because of this, LISP is limited to carry IP payloads.

The LISP header [I-D.ietf-lisp-rfc6830bis] contains a series of flags (some defined, some reserved), a Nonce/Map-version field and an instance ID/Locator-status-bit field. The flags provide flexibility to define how the various fields are encoded. Notably, Flag bit 5 is the last reserved bit in the LISP header.



LISP Header

3. Generic Protocol Extension for LISP (LISP-GPE)

This document defines the following changes to the LISP header in order to support multi-protocol encapsulation:

P Bit: Flag bit 5 is defined as the Next Protocol bit. The P bit MUST be set to 1 to indicate the presence of the 8 bit next protocol field.

P = 0 indicates that the payload MUST conform to LISP as defined in [I-D.ietf-lisp-rfc6830bis]. Flag bit 5 was chosen as the P bit because this flag bit is currently unallocated.

Next Protocol: The lower 8 bits of the first 32-bit word are used to carry a Next Protocol. This Next Protocol field contains the protocol of the encapsulated payload packet.

LISP uses the lower 24 bits of the first word for either a nonce, an echo-nonce, or to support map-versioning [RFC6834]. These are all optional capabilities that are indicated in the LISP header by setting the N, E, and the V bit respectively.

When the P-bit and the N-bit are set to 1, the Nonce field is the middle 16 bits.

When the P-bit and the V-bit are set to 1, the Version field is the middle 16 bits.

When the P-bit is set to 1 and the N-bit and the V-bit are both 0, the middle 16-bits are set to 0.

This draft defines the following Next Protocol values:

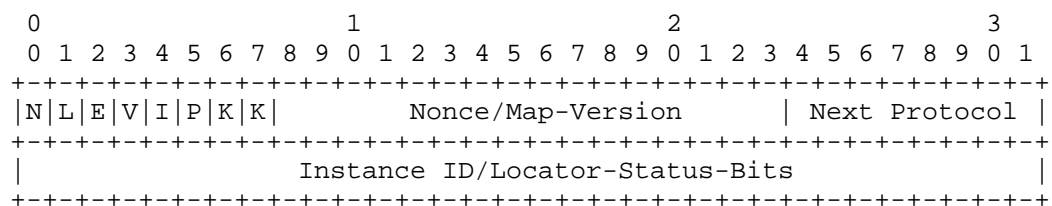
0x1 : IPv4

0x2 : IPv6

0x3 : Ethernet

0x4 : Network Service Header [I-D.ietf-sfc-nsh]

0x6: Group-Based Policy (GBP) [I-D.lemon-vxlan-gpe-gbp].



LISP-GPE Header

4. Backward Compatibility

LISP-GPE uses the same UDP destination port (4341) allocated to LISP.

A LISP-GPE router MUST not encapsulate non-IP packets to a LISP router. A method for determining the capabilities of a LISP router (GPE or "legacy") is out of the scope of this draft.

When encapsulating IP packets to a LISP "legacy" router the P bit MUST be set to 0.

4.1. Type of Service

When a LISP-GPE router performs Ethernet encapsulation, the inner 802.1Q [IEEE8021Q] priority code point (PCP) field MAY be mapped from the encapsulated frame to the Type of Service field in the outer IPv4 header, or in the case of IPv6 the 'Traffic Class' field.

4.2. VLAN Identifier (VID)

When a LISP-GPE router performs Ethernet encapsulation, the inner header 802.1Q [IEEE8021Q] VLAN Identifier (VID) MAY be mapped to, or used to determine the LISP Instance ID field.

5. IANA Considerations

IANA is requested to set up a registry of LISP-GPE "Next Protocol". These are 8-bit values. Next Protocol values in the table below are defined in this draft. New values are assigned via Standards Action [RFC5226].

Next Protocol	Description	Reference
0	Reserved	This Document
1	IPv4	This Document
2	IPv6	This Document
3	Ethernet	This Document
4	NSH	This Document
5	Reserved	
6	GBP	This Document
7	Reserved	
8..255	Unassigned	

6. Security Considerations

LISP-GPE security considerations are similar to the LISP security considerations documented at length in [I-D.ietf-lisp-rfc6830bis]. With LISP-GPE, issues such as dataplane spoofing, flooding, and traffic redirection may depend on the particular protocol payload encapsulated.

7. Acknowledgements

A special thank you goes to Dino Farinacci for his guidance and detailed review.

8. References

8.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", RFC 5226, DOI 10.17487/RFC5226, May 2008, <<https://www.rfc-editor.org/info/rfc5226>>.
- [RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, DOI 10.17487/RFC6830, January 2013, <<https://www.rfc-editor.org/info/rfc6830>>.
- [RFC6834] Iannone, L., Saucez, D., and O. Bonaventure, "Locator/ID Separation Protocol (LISP) Map-Versioning", RFC 6834, DOI 10.17487/RFC6834, January 2013, <<https://www.rfc-editor.org/info/rfc6834>>.
- [RFC7348] Mahalingam, M., Dutt, D., Duda, K., Agarwal, P., Kreeger, L., Sridhar, T., Bursell, M., and C. Wright, "Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks", RFC 7348, DOI 10.17487/RFC7348, August 2014, <<https://www.rfc-editor.org/info/rfc7348>>.

8.2. Informative References

[I-D.ietf-lisp-rfc6830bis]

Farinacci, D., Fuller, V., Meyer, D., Lewis, D., and A. Cabellos-Aparicio, "The Locator/ID Separation Protocol (LISP)", draft-ietf-lisp-rfc6830bis-07 (work in progress), November 2017.

[I-D.ietf-sfc-nsh]

Quinn, P., Elzur, U., and C. Pignataro, "Network Service Header (NSH)", draft-ietf-sfc-nsh-28 (work in progress), November 2017.

[I-D.lemon-vxlan-gpe-gbp]

Lemon, J., Maino, F., and M. Smith, "Group Policy Encoding with VXLAN-GPE", draft-lemon-vxlan-gpe-gbp-00 (work in progress), October 2017.

