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

A. Cabellos
UPC-BarcelonaTech
D. Saucez (Ed.)
INRIA
September 22, 2014

An Architectural Introduction to the LISP Location-Identity Separation System
draft-ietf-lisp-introduction-05.txt

Abstract

This document describes the Locator/ID Separation Protocol (LISP) architecture, its main operational mechanisms as well as its design rationale.

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 March 26, 2015.

Table of Contents

1. Introduction

There is a rough consensus that the Internet routing and addressing system is facing severe scalability issues [RFC4984]. Specifically, the growth in the size of the routing tables of the Default-Free Zone (DFZ) is accelerating and showing a supra-linear slope [DFZ]. The main driving force behind this growth is the de-aggregation of BGP prefixes, which results from the existing BGP multihoming and traffic engineering mechanisms that are used -at the time of this writing- on the Internet, as well as non-aggregatable address allocations.

This issue has two profound implications, on the one hand Internet core routers are exposed to the network dynamics of the edge. For instance this typically leads to an increased amount of BGP UPDATE messages (churn), which results in additional processing requirements of Internet core routers in order to timely compute the DFZ RIB. Secondly, the supra-linear growth imposes strong requirements on the size of the memory storing the DFZ FIB. Both aspects lead to an increase on the development and production cost of high-end routers, and it is unclear if the semiconductor and router manufacturer industries will be able to cope, in the long-term, with such stringent requirements in a cost-effective way[RFC4984].

Although this important scalability issue is relatively new, the architectural reasons behind it are well-known many years ago. Indeed, and as pointed out by [Chiappa], IP addresses have overloaded semantics. 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.

The Locator/ID Separation Protocol (LISP), specified in [RFC6830], is built on top of this basic idea: decoupling the IP address overloaded semantics. LISP creates two separate namespaces, EIDs (End-host IDentifiers) and RLOCs (Routing LOCators), both are -typically, but not limited to- 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- are architecturally separated and interconnected by LISP-capable routers. LISP also introduces a publicly accessible 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, the Internet core is populated with RLOCs which can be quasi-static and highly aggregatable, hence scalable [Quoitin].

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] for the protocol specifications along with the LISP deployment guidelines [RFC7215].

2. LISP Architecture

This section presents the LISP architecture, we first detail the design principles of LISP and then we proceed to describe its main aspects: data-plane, control-plane, and internetworking mechanisms.

2.1. Design Principles

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

- Locator/Identifier split: By decoupling the overloaded semantics of the current IP addresses the Internet core can be assigned with topological meaningful address and hence, can use aggregation to scale. Devices are assigned with identity meaningful address that are independent of its topological location.

- Overlay architecture: Overlays route packets over the current Internet, allowing to deploy new protocols without changing the current infrastructure hence, resulting from a low deployment cost.

- 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.

- Incremental deployability: This principle ensures that the protocol is compatible with the legacy Internet while providing some of the targeted benefits to early adopters.

2.2. 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 (RLOC). The edge are 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 endhosts and are assigned and configured by the same mechanisms that we have at the time of this writing. EIDs can be are typically Provider Independent (PI [RFC4116]) addresses and can be thought as they don't contain intra-domain topological information. Because of this, EIDs are usually only routable in the edge.

With LISP, LISP sites (edge) and the core of the Internet are inter-connected by means of LISP-capable routers (e.g., border routers). When they provide egress (from the core perspective) to a LISP site they are called Egress Tunnel Routers (ETR), Ingress Tunnel Routers (ITR) when they provide ingress, and xTR when they provide both. ITRs and ETRs exchange packets by encapsulating them, hence LISP operates as an overlay to the current Internet core.

