Network Working Group
Internet-Draft
Intended status: Experimental
Expires: January 3, 2019

D. Farinacci
lispers.net
C. Cantrell
Nexus

July 2, 2018

A Decent LISP Mapping System (LISP-Decent)
draft-farinacci-lisp-decent-01

Abstract

This draft describes how the LISP mapping system designed to be distributed for scale can also be decentralized for management and trust.

Table of Contents

1. Introduction

The LISP architecture and protocols [RFC6830] introduces two new numbering spaces, Endpoint Identifiers (EIDs) and Routing Locators (RLOCs) which is intended to provide overlay network functionality. To map from EID to a set or RLOCs, a control-plane mapping system are used [RFC6836] [RFC8111]. These mapping systems are distributed in nature in their deployment for scalability but are centrally managed by a third- party entity, namely a Mapping System Provider (MSP). The entities that use the mapping system, such as data-plane xTRs, depend on and trust the MSP. They do not participate in the mapping system other than to register and retrieve information to/from the mapping system [RFC6833].

This document introduces a Decentralized Mapping System (DMS) so the xTRs can participate in the mapping system as well as use it. They can trust each other rather than rely on third-party infrastructure. The xTRs act as Map-Servers to maintain distributed state for scale and reducing attack surface.

2. Definition of Terms

Decentralized Mapping System (DMS): is a mapping system entity that is not third-party to the xTR nodes that use it. The xTRs themselves are part of the mapping system. The state of the mapping system is fully distributed, decentralized, and the trust relies on the xTRs that use and participate in their own mapping system.

Mapping System Provider (MSP): is an infrastructure service that deploys LISP Map-Resolvers and Map-Servers [RFC6833] and possibly ALT-nodes [RFC6836] or DDT-nodes [RFC8111]. The MSP can be managed by a separate organization other than the one that manages xTRs. This model provides a business separation between who manages and is responsible for the control-plane versus who manages the data-plane overlay service.

Peer-Group: is a set of Map-Servers which are joined to the same multicast group that send and receive Map-Register messages addressed to the multicast group. Map-Resolvers can use the peer-group to resolve mappings by sending Map-Requests to the multicast group or to any member of the peer-group. Map-Resolvers can do a mapping system lookup for the peer-group multicast address to obtain members of the peer-group.

Core Peer-Group: is a set of Map-Servers and Map-Resolvers who are joined to a multicast group to bootstrap a multi-layer decentralized mapping system.

Replication List Entry (RLE): is an RLOC-record format that contains a list of RLOCs that an ITR replicates multicast packets on a multicast overlay. The RLE format is specified in [RFC8060].

Group Address EID: is an EID-record format that contains IPv4 (0.0.0.0/0, G) or IPv6 (0::/0, G) state. This state is encoded as a Multicast Info Type LCAF specified in [RFC8060]. Members of a peer-group send Map-Registers for (0.0.0.0/0, G) or (0::/0, G) with an RLOC-record that RLE encodes its RLOC address. Details are specified in [I-D.ietf-lisp-signal-free-multicast].

3. Overview

The clients of the Decentralized Mapping System (DMS) are also the providers of mapping state. Clients are typically ETRs that Map-Register EID-to-RLOC mapping state to the mapping database system. ITRs are clients in that they send Map-Requests to the mapping database system to obtain EID-to-RLOC mappings that are cached for data-plane use. When xTRs participate in a DMS, they are also acting as Map-Resolvers and Map-Servers using the protocol machinery defined in LISP control-plane specifications [RFC6833], [I-D.ietf-lisp-sec], and [I-D.farinacci-lisp-ecdsa-auth]. The xTRs are not required to run the database mapping transport system protocols specified in [RFC6836] or [RFC8111].

The xTRs are organized in a peer-group. The peer-group is identified by an IPv4 or IPv6 multicast group address. The xTRs join the same multicast group and receive LISP control-plane messages addressed to

the group. Messages sent to the multicast group are distributed when the underlay network supports IP multicast [RFC6831] or is achieved with the overlay multicast mechanism described in [I-D.ietf-lisp-signal-free-multicast]. When overlay multicast is used and LISP Map-Register messages are sent to a peer-group, they are LISP data encapsulated with a instance-ID set to 0xffffff in the LISP header. The inner header of the encapsulated packet has the destination address set to the peer-group multicast group address and the outer header that is prepended has the destination address set to the RLOC of peer-group member. The members of the peer-group are kept in the LISP data-plane map-cache so packets for the peer-group can be replicated to each member RLOC.

All xTRs in a peer-group will store the same registered mappings and maintain the state as Map-Servers normally do. The peer-group members are not only receivers of the multicast group but also send packets to the group.

4. Components of a LISP-Decent xTR

When an xTR is configured to be a LISP-Decent xTR (or PxTR [RFC6832]), it runs the ITR, ETR, Map-Resolver, and Map-Server LISP network functions.