Authors' Addresses

Darrel Lewis
Cisco Systems

Email: darlewis@cisco.com

John Lemon
Broadcom
3151 Zanker Road
San Jose, CA 95134
USA

Email: john.lemon@broadcom.com

Puneet Agarwal
Innovium
USA

Email: puneet@acm.org

Larry Kreeger
USA

Email: lkreeger@gmail.com

Paul Quinn
Cisco Systems

Email: pquinn@cisco.com

Michael Smith
Cisco Systems

Email: michsmit@cisco.com

Navindra Yadav
Cisco Systems

Email: nyadav@cisco.com

Fabio Maino (editor)
Cisco Systems
San Jose, CA 95134
USA

Email: fmaino@cisco.com

Network Working Group
Internet-Draft
Intended status: Informational
Expires: April 27, 2015

D. Saucez
INRIA
L. Iannone
Telecom ParisTech
A. Cabellos
F. Coras
Technical University of Catalonia
October 24, 2014

LISP Impact
draft-saucez-lisp-impact-07.txt

Abstract

The Locator/Identifier Separation Protocol (LISP) aims at improving the Internet scalability properties leveraging on three simple principles: address role separation, encapsulation, and mapping. In this document, based on implementation, deployment, and theoretical studies, we discuss the impact that deployment of LISP can have on both the Internet in general and for the end-users in particular.

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on April 27, 2015.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents

carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. LISP in a nutshell	3
3. LISP for scaling the Internet	4
4. Beyond scaling the Internet	5
4.1. Traffic engineering	6
4.2. LISP for IPv6 Co-existence	7
4.3. Inter-domain multicast	8
5. Impact of LISP on operations and business model	8
5.1. Impact on non-LISP traffic and sites	8
5.2. Impact on LISP traffic and sites	9
6. IANA Considerations	11
7. Security Considerations	11
8. Acknowledgments	11
9. References	11
9.1. Normative References	11
9.2. Informative References	12
Authors' Addresses	15

1. Introduction

The Locator/Identifier Separation Protocol (LISP) relies on three simple principles to scale the Internet: address role separation, encapsulation, and mapping. The main goal of LISP is to make the Internet more scalable by reducing the number of prefixes announced in the Default Free Zone (DFZ) as well as its related churn. As LISP relies on mapping and encapsulation, it turns out that it provides more benefits than just scalability. For example, LISP provides a mean for a LISP site to precisely control its inter-domain outgoing and incoming traffic, with the possibility to apply different policies to the different domains exchanging traffic with it. LISP can also be used to ease the transition from IPv4 to IPv6 as it allows to transport IPv4 over IPv6 or IPv6 over IPv4. Furthermore, LISP also provides a solution to perform inter-domain multicast.

This document discusses the impact of LISP's deployment on the Internet and on end-users and shows the consequences of the interworking infrastructure in path stretch. There still are many, economical rather than technical, open questions related to the deployment of such infrastructure. Moreover, encapsulation may raise some issues (that do not have a real impact in practice) because it

reduces the Maximum Transmission Unit (MTU) size. An important impact of LISP on network operations is related to resiliency and troubleshooting. Indeed, as LISP relies on cached mappings and on encapsulation, troubleshooting is harder than in the traditional Internet. Also, end-to-end encapsulation stresses resiliency as it makes failure detection and recovery slower than with hop-by-hop routing.

2. LISP in a nutshell

The Locator/Identifier Separation Protocol (LISP) relies on three simple principles: address role separation, encapsulation, and mapping.

Semantics of address are separated in two: the Routing Locators (RLOCs) and the Endpoint Identifiers (EIDs). RLOCs are assigned from the address space of the Internet service providers (PA). The EIDs are attributed, to the nodes in the edge network, by block of contiguous addresses extracted from the EID Space. To limit the scalability problem of today's Internet, only the routes towards the RLOCs are announced in the Internet while EIDs are also propagated today.

LISP routers are used at the boundary between the EID and the RLOC spaces. Routers used to exit the EID space are called Ingress Tunnel Router (ITRs) and those used to enter the EID space the Egress Tunnel Routers (ETRs). When a host sends a packet to a remote destination, it sends it as in today's Internet. The packet eventually arrives at the border of its site at an ITR. Because EIDs are not routable on the Internet, the packet is encapsulated with the source address set to the ITR RLOC and the destination address set to the ETR RLOC. The encapsulated packet is then forwarded in the Internet until it reaches the selected ETR. The ETR decapsulates the packet and forwards it to its final destination. The acronym xTR for Ingress/Egress tunnel router is used for a router playing these two roles.

The correspondence between EIDs and RLOCs is given by the mappings. When an ITR needs to find ETR RLOCs that serve an EID it queries the mapping system. It is worth noticing that with the LISP Canonical Address Format (LCAF) [I-D.ietf-lisp-lcaf], LISP is not restricted to the Internet Protocol for the EID addresses. With LCAF, any address type can be used as EID (the address is the key for the mapping lookup) and LISP can then transport, for example, Ethernet frames over the Internet.

A more thorough introduction to LISP can be found in [I-D.ietf-lisp-introduction]. The complete specifications are given

in [RFC6830], [RFC6833], [I-D.fuller-lisp-ddt], [RFC6836], [RFC6832], [RFC6834], and [I-D.ietf-lisp-sec].

3. LISP for scaling the Internet

The first goal of LISP is to scale the Internet. LISP improves the Internet's scalability because traffic engineering and stub AS prefixes are not propagated in the DFZ, so routing tables are smaller and more stable (i.e., less affected by churn). Also, at the edge network, information necessary to forward packets (i.e., the mappings) is usually obtained on demand using a pull model. Therefore, for each edge network they scale with the traffic matrix of the edge network and are independent of the Internet's size. This scaling improvement is proven by several works.