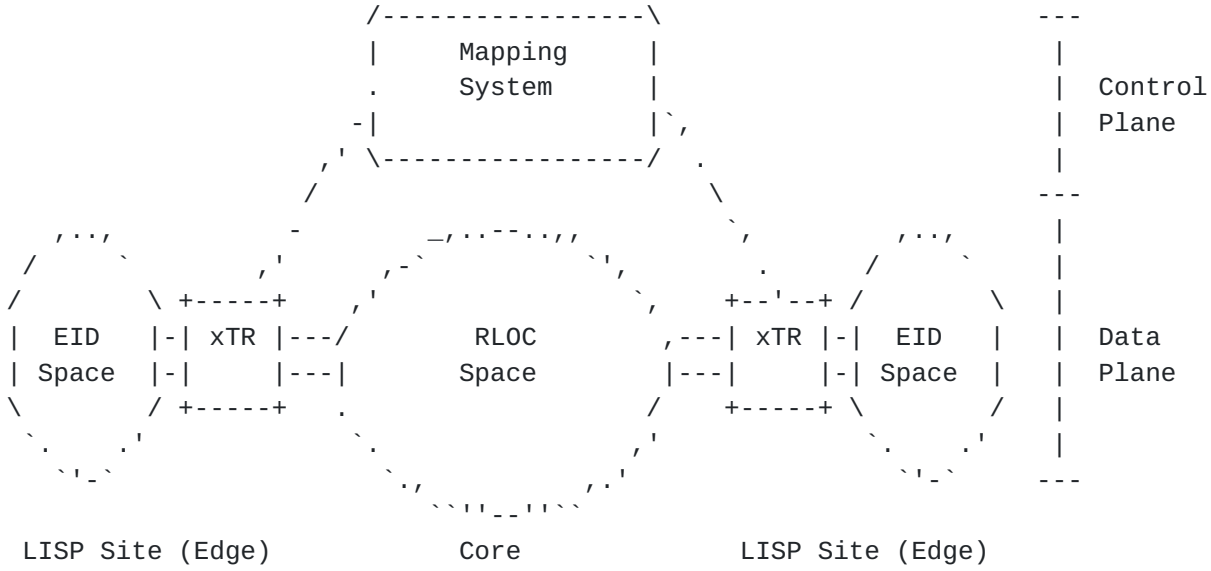


Figure 1.- A schema of the LISP Architecture

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 though as Provider Aggregatable addresses [RFC4116].

A publicly accessible and usually 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 can be thought as the equivalent of a DNS that would be accessed by ETRs to register mappings and by ITRs to retrieve them.

Finally, the LISP architecture has a strong emphasis in 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 typically require only a software upgrade. Additionally, LISP requires the deployment of an independent Mapping System, this distributed database is a new network entity.

In what follows we describe a simplified packet flow sequence between two nodes that are attached to LISP sites. Client hostA wants to send a packt to server hostB.