The following diagram shows 3 LISP-Decent xTRs joined to peer-group 224.1.1.1. When the ETR function of xTR1 originates a Map-Register, it is sent to all xTRs (including itself) synchronizing all 3 Map-Servers in xTR1, xTR2, and xTR3. The ITR function can populate its map-cache by sending a Map-Request locally to its Map-Resolver so it can replicate packets to each RLOC for EID 224.1.1.1.

```
                        xTR1
 Map-Request    +--------------------+
(always local)  |  +-----+  +-----+  |
   +---------------| ITR |  | ETR |-------------+
   |            |  +-----+  +-----+  |          |
   |            |                    |          |    Map-Register to EID
   |            |      +-------+     |          |  224.1.1.1 encapsulated to
   +------------------>| MR/MS |<---------------+  RLOCs xTR1, xTR2, and xTR3
                |      +-------+     |          |
                +--------------------+          |
                                                |
                           +--------------------+------------+
                           |                                 |
                           |                                 |
                +----------v---------+            +----------v---------+
                |     +--------+     |            |     +--------+     |
                |     |  MR/MS |     |            |     |  MR/MS |     |
                |     +--------+     |            |     +--------+     |
                |  +-----+  +-----+  |            |  +-----+  +-----+  |
                |  | ITR |  | ETR |  |            |  | ITR |  | ETR |  |
                |  +-----+  +-----+  |            |  +-----+  +-----+  |
                +--------------------+            +--------------------+
                         xTR2                              xTR3
```

Note if any external xTR would like to use a Map-Resolver from the peer-group, it only needs to have one of the LISP-Decent Map-Resolvers configured. By doing a looking to this Map-Resolver for EID 224.1.1,1, the external xTR could get the complete list of members for the peer-group.

For future study, an external xTR could multicast the Map-Request to 224.1.1.1 and either one of the LISP-Decent Map-Resolvers would

return a Map-Reply or the external xTR is prepared to receive multiple Map-Replies.

5. No LISP Protocol Changes

There are no LISP protocol changes required to support this LISP-Decent specification. However, an implementation that sends Map-Register messages to a multicast group versus a specific Map-Server unicast address must change to call the data-plane component so the ITR functionality in the node can encapsulate the Map-Register as a unicast packet to each member of the peer-group.

An ITR SHOULD lookup its peer-group address periodically to determine if the membership has changed. The ITR can also use the pubsub capability documented in [I-D.rodrigueznatal-lisp-pubsub] to be notified when a new member joins or leaves the peer-group.

6. Configuration and Authentication

When xTRs are joined to a multicast peer-group, they must have their site registration configuration consistent. Any policy or authentication key material must be configured correctly and consistently among all members. When [I-D.farinacci-lisp-ecdsa-auth] is used to sign Map-Register messages, public-keys can be registered to the peer-group using the site authentication key mentioned above or using a different authentication key from the one used for registering EID records.

7. Core Peer-Group

A core peer-group multicast address can be preconfigured to bootstrap the decentralized mapping system. The group address (or DNS name that maps to a group address) can be explicitly configured in a few xTRs to start building up the mappings. Then as other xTRs come online, they can add themselves to the core peer-group by joining the peer-group multicast group.

Alternatively or additionally, new xTRs can join a new peer-group multicast group to form another layer of a decentralized mapping system. The group address and members of this new layer peer-group would be registered to the core peer-group address and stored in the core peer-group mapping system. Note each mapping system layer could have a specific function or a specific circle of trust.

This multi-layer mapping system can be illustrated:

```
           __________               ---------
          /   core   \  224.2.2.2  / layer-1 \
         | peer-group | --------> |     I     |
         |  224.1.1.1 |           |    / \    |
          \__________/            |   J---K   |
               |                   \_________/
               | 224.3.3.3
               |
               v
           ---------
          / layer-2 \
         |     X     |
         |    / \    |
         |   Y---Z   |
          \_________/
```

```
Configured in xTRs A, B, and C (they make up the core peer-group):
  224.1.1.1 -> RLE: A, B, C

core peer-group DMS, mapping state in A, B, and C:
  224.2.2.2 -> RLE: I, J, K
  224.3.3.3 -> RLE: X, Y, Z

layer-1 peer-group DMS (inter-continental), mapping state in I, J, K:
   EID1 -> RLOCs: i(1), j(2)
   ...
   EIDn -> RLOCs: i(n), j(n)

layer-2 peer-group DMS (intra-continental), mapping sate in X, Y, Z::
   EIDa -> RLOCs: x(1), y(2)
   ...
   EIDz -> RLOCs: x(n), y(n)
```

The core peer-group multicast address 224.1.1.1 is configured in xTRs A, B and C so when each of them send Map-Register messages, they would all be able to maintain synchronized mapping state. Any EID can be registered to this DMS but in this example, peer-group multicast group EIDs are being registered only to find other peer-groups.