Quoitin et al. show in [QIdLB07] that the separation between locator and identifier roles at the network level improves the routing scalability by reducing the RIB size (up to one order of magnitude) and increases the path diversity and thus the traffic engineering capabilities. In addition, Iannone and Bonaventure show in [IB07] that the number of mapping entries that must be supported at an ITR of a 10,000 users campus network is limited and does not represent more than 3 to 4 Megabytes of memory. Furthermore, they show that signaling traffic (i.e., Map-Request/Map-Reply packets) is in the same order of magnitude like DNS requests traffic and that encapsulation overhead, while not negligible, is very limited (in the order of few percentage points of the total traffic volume). Similarly, Kim et al. show that the EID-to-RLOC cache size should not exceed 14 MB for an ITR responsible of more than 20,000 residential ADSL users at a large ISP [KIF11]. [IB07], [KIF11] rely on BGP and traffic traces to determine the number of entries to keep in the EID-to-RLOC cache. In both papers, the size of the cache is inferred from the number of entries by considering that every EID is associated with two or three locators. [S11] confirms these results by looking at the distribution of the number of locators per EID if LISP were deployed in the 2010's Internet. The assumptions in these studies are:

- o contiguous addresses tend to be used similarly, EID prefixes follow the current BGP prefixes decomposition;
- o EIDs are used only at the stub ASes, not in the transit ASes;
- o the RLOCs of an EID prefix are deployed at the edge between the stubs owning the EID prefix and the providers and locator addresses are allocated in a Provider Aggregatable (PA) mode.

While all previous studies consider the case of a timer-based cache eviction policy (i.e., mappings are deleted from the cache upon timeout), [CCD12] generalizes the caching discussion for the Least Recently Used (LRU) eviction policy and proposes an analytic model for the EID-to-RLOC cache size when prefix-level traffic has a stationary generating process. The model shows that miss rate can be accurately predicted from the EID-to-RLOC cache size and a small set of easily measurable traffic parameters. The model was validated using four one-day-long packet traces collected at egress points of a campus network and an academic exchange point considering EID-prefixes as being of BGP-prefix granularity. Consequently, operators can provision the EID-to-RLOC cache of their ITRs according to the miss rate they want to achieve for their given traffic.

The results indicate that for a given miss ratio, cache size only depends on the parameters of the popularity distribution and is in fact independent of the number of users (the size of the LISP site) and the number of destinations (the size of the EID-prefix space). Assuming that the popularity distribution remains constant, this means that as the number of users and the number of destinations grow, the cache size needed to obtain a given miss rate remains constant $O(1)$.

Under normal user traffic, miss-ratio decreases at an accelerated pace with cache size and finally settles to a power-law decrease. However, [CDLC] extends the model to account for scanning attacks, whereby attackers generate a constant flux of packets according to random scans of the destination prefix space and shows that miss-ratios are be very high and independent of cache size. In fact, if the attack is merely 1% of the legitimate traffic, the miss rate does not drop under 1% as long as the cache cannot accommodate the whole prefix space. Locality measurements also suggested that LRU eviction policy should be close to optimal.

TBD: add a paragraph to explain thhe operational difference while dealing with a pull model instead of a push.

4. Beyond scaling the Internet

Even though it is its main goal, LISP is more than just a scalability solution, it is also a tool to provide both incoming and outgoing traffic engineering [S11], can be used as an IPv6 transition at the routing level, and for inter-domain multicast [RFC6831], [I-D.coras-lisp-re]. LISP has also proven to be a good protocol for mobility of devices in the Internet [I-D.meyer-lisp-mn] or even virtual machine mobility in data centers and multi-tenant VPN, however, we don't further discuss in details the two last points as they are out of the scope of the charter.

Lisp architecture facilitates routing in environments where there is little to no correlation between network endpoints and topological location. In service provider environment this use is evident in a range of consumer use cases which require an inline anchor in-order to deliver a service to a subscribers. Inline anchors provide one of three types of capabilities:

- o enable mobility of subscriber end points
- o enable chaining of middle-box functions
- o enable seamless scale-out of functions

Without LISP operators are forced to centralize service anchors in custom built special boxes. This means that end-points can move as long as their traffic ends up on the same mobile gateway, functions can be chained as long as all traffic traverses the same wire or the same DPI box, and capacity can scale out as long as traffic fans out to and form a specific load balancer.

With LISP service providers are able to distribute, virtualize, and insatiate subscriber-service anchors anywhere in the network. Typical use cases that Virtualize inline anchors and network functions include: Distributed Mobility and Virtualized Evolved Packet Core (vEPC), where centralization makes way to distributed and virtualized inline anchoring of mobility, Virtualized Customer Premise Equipment or vCPE, where functionality previously anchored at customer prem is now dynamically allocated in-network, Virtualized SGi LAN, where value added mobile services previously anchored inside full-stack boxes or anchored to physical wires with permutation setups aka "Rails", Virtual IMS and Virtual SBC, etc.

Current deployments by ConteXtream, using a pre standards (designed 2006) based architecture, support a total of 100 millions subscribers with such an architecture. A deployment at a tier-1 US Mobile operator over 50 millions subscribers provides a 39% download rate improvement over LTE.

4.1. Traffic engineering

In today's Internet, stub networks are globally routable and the routing system distributes the routes to reach these stubs. On the contrary, the EID prefixes of a LISP site are not routable on the Internet and mappings are needed to determine the list of LISP routers to contact to send them packets. The difference is significant for two reasons. First, packets are not sent to a site but to a specific ingress router. Second, a site can control the entry points for its traffic by controlling its mappings.

For traffic engineering purpose, a mapping associates an EID prefix to a list of RLOCs. Each RLOC is annotated with a priority and a weight. When there are several RLOCs, the ITR selects the one with the lowest priority value and sends the encapsulated packet to this RLOC. If several such RLOCs exist, then the traffic is balanced proportionally to their weight among the RLOCs with the lowest priority value. Traffic engineering in LISP thus allows the mapping owner to have a fine-grained control on the primary and backup path its incoming and outgoing packets use. In addition, it can share the load among its links. An example of the use of such a feature is described in [SDIB08], where Saucez et al. show how to use LISP to direct different types of traffic on different links having different capacity.

Traffic engineering in LISP goes one step further. As every Map-Request contains the Source EID Address of the packet that caused a cache miss and triggered the Map-Request. It is thus possible for a mapping owner to differentiate the answer (Map-Reply) it gives to Map-Requests based on the requester. This functionality is not available today with BGP because a domain cannot control exactly the routes that will be received by domains that are not in the direct neighborhood.