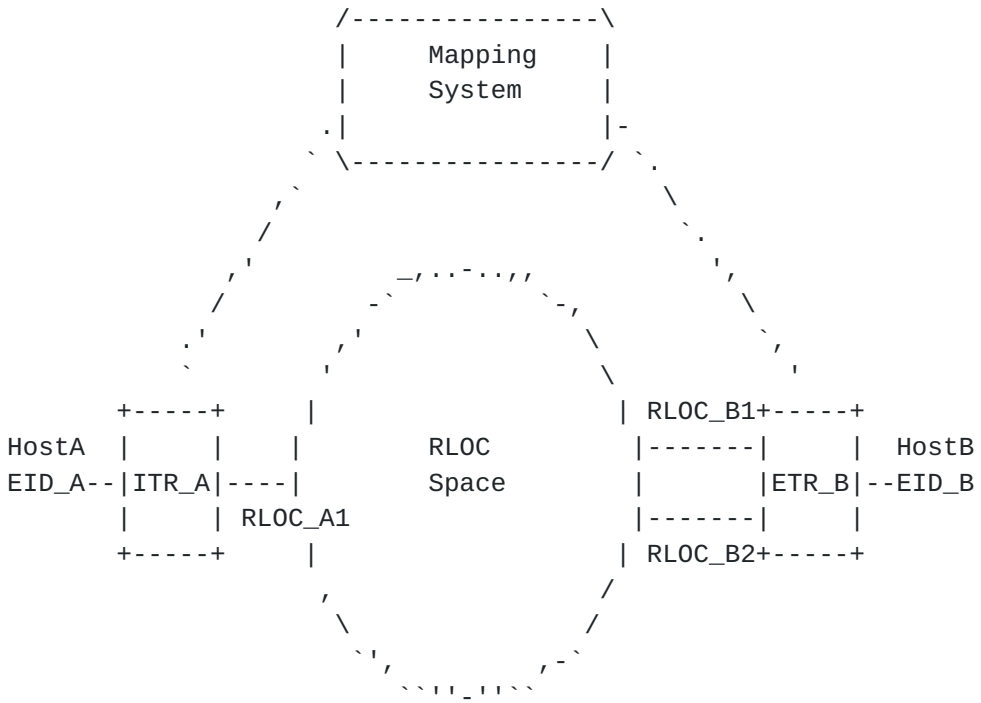


Figure 2.- Packet flow sequence in LISP

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. In order to do so it uses a LISP control message called Map-Request, the message contains EID_A 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 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 2.3.1.

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.

2.3. Data-Plane

This section describes the LISP data-plane, which is specified in [RFC6830]. The LISP data-plane is responsible of 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 viceversa (ETR). We first describe how packets are LISP-encapsulated and then we proceed to explain how ITRs cache forwarding state.

2.3.1. 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). The LISP header is prepended by ITRs and striped by ETRs. It carries reachability information (see more details in Section 3.2) and the Instance ID field. The Instance ID field is used to distinguish traffic that belongs to multiple tenants inside a LISP site, and that may use overlapped but logically separated addressing space.

Overall, LISP encapsulated data packets carry 4 headers [RFC6830] ("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 may contain reachability information 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 remains unchanged.

Finally and 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 an ETR removes a LISP header and then acts as an ITR to prepend another one. On the other hand, Recursive tunnels are nested tunnels and are implemented by using multiple LISP encapsulations on a packet.

2.3.2. 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- to increase the forwarding speed of subsequent packets addressed to the same EID prefix. Mappings include a (Time-to-Live) TTL (set by the ETR) and are expired according to this value, more details about the Map-Cache management can be found in Section 3.1.

2.4. Control-Plane

The LISP control-plane, specified in [RFC6833], provides a standard interface to register, query, and retrieve mappings. The LISP Mapping System, is a publicly accessible database that stores such mappings. In what follows we first describe the mappings, then the standard interface, and finally the Mapping System architecture.

2.4.1. 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. Such addresses are encoded using a general syntax called LISP Canonical Address Format (LCAF), specified in [I-D.ietf-lisp-lcaf]. The syntax is general enough to support encoding of IPv4 and IPv6 addresses and any other type of value.

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.

2.4.2. 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 low 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.

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 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.

2.4.3. Mapping System

LISP architecturally decouples control and data-plane by means of a standard interface. This interface glues the data-plane, routers responsible of forwarding data-packets, with the LISP Mapping System, a publicly accessible database responsible of 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 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 discussed for the Mapping System architecture the four main techniques available in distributed systems, namely: 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 [I-D.lear-lisp-nerd] and flat databases in the form of LISP-DHT [I-D.cheng-lisp-shdht],[I-D.mathy-lisp-dht]. Furthermore it is worth noting that, in some scenarios such as private deployments, the Mapping System can operate logically centralized. In such cases it is typically composed of a single Map-Server/Map-Resolver.

In what follows we focus on the two mapping systems that have been implemented and deployed (LISP-ALT and LISP+DDT).