For example, lets say that xTR I boots up and it wants to find its other peers in its peer-group 224.2.2.2. Group address 224.2.2.2 is configured so xTR I knows what group to join for its peer-group. But xTR I needs a mapping system to register to, so the core peer-group is used and available to receive Map-Registers. The other xTRs J and

K in the peer-group do the same so when any of I, J or K needs to register EIDs, they can now send their Map-Register messages to group 224.2.2.2. Examples of EIDs being register are EID1 through EIDn shown above.

When Map-Registers are sent to group 224.2.2.2, they are encapsulated by the LISP data-plane by looking up EID 224.2.2.2 in the core peer-group mapping system. For the map-cache entry to be populated for 224.2.2.2, the data-plane must send a Map-Request so the RLOCs I, J, and K are cached for replication. To use the core peer-group mapping system, the data-plane must know of at least one of the RLOCs A, B, and/or C.

8. Security Considerations

Refer to the Security Considerations section of [I-D.ietf-lisp-rfc6833bis] for a complete list of security mechanisms as well as pointers to threat analysis drafts.

9. IANA Considerations

At this time there are no specific requests for IANA.

10. References

10.1. Normative References

[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>.

[RFC6831] Farinacci, D., Meyer, D., Zwiebel, J., and S. Venaas, "The Locator/ID Separation Protocol (LISP) for Multicast Environments", RFC 6831, DOI 10.17487/RFC6831, January 2013, <https://www.rfc-editor.org/info/rfc6831>.

[RFC6832] Lewis, D., Meyer, D., Farinacci, D., and V. Fuller, "Interworking between Locator/ID Separation Protocol (LISP) and Non-LISP Sites", RFC 6832, DOI 10.17487/RFC6832, January 2013, <https://www.rfc-editor.org/info/rfc6832>.

[RFC6833] Fuller, V. and D. Farinacci, "Locator/ID Separation Protocol (LISP) Map-Server Interface", RFC 6833, DOI 10.17487/RFC6833, January 2013, <https://www.rfc-editor.org/info/rfc6833>.

[RFC6836] Fuller, V., Farinacci, D., Meyer, D., and D. Lewis, "Locator/ID Separation Protocol Alternative Logical Topology (LISP+ALT)", RFC 6836, DOI 10.17487/RFC6836, January 2013, <https://www.rfc-editor.org/info/rfc6836>.

[RFC8060] Farinacci, D., Meyer, D., and J. Snijders, "LISP Canonical Address Format (LCAF)", RFC 8060, DOI 10.17487/RFC8060, February 2017, <https://www.rfc-editor.org/info/rfc8060>.

[RFC8111] Fuller, V., Lewis, D., Ermagan, V., Jain, A., and A. Smirnov, "Locator/ID Separation Protocol Delegated Database Tree (LISP-DDT)", RFC 8111, DOI 10.17487/RFC8111, May 2017, <https://www.rfc-editor.org/info/rfc8111>.

10.2. Informative References

[I-D.farinacci-lisp-ecdsa-auth]
Farinacci, D. and E. Nordmark, "LISP Control-Plane ECDSA Authentication and Authorization", draft-farinacci-lisp-ecdsa-auth-02 (work in progress), April 2018.

[I-D.ietf-lisp-rfc6833bis]
Fuller, V., Farinacci, D., and A. Cabellos-Aparicio, "Locator/ID Separation Protocol (LISP) Control-Plane", draft-ietf-lisp-rfc6833bis-10 (work in progress), March 2018.

[I-D.ietf-lisp-sec]
Maino, F., Ermagan, V., Cabellos-Aparicio, A., and D. Saucez, "LISP-Security (LISP-SEC)", draft-ietf-lisp-sec-15 (work in progress), April 2018.

[I-D.ietf-lisp-signal-free-multicast]
Moreno, V. and D. Farinacci, "Signal-Free LISP Multicast", draft-ietf-lisp-signal-free-multicast-09 (work in progress), March 2018.

[I-D.rodrigueznatal-lisp-pubsub]
Rodriguez-Natal, A., Ermagan, V., Leong, J., Maino, F., Cabellos-Aparicio, A., Barkai, S., Farinacci, D., Boucadair, M., Jacquenet, C., and S. Secci, "Publish/Subscribe Functionality for LISP", draft-rodrigueznatal-lisp-pubsub-02 (work in progress), March 2018.

Appendix A. Acknowledgments

The authors would like to thank the LISP WG for their review and acceptance of this draft.

The authors would also like to give a special thanks to Roman Shaposhnik for several discussions that occured before the first draft was published.

Appendix B. Document Change Log

[RFC Editor: Please delete this section on publication as RFC.]

B.1. Changes to draft-farinacci-lisp-decent-01

o Posted July 2018.

o Document timer and reference update.

B.2. Changes to draft-farinacci-lisp-decent-00

o Initial draft posted January 2018.

Authors' Addresses

Dino Farinacci
lispers.net
San Jose, CA
USA

Email: farinacci@gmail.com

Colin Cantrell
Nexus
Scottsdale, AZ
USA

Email: colin@nexus.io