4.2. LISP for IPv6 Co-existence

The LISP encapsulation mechanism is designed to support any combination of locators and identifiers address family. It is then possible to bind IPv6 EIDs with IPv4 RLOCs and vice-versa. This allows transporting IPv6 packets over an IPv4 network (or IPv4 packets over an IPv6 network), making LISP a valuable mechanism to ease the transition to IPv6.

A not so uncommon example is the case of the network infrastructure of a datacenter being IPv4-only while dual-stack front-end load balancers are used. In this scenario, LISP can be used to provide IPv6 access to servers even though the network and the servers only support IPv4. Assuming that the datacenter's ISP offers IPv6 connectivity, the datacenter only needs to deploy one (or more) xTR(s) at its border with the ISP and one (or more) xTR(s) directly connected to the load balancers. The xTR(s) at the ISP's border tunnels IPv6 packets over IPv4 to the xTR(s) directly attached to the load balancer. The load balancer's xTR decapsulates the packets and forward them to the load balancer, which act as proxies, translating each IPv6 packet into an IPv4. IPv4 packets are then sent to the appropriate servers. Similarly, when the server's response arrives at the load balancer, the packet is translated back into an IPv6 packet and forwarded to its xTR(s), which in turn will tunnel it back, over the IPv4-only infrastructure, to an xTR connected to the

ISP. The packet is then decapsulated and forwarded to the ISP natively in IPv6.

4.3. Inter-domain multicast

LISP has native support for multicast [RFC6831]. From the data-plane perspective, at a multicast enabled xTR, an EID sourced multicast packet is encapsulated in another multicast packet and subsequently forwarded in a RLOC-level distribution tree. Therefore, xTRs must participate in both EID and RLOC level distribution trees. Control-plane wise, since group addresses have no topological significance they need not be mapped. It is worth noting that, to properly function inter-domain, LISP-Multicast requires that inter-domain multicast be prior deployed.

[I-D.coras-lisp-re] and [CDM12] propose a technique to construct xTR based inter-domain multicast distribution trees. Simulations of three different management strategies for low latency content delivery show that such overlays can support thousands of member xTRs, hundreds of thousands of end-hosts and deliver content at latencies close to unicast ones [CDM12]. It was also observed that high client churn has a limited impact on performance and management overhead.

5. Impact of LISP on operations and business model

Important implementation efforts ([IOSNXOS], [OpenLISP], [LISPmob], [LISPClick], [LISPcp], and [LISPfritz]) have been made to assess the specifications and interoperability tests [Was09] have been a success. World-wide large deployment in the international lisp4.net testbed, which is currently composed of nodes running at least three different implementations, allows to learn operational matters related to LISP.

We have to distinguish the impact of LISP on LISP sites from the impact on non-LISP sites.

5.1. Impact on non-LISP traffic and sites

LISP has no impact on traffic which has neither LISP origin nor LISP destination. However, LISP can have a significant impact on traffic between a LISP site and a non-LISP site. Traffic between a non-LISP site and a LISP site are subject to the same issues than those observed for LISP-to-LISP traffic (cf infra) but also have issues specific to the transition mechanism that allow LISP site to exchange packets with non-LISP site ([RFC6832], [I-D.ietf-lisp-deployment]).

Indeed, the transition requires to setup proxy tunnel routers (PxTRs). PxTRs do not cause particular technical issue. However, by definition proxies cause path stretch and make troubleshooting harder. There are still big questions related to PxTRs that have to be answered:

- o Where to deploy PxTRs? The placement in the topology has an important impact on the path stretch.
- o How many PxTRs? The number of PxTR has a direct impact on the load and the impact of the failure of a PxTR on the traffic.
- o What part of the EID space? Will all the PxTRs be proxies for the whole EID space or will it be segmented between different PxTRs?
- o Who to operate PxTRs? The IETF does not aim at providing business model hints, however, an important question to answer is related to the entities that will deploy PxTRs, how they will manage their CAPEX/OPEX and how the traffic will be carried with respect for the security and privacy.

PxTR also normally have to advertise in BGP the EID prefix they are proxy for. However, if proxies are managed by different entities, they will belong to different ASes. In this case, we have to be sure that it will not cause MOA issues that could negatively influence routing. Moreover, we have to be sure that the way EID prefixes will be deaggregated by the proxies will remain reasonable to not take part in the BGP scalability issues.

5.2. Impact on LISP traffic and sites

LISP is a protocol based on the map-and-encap paradigm which has the positive effects that we have given in the sections above. However, by design, LISP also has side impact on operations:

MTU issue: as LISP uses encapsulation, the MTU is reduced, this has implication on potentially all the traffic. However, in practice, on the lisp4.net network, no major issue due to the MTU has been observed. This is probably due to the fact that current end-host stacks are well designed to deal with the problem of MTU.

Resiliency issue: the advantage of flexibility and control offered by the Locator/ID separation comes at the cost of increasing the complexity of the reachability detection. Indeed, identifiers are not directly routable and have to be mapped to locators but a locator may be unreachable while others are still reachable. This is an important problem for any tunnel-

based solution. In the current Internet, packets are forwarded independently of the border router of the network meaning that in case of the failure of a border router, another one can be used. With LISP, the destination RLOC specifically designate one particular ETR, hence if this ETR fails, the traffic is dropped even though other ETRs are available for the destination site. Another resiliency issue is linked to the fact that mappings are learned on demand. When an ITR fails, all its traffic is redirected to other ITRs that might not have yet the mappings for the redirected traffic. The study in [SKI12] and [SD12] show, based on measurements and traffic traces, that failure of ITRs and RLOC are infrequent but that when such failure happens, an important number of packet can be dropped. Unfortunately, the current techniques for LISP resiliency, based on monitoring or probing are not rapid enough (failure recovery of the order of a few seconds). To tackle this issue [I-D.bonaventure-lisp-preserve] and [I-D.saucez-lisp-itr-graceful] propose techniques based on local failure detection and recovery.

Middle boxes/filters: because of encapsulation, the middle boxes might not understand the traffic which can cause firewall to drop legitimate packets. In addition, LISP allows triangular or even rectangular routing, so it is hard to maintain a correct state even if the middle box perfectly understands LISP. Finally, filtering might also have problems because they might think only one host is generating the traffic (the ITR), as long as it is not decapsulated. To deal with LISP encapsulation, LISP aware firewalls that inspect inner LISP packets are proposed [lispfirewall].