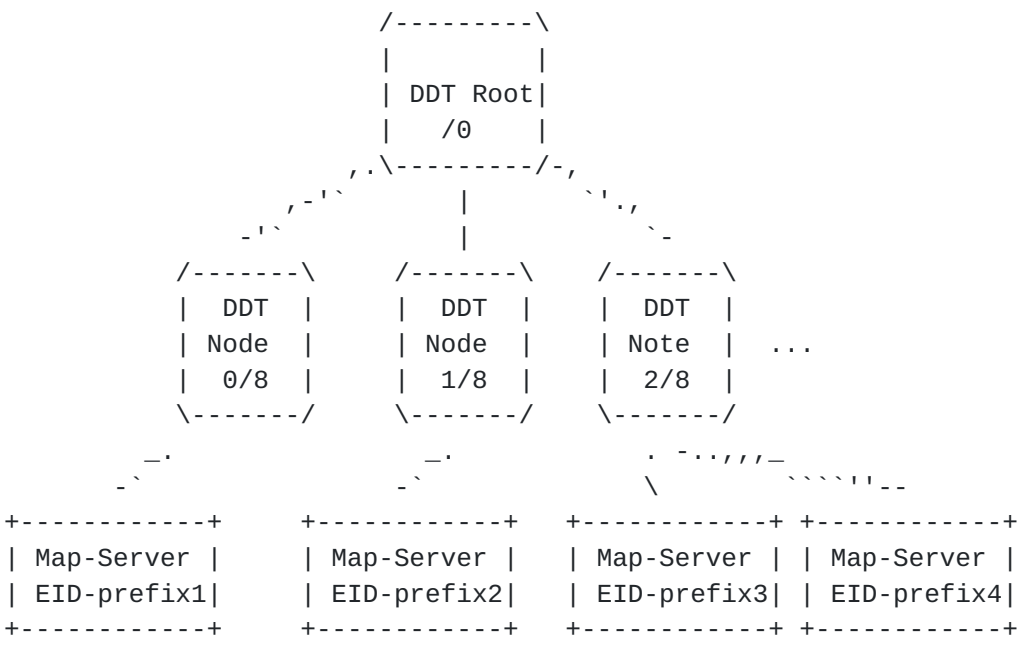
2.4.3.1. LISP+ALT

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. All the participating nodes connect to their peers through static tunnels. Every ETR involved in the ALT topology advertises its EID prefixes making the EID routable on the overlay.

When an ITR needs a mapping, it sends a Map-Request to a nearby ALT router. The ALT routers then forward the Map-Request on the overlay by inspecting their ALT routing tables. When the Map-Request reaches the ETR responsible for the mapping, a Map-Reply is generated and directly sent to the ITR's RLOC, without using the ALT overlay.

2.4.3.2. 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. The following figure presents a schematic representation of the DDT hierarchy.



Figre 3.- An schematic representation of the DDT tree structure, please note that the prefixes and the structure depitected 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 iteratively and in a similar way to the DNS. 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).

2.5. Internetworking Mechanisms

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

LISP defines two entities to provide internetworking:

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 data-packet, 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 Engress 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 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 allows to overcome such limitations by encapsulating packets over the network. Finally, the RLOC of PETRs must be statically configured in ITRs.

3. LISP Operational Mechanisms

In this section we detail the main operational mechanisms defined in LISP.

3.1. 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 could refresh the mapping by sending a new Map-Request. Typical values for TTL defined by LISP are 24h.

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 it a SMR to notify it that a new mapping is available.

3.2. 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 of 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 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 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:

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.

3.3. 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

3.4. 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 defienes two mechanisms:

Stateless: With this mechanism ITRs fragment packets that are too big, typically 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.

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

4. Mobility

LISP can also be used to enable mobility of devices not located in LISP networks. The problem with mobility of such devices is that their IP address changes whenever they change location, interrupting so flows.

To enable mobility on such devices, the device can implement the xTR functionality where the IP address presented to applications is an EID that never changes while the IP address obtained from the network is used by the xTR as RLOC. Packets are then transported on the network using the IP address assigned to the device by the visited network while at the application level IP addresses remain independent of the location of the device.