Troubleshooting/debugging: the major issue years of LISP experimentation have shown is the difficulty of troubleshooting. When there is a problem in the network, it is hard to pin-point the reason as the operator only has a partial view of the network. The operator can see what is in its EID-to-RLOC cache/database, and can try to obtain what is potentially elsewhere by querying the Map Resolvers but the knowledge remains partial. On top of that, ICMP is too small, which means that when an ICMP arrives at the ITR, it might not contain enough information to make correct troubleshooting. Interestingly, deployment in the beta network has shown that LISP+ALT was not easy to maintain and control, which explains the migration to LISP-DDT [I-D.fuller-lisp-ddt].

Business: the IETF is not aiming at providing business models. However, even though [IL10] shown that there is economical incentives to migrate to LISP, some questions are on hold. For

example, how will the EIDs be allocated to allow aggregation and hence scalability of the mapping system? Who will operate the mapping system infrastructure and for what benefit?

6. IANA Considerations

This document makes no request to the IANA.

7. Security Considerations

Security and threats analysis of the LISP protocol is out of the scope of the present document. A thorough analysis of LISP security threats is detailed in [I-D.ietf-lisp-threats].

8. Acknowledgments

The people that contributed to this document are Sharon Barkai, Vince Fuller, Joel Halpern, Terry Manderson, and Gregg Schudel.

9. References

9.1. Normative References

- [I-D.fuller-lisp-ddt]
Fuller, V., Lewis, D., Ermagan, V., and A. Jain, "LISP Delegated Database Tree", draft-fuller-lisp-ddt-04 (work in progress), September 2012.
- [I-D.ietf-lisp-deployment]
Jakab, L., Cabellos-Aparicio, A., Coras, F., Domingo-Pascual, J., and D. Lewis, "LISP Network Element Deployment Considerations", draft-ietf-lisp-deployment-12 (work in progress), January 2014.
- [I-D.ietf-lisp-sec]
Maino, F., Ermagan, V., Cabellos-Aparicio, A., and D. Saucez, "LISP-Security (LISP-SEC)", draft-ietf-lisp-sec-07 (work in progress), October 2014.
- [RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, January 2013.
- [RFC6831] Farinacci, D., Meyer, D., Zwiebel, J., and S. Venaas, "The Locator/ID Separation Protocol (LISP) for Multicast Environments", RFC 6831, January 2013.

- [RFC6832] Lewis, D., Meyer, D., Farinacci, D., and V. Fuller, "Interworking between Locator/ID Separation Protocol (LISP) and Non-LISP Sites", RFC 6832, January 2013.
- [RFC6833] Fuller, V. and D. Farinacci, "Locator/ID Separation Protocol (LISP) Map-Server Interface", RFC 6833, January 2013.
- [RFC6834] Iannone, L., Saucez, D., and O. Bonaventure, "Locator/ID Separation Protocol (LISP) Map-Versioning", RFC 6834, January 2013.
- [RFC6836] Fuller, V., Farinacci, D., Meyer, D., and D. Lewis, "Locator/ID Separation Protocol Alternative Logical Topology (LISP+ALT)", RFC 6836, January 2013.

9.2. Informative References

- [CCD12] Coras, F., Cabellos-Aparicio, A., and J. Domingo-Pascual, "An Analytical Model for the LISP Cache Size", In Proc. IFIP Networking 2012, May 2012.
- [CDLC] Coras, F., Domingo, J., Lewis, D., and A. Cabellos, "An Analytical Model for Loc/ID Mappings Caches", Technical Report <http://arxiv.org/pdf/1312.1378v2.pdf>, 2013.
- [CDM12] Coras, F., Domingo-Pascual, J., Maino, F., Farinacci, D., and A. Cabellos-Aparicio, "Lcast: Software-defined Inter-Domain Multicast", Technical Report, Universitat Politcnica de Catalunya, 2012, July 2012.
- [I-D.bonaventure-lisp-preserve] Bonaventure, O., Francois, P., and D. Saucez, "Preserving the reachability of LISP ETRs in case of failures", draft-bonaventure-lisp-preserve-00 (work in progress), July 2009.
- [I-D.chiappa-lisp-architecture] Art, Y., "An Architectural Perspective on the LISP Location-Identity Separation System", draft-chiappa-lisp-architecture-01 (work in progress), July 2012.
- [I-D.coras-lisp-re] Coras, F., Cabellos-Aparicio, A., Domingo-Pascual, J., Maino, F., and D. Farinacci, "LISP Replication Engineering", draft-coras-lisp-re-05 (work in progress), April 2014.

- [I-D.ietf-lisp-introduction]
Cabellos-Aparicio, A. and D. Saucez, "An Architectural Introduction to the Locator/ID Separation Protocol (LISP)", draft-ietf-lisp-introduction-06 (work in progress), October 2014.
- [I-D.ietf-lisp-lcaf]
Farinacci, D., Meyer, D., and J. Snijders, "LISP Canonical Address Format (LCAF)", draft-ietf-lisp-lcaf-06 (work in progress), October 2014.
- [I-D.ietf-lisp-threats]
Saucez, D., Iannone, L., and O. Bonaventure, "LISP Threats Analysis", draft-ietf-lisp-threats-10 (work in progress), July 2014.
- [I-D.meyer-lisp-mn]
Farinacci, D., Lewis, D., Meyer, D., and C. White, "LISP Mobile Node", draft-meyer-lisp-mn-11 (work in progress), July 2014.
- [I-D.saucez-lisp-itr-graceful]
Saucez, D., Bonaventure, O., Iannone, L., and C. Filsfils, "LISP ITR Graceful Restart", draft-saucez-lisp-itr-graceful-03 (work in progress), December 2013.
- [IB07] Iannone, L. and O. Bonaventure, "On the cost of caching locator/id mappings", In Proc. ACM CoNEXT 2007, December 2007.
- [IL10] Iannone, L. and T. Leva, "Modeling the economics of Loc/ID Separation for the Future Internet", Book Chapter, Towards the Future Internet - Emerging Trends from the European Research, IOS Press, May 2010.
- [IOSNXOS] Cisco Systems Inc., , "Locator/ID Separation Protocol (LISP)", <http://lisp4.cisco.com>, 2013.
- [KIF11] Kim, J., Iannone, L., and A. Feldmann, "Deep dive into the lisp cache and what isps should know about it", In Proc. IFIP Networking 2011, May 2011.
- [LISPClick]
Saucez, D. and V. Nguyen, "LISP-Click: A Click implementation of the Locator/ID Separation Protocol", 1st Symposium on Click Modular Router, 2009, November 2009.

- [LISPcp] "The lip6-lisp Project", <https://github.com/lip6-lisp/>, 2014.
- [LISPfritz] "Unsere FRITZ!Box-Produkte", <http://avm.de/produkte/fritzbox/>, 2014.
- [LISPMob] "LISP Mobile Node for Linux", <http://lispmob.org>, 2013.
- [OpenLISP] "The OpenLISP Project", <http://www.openlisp.org>, 2013.
- [QIdLB07] Quoitin, B., Iannone, L., de Launois, C., and O. Bonaventure, "Evaluating the benefits of the locator/identifier separation", In Proc. ACM MobiArch 2007, May 2007.
- [S11] Saucez, D., "Mechanisms for Interdomain Traffic Engineering with LISP", PhD Thesis, Universite catholique de Louvain, 2011, October 2011.
- [SD12] Saucez, D. and B. Donnet, "On the Dynamics of Locators in LISP", In Proc. IFIP Networking 2012, May 2012.
- [SDIB08] Saucez, D., Donnet, B., Iannone, L., and O. Bonaventure, "Interdomain Traffic Engineering in a Locator/Identifier Separation Context", In Proc. of Internet Network Management Workshop, 2008, October 2008.
- [SKI12] Saucez, D., Kim, J., Iannone, L., Bonaventure, O., and C. Filsfils, "A Local Approach to Fast Failure Recovery of LISP Ingress Tunnel Routers", In Proc. IFIP Networking 2012, May 2012.
- [Was09] Wasserman, M., "LISP Interoperability Testing", IETF 76, LISP WG presentation, 2009., November 2009.
- [lispfirewall] "LISP and Zone-Based Firewalls Integration and Interoperability", http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/sec_data_zbf/configuration/xe-3s/sec-data-zbf-xe-book/sec-zbf-lisp-inner-pac-insp.html, 2014.

Authors' Addresses

Damien Saucez
INRIA
2004 route des Lucioles BP 93
06902 Sophia Antipolis Cedex
France

Email: damien.saucez@inria.fr

Luigi Iannone
Telecom ParisTech
23, Avenue d'Italie, CS 51327
75214 PARIS Cedex 13
France

Email: luigi.iannone@telecom-paristech.fr

Albert Cabellos
Technical University of Catalonia
C/Jordi Girona, s/n
08034 Barcelona
Spain

Email: fcoras@ac.upc.edu

Florin Coras
Technical University of Catalonia
C/Jordi Girona, s/n
08034 Barcelona
Spain

Email: fcoras@ac.upc.edu

Internet Engineering Task Force
Internet-Draft
Intended status: Experimental
Expires: January 21, 2015

N. Shen
Cisco Systems
D. Farinacci
lispers.net
July 20, 2014

LISP Multi-Provider VPN Use-Cases
draft-shen-lisp-multiprovider-vpn-00

Abstract

This document describes how LISP sites communicate with each other in a VPN when there are multiple mapping database systems administered by multiple providers. The detail of VPN segmentation across mapping databases will be provided.

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on January 21, 2015.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Requirements Language	2
2. Introduction	2
3. Definition of Terms	3
4. Overview	3
5. LISP Mapping Database System	5
6. LISP Packet Flow	7
6.1. Packet from Site1 to Site2	7
6.2. Packet from Site1 to Site3	7
6.3. Packet from Site1 to Site4	7
7. Security Considerations	7
8. IANA Considerations	8
9. References	8
9.1. Normative References	8
9.2. Informative References	8
Appendix A. Acknowledgments	8
Appendix B. Document Change Log	8
Authors' Addresses	9

1. Requirements Language

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 RFC 2119 [RFC2119].

2. Introduction

This document describes how the Locator/Identifier Separation Protocol (LISP) [RFC6830] is used for Multi-Provider LISP VPN where the providers maintain their own local LISP mapping databases, and their own security and crypto mechanisms within the provider's LISP network. Each provider will have a number of Gateway Tunnel Routers (GTR) to send and receive LISP encapsulated packets to and from the other provider. Those Gateway Tunnel Routers behave the same as the Re-Encapsulating Tunnel Routers (RTRs) on traffic engineered LISP paths. Special security mechanisms between a pair of GTRs from two providers can be enforced, such as firewall and IPsec encryption can be applied over this Multi-Provider LISP overlay tunnel. This specification will define how an Explicit Locator Path (ELP) [LISP-LCAF] can be used for an ITR to encapsulate an Multi-Provider VPN packet to its own Gateway Tunnel Router (GTR), then to the peering provider's Gateway Tunnel Router (GTR), and finally to the peering provider's ETR. This specification will examine how each provider's GTR can interface with its own local LISP mapping database system and allow the instance-ID allocated to sites of one provider can be different from the instance-ID allocated to sites in the other providers. This allows policy and control to be contained within

each provider while allowing segmented connectivity across providers with secure LISP overlay.

3. Definition of Terms

Mapping Service Provider (MSP): A LISP control-plane network provider which maintains its own LISP mapping database system.

LISP Provider Network: Is a delivery network that administers its own mapping database acting as an MSP as well as managing a set of GTRs so multi-provider VPNs can be provided.

Re-Encapsulating Tunnel Router (RTR): An RTR is a router that acts as an ETR (or PETR) by decapsulating packets where the destination address in the "outer" IP header is one of its own RLOCs. Then acts as an ITR (or PITR) by making a decision where to encapsulate the packet based on the next locator in the ELP towards the final destination ETR.