Whenever the device changes of RLOC, the ITR 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 signalling is strictly limited the Map-Server and to hosts with which communications are ongoing.

5. Multicast

LISP also supports multicast environments, the operational changes required to the multicast protocols are documented in [RFC6831].

In such scenarios, LISP creates multicast state both at the core and at the sites (both source and receiver). In order 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 that belongs to a LISP site transmits a PIM Join/ Prune message (S-EID,G) to join a multicast group.

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 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 outter 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.

6. 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 must be envisioned.

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 though carefully so to not overload the slow path and rate limiting should be used as specified in [RFC6830].

Care must also been 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 control informations, such as reachabilty 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-Server that strictly respect [RFC6833] and the lightweight authentication mechanism proposed by LISP-Sec [I-D.ietf-lisp-sec] is a possibility to reduce the risk. In more critical environments, stronger authentication may have to be used.

Packets are transported encapsulated with LISP meaning that devices on the path between an ITR (or PITR) and an ETR (or PETR) cannot correctly inspect the content of packets unless they implement methods to strip the headers added by LISP. Similarly, mappings enable triangular routing (i.e., packets of a flow cross different border routers depending on their direction) which means that intermediate boxes may have incomplete view on the traffic they inspect or manipulate.

More details about security implications of LISP can be found in [I-D.ietf-lisp-threats].

7. Use Cases

7.1. 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 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 the owner of 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 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 owner of the EID and 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.

7.2. LISP for IPv6 Transition

LISP encapsulations permits to transport packets using EIDs from a given address family (e.g., IPv6) with packets with addresses belonging to another address family (e.g., IPv4). The absence of correlation between the address family of RLOCs and EIDs makes LISP a candidate to ease the transition to IPv4.

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.

7.3. LISP for Network Virtualization

It is nowadays common to operate several virtual networks over the same physical infrastructure. The current approach usually rely on BGP/MPLS VPNs, where BGP is used to exchange routing information and MPLS to segregate packets of the different logical networks. This functionality could be achieved with LISP where the mappings and the mapping system are used instead of BGP and the LISP encapsulation is used to replace MPLS.

In virtual networks, it is essential to distinguish to 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 threat the packet.

Appart from the simplicity of managing mappings, the advantage of using LISP for virtual network is that it does not impose any requirement on the underlying network, except running IP.

7.4. 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 subnetworks where servers are hosted as forming the EID space. A LISP router is placed at the border between the backbone and each sub-network. When a virtual machine is moved to another subnetwork, it can (temporarily) keep the address of the sub-network it was hosted before the move so to allow ongoing communications to subsist. When a subnetwork detects the presence of a host with an address that does not belong to the subnetwork (e.g., via a message sent by the hypervisor), the LISP router of the new subnetwork registers the IP address of the virtual machine as an EID to the Map-Server of the subnetwork and associates its own address as RLOC.

To inform the other LISP routers that the machine moved and where, and then to avoid detours via the initial subnetwork, every Map-Server can listen on a predefined multicast address that is used as source address for Map-Register. As a result, the Map-Notify sent back by the Map-Server will be received by all the LISP routers that hence automatically learn the new location of the virtual machine.

8. Security Considerations

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

9. IANA Considerations

This memo includes no request to IANA.

10. Acknowledgements

To Do.

11. References

11.1. Normative References

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

[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.

[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.

[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.

11.2. 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.ermagan-lisp-nat-traversal] Ermagan, V., Farinacci, D., Lewis, D., Skriver, J., Maino, F., and C. White, "NAT traversal for LISP", draft-ermagan-lisp-nat-traversal-03 (work in progress), March 2013.

[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.

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

[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. The first Internet-Draft on LISP appeared in January, 2007, along with a LISP mailing list at the IETF.

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 undoubtly will for a long time to come).

A.1. Old LISP Models

LISP, as initilly conceived, had a number of potential operating modes, named 'models'. Although they are now obsolete, 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 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