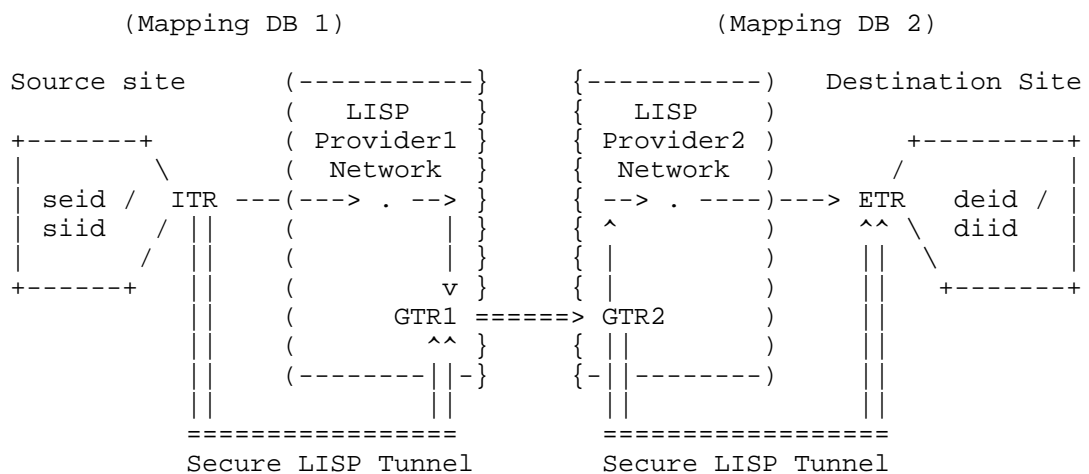
Gateway Tunnel Router (GTR): An GTR is a router serves as a gateway re-encapsulating tunnel router (RTR) on the edge of the provider's LISP network. It services as an ETR for one LISP network for sending out the packet to the other provider, and as an ITR for the another LISP network when receiving a packet from the other provider.

Re-Encapsulating Tunnels: Re-Encapsulating tunneling occurs when an RTR acts as an ETR and then an ITR on a given packet. As an ETR it removes a LISP header, then acts as an ITR to prepend another LISP header. Doing this allows a packet to be re-routed by the re-encapsulating router without adding the overhead of additional tunnel headers. Any references to tunnels in this specification refers to dynamic encapsulating tunnels and they are never statically configured. When using multiple mapping database systems, care must be taken to not create re-encapsulation loops through misconfiguration.

4. Overview

A packet that is sourced by an EID which is destined for an EID travels across a core network based on the locators that ITR uses as the outer source and destination addresses towards a given ETR. If the Mapping Service Provider (MSP) the ITR uses to obtain the destination ETR's locator address can be a local mapping database or one deployed on globally on the Internet.

In a LISP Provider Network, the security mechanism is usually applied with the LISP tunneled packets within a single administrative organization. When two LISP Provider Networks need to communicate with each other, it is undesirable to have xTRs belong to two organizations to simply exchange the crypto keys. This specification proposes the solution to have the xTRs setup normal LISP overlay to the local GTR, and to have both GTRs of peering LISP Provider Networks to share common crypto keys if needed, and use the LISP re-encapsulation tunnels to have the Multi-Provider VPN packets traffic engineered among two providers without compromising the security aspect of the VPNs and simplify the Multi-Provider secure VPN management.



Typical Multi-Provider Secure VPN Data Path from ITR to ETR

This diagram shows the packet flow from Provider1's network into the Provider2's network, and the other direction is logically the same. Although this diagram only showing one pair of GTRs between two providers, but in general, there can be a number of GTRs to be deployed on each side to either load share or for Multi-Provider VPN traffic engineering purposes. The policy agreed upon both providers decides which EIDs/IIDs to be exported to the other provider's mapping database.

The ETR in Provider2 network registers the destination EID/IID into its own mapping system (Mapping DB 2); base on the Multi-Provider VPN policies, Provider2 network will provide the VPN mapping information along with the gateway Tunnel Router (GTR2) hop address to Provider1.

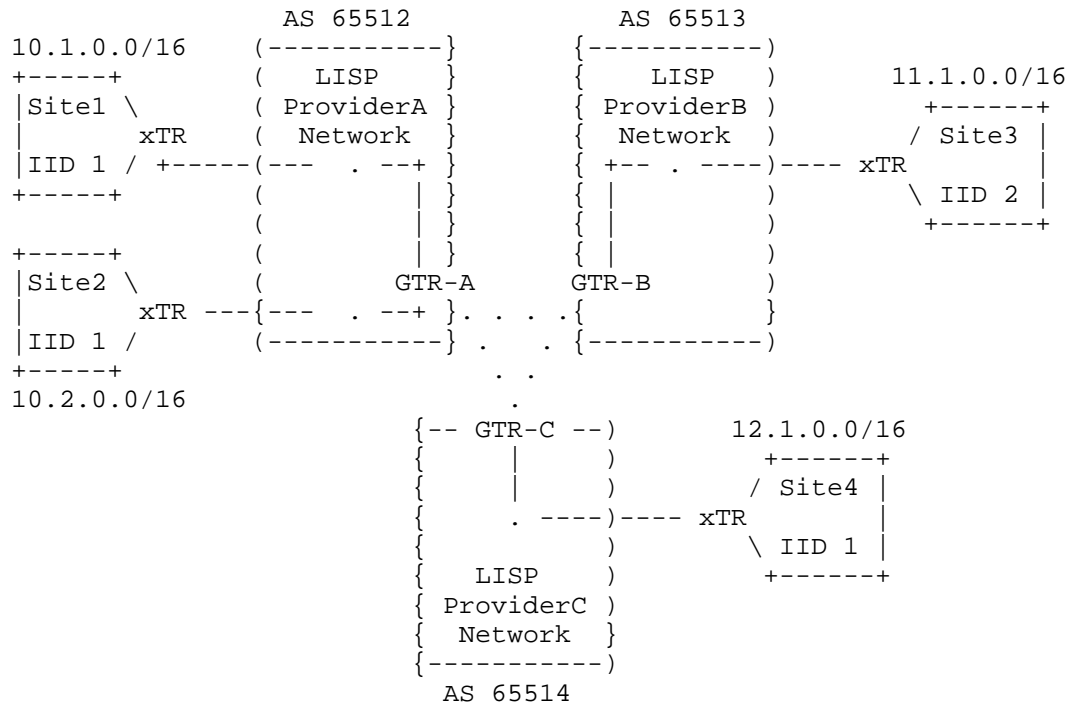
Provider1 will provision the Gateway Tunnel Router (GTR1) to register the Multi-Provider VPN LISP mapping into its own mapping database along with the ELP to list the GTR1 and GTR2 as hops.

An AS Number [LISP-LCAF] can be part of the EID mapping entry to clearly define the mapping is an Multi-Provider LISP entry and belong to which MSP network. Other usages of this AS Number includes to detect the LISP mapping system looping and to facilitate multi-provider LISP VPN trouble-shooting.

5. LISP Mapping Database System

For sites attached to LISP Provider Networks to communicate with one another in a VPN, we assume the EID space is unique, while the IID space is maintained individually by each LISP Provider Network and there is no coordination among providers on the LISP IIDs. The Multi-Provider VPN mapping entries are registered into its own mapping database by the GTRs.

Take an example to illustrate the mapping database systems in the Multi-Provider LISP VPN. In this instance, we have 4 LISP sites that want to be part of the same VPN. There are 3 LISP Provider Networks each which manage their own LISP mapping database systems. Each provider allocates IIDs to their local LISP sites and there is no IID space coordination among the MSPs.



Four Multi-Provider LISP VPN sites from three LISP Providers

In above diagram, there are three LISP Provider Networks and four LISP sites to be provisioned within the same Multi-Provider VPN. MSP A has two LISP sites with the same IID, and with prefix 10.1.0.0/16 in Site1 and 10.2.0.0/16 in Site2, IID 1; The MSP B has one site with IID 2 with prefix 11.1.0.0/16 in Site3 and the MSP C has one site with IID 1 with prefix 12.1.0.0/16 in Site4. The GTR-A, GTR-B and GTR-C are the re-encapsulation tunnel routers to facilitate Multi-Provider VPN communication among the three MSPs.

In ProviderA's mapping database (registered by GTR-A):

```
(IID1, 11.1.0.0/16) -> ELP: [GTR-A, (IID2, GTR-B)]
(IID1, 12.1.0.0/16) -> ELP: [GTR-A, (IID1, GTR-C)]
```

In ProviderB's mapping database (registered by GTR-B):

```
(IID2, 10.1.0.0/16) -> ELP: [GTR-B, (IID1, GTR-A)]
(IID2, 10.2.0.0/16) -> ELP: [GTR-B, (IID1, GTR-A)]
(IID2, 12.1.0.0/16) -> ELP: [GTR-B, (IID1, GTR-C)]
```

In ProviderC's mapping database (registered by GTR-C):

```
(IID2, 10.1.0.0/16) -> ELP: [GTR-C, (IID1, GTR-A)]  
[IID2, 10.2.0.0/16) -> ELP: [GTR-C, (IID1, GTR-A)]  
[IID2, 11.1.0.0/16) -> ELP: [GTR-C, (IID2, GTR-B)]
```

6. LISP Packet Flow

Using the same example as in the previous section, this section shows how the VPN packet flow operations either within the same LISP Provider Network sites as well as sites across LISP Provider Networks.

6.1. Packet from Site1 to Site2

This packet flow from 10.1 network to 10.2 network is within the same LISP Provider Network uses the traditional LISP-VPN mechanism [LISP-VPN]. Each ETR registers the site (IID1, eid-prefix) with ProviderA's mapping database. The xTR at Site1 sends the Map-Requests for(IID1, eid) to mapping database, and encapsulates the packet direct to the xTR at Site2.

6.2. Packet from Site1 to Site3

This is the Multi-Provider VPN case. The xTR at Site1 of 10.1 sends a Map-Request for (IID1, 11.1.1.1) to its local LISP mapping database. The returned result will be (IID1, 11.1.0.0/16) with ELP of [GTR-A, (IID2, GTR-B)]. The xTR at Site1 encapsulates to GTR-A with the IID1 in the LISP header. The GTR-A does a lookup on (IID1, 11.1.1.1) in its local mapping database (same as the xTRs) and it will use the second node in the ELP list. The GTR-A encapsulates the packet to GTR-B with IID2 in the LISP header. GTR-B decapsulates and looks up the (IID2, 11.1.1.1) in MSP B's local mapping database, and the RLOC of TR at Site3 is returned. The GTR-B encapsulates the packet to that RLOC. The xTR on Site3 decapsulates the packet and sends the packet to the host of 11.1.1.1.

6.3. Packet from Site1 to Site4

The operation is almost identical as the above sub-section except for that the IIDs between the two sites are the same. Thus there is no IID change during the packet hops across LISP Provider Networks.

7. Security Considerations

This specification allows provider's VPN to communicate with each other in a secure fashion. The LISP tunnel from ITR to GTR1 and from GTR2 to ETR may use their own encryption mechanisms with each

provider. There can be cases one provider uses encryption for the LISP overlay while the other provider does not. Also whether or not to use the encryption over the tunnel between the GTR1 and GTR2 depends on the data sensitivity and the underlining network. The GTRs MAY choose to drop the packet if the local security policy does not match Multi-Provider VPN packet attributes.

8. IANA Considerations

At this time there are no requests for IANA.

9. References

9.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, January 2013.

9.2. Informative References

- [LISP-LCAF] Farinacci, D., Meyer, D., and J. Snijders, "LISP Canonical Address Format", draft-ietf-lisp-lcaf-06 (work in progress), 2014.
- [LISP-VPN] Lewis, D. and G. Schudel, "LISP Virtual Private Networks (VPNs)", draft-ietf-lisp-vpns-00 (work in progress), 2014.
- [RFC2629] Rose, M., "Writing I-Ds and RFCs using XML", RFC 2629, June 1999.

Appendix A. Acknowledgments

TBD.

Appendix B. Document Change Log

Initial draft posted on July 2014.

Authors' Addresses

Naiming Shen
Cisco Systems
San Jose, California
USA

Email: naiming@cisco.com

Dino Farinacci
lispers.net
San Jose, California
USA

Phone: 408-718-2001
Email: farinacci@gmail.com