Network Working Group
Internet-Draft
Intended status: Experimental
Expires: September 8, 2019

D. Farinacci
lispers.net
C. Cantrell
Nexus

March 7, 2019

A Decent LISP Mapping System (LISP-Decent)
draft-farinacci-lisp-decent-03

Abstract

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

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 https://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 September 8, 2019.

Copyright Notice

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

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.

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.

Pull-Based Mapping System: the mapping system is pull-based meaning that xTRs will lookup and register mappings by algorithmic transformation to locate which Map-Resolvers and Map-Servers are used. It is required that the lookup and registration uses a consistent algorithmic transformation function. Map-Registers are pushed to specific Map-Servers. Map-Requests are external lookups to Map-Resolvers on xTRs that do not participate in the mapping system and internal lookups when they do.

Modulus Value: this value is used in the Pull-Based Mapping System. It defines the number of map-server sets used for the mapping system. The modulus value is used to produce a Name Index used for a DNS lookup.

Name Index: this index value <index> is used in the Pull-Based Mapping System. For a mapping system that is configured with a map-server set of DNS names in the form of <name>.domain.com, the name index is prepended to <name> to form the lookup name <index>.<name>.domain.com. If the Modulus Value is 8, then the

name indexes are 0 through 7.

Hash Mask: The Hash Mask is used in the Pull-Based Mapping System. It is a mask value with 1 bits left justified. The mask is used to select what high-order bits of an EID-prefix is used in the hash function.

Push-Based Mapping System: the mapping system is push-based meaning that xTRs will push registrations via IP multicast to a group of Map-Servers and do local lookups acting as their own Map-Resolvers.

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]. RLEs are used with the Pushed-Based mapping system.

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 seed-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 [RFC8378].

Seed-Group: is a set of Map-Servers joined to a multicast group for the Push-Based Mapping system or are mapped by DNS names in a Pull-Based Mapping System. A core seed-group is used to bootstrap a set of LISP-Decent xTRs so they can learn about each other and use each other's mapping system service. A seed-group can be pull-based to bootstrap a push-based mapping system. That is, a set of DNS mapped map-servers can be used to join the mapping system's IP multicast group.

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.ietf-lisp-ecdsa-auth]. The xTRs are not required to run the database mapping transport system protocols specified in [RFC6836] or [RFC8111].

This document will describe two decentralized and distributed mapping system mechanisms. A Push-Based Mapping System uses IP multicast so xTRs can find each other by locally joining an IP multicast group. A Pull-Based Mapping System uses DNS with an algorithmic transformation function so xTRs can find each other.

4. Push-Based Mapping System

The xTRs are organized in a mapping-system group. The group is identified by an IPv4 or IPv6 multicast group address or using a pull-based approach in described in Section 5. When using multicast, 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 [RFC8378]. When overlay multicast is used and LISP Map-Register messages are sent to the 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 multicast group address and the outer header that is prepended has the destination address set to the RLOC of mapping system member. The members of the mapping system group are kept in the LISP data-plane map-cache so packets for the group can be replicated to each member RLOC.

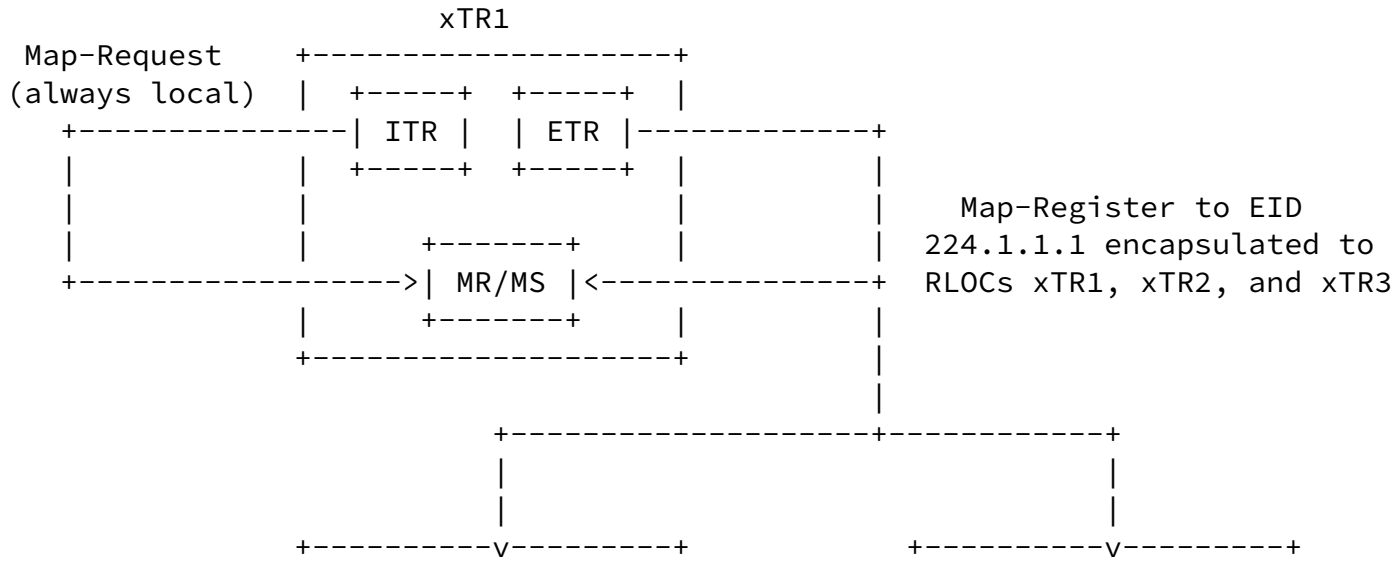
All xTRs in a mapping system group will store the same registered mappings and maintain the state as Map-Servers normally do. The members are not only receivers of the multicast group but also send

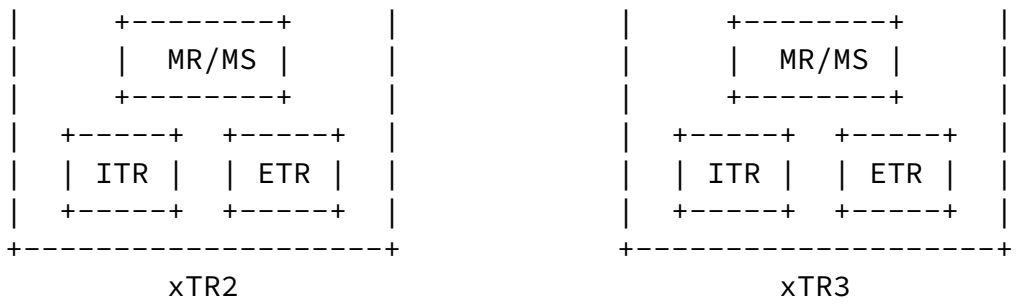
packets to the group.

4.1. Components of a Pushed-Based 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 mapping system 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.





Note if any external xTR would like to use a Map-Resolver from the mapping system 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 mapping system 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.

4.2. No LISP Protocol Changes

There are no LISP protocol changes required to support the push-based LISP-Decent set of procedures. 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 mapping system group.

An ITR SHOULD lookup its mapping system group address periodically to determine if the membership has changed. The ITR can also use the

pubsub capability documented in [I-D.ietf-lisp-pubsub] to be notified when a new member joins or leaves the multicast group.

4.3. Configuration and Authentication

When xTRs are joined to a multicast 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.ietf-lisp-ecdsa-auth] is used to sign Map-Register messages, public-keys can be registered to the mapping system group using the site authentication key mentioned above or using a different authentication key from the one used for registering EID records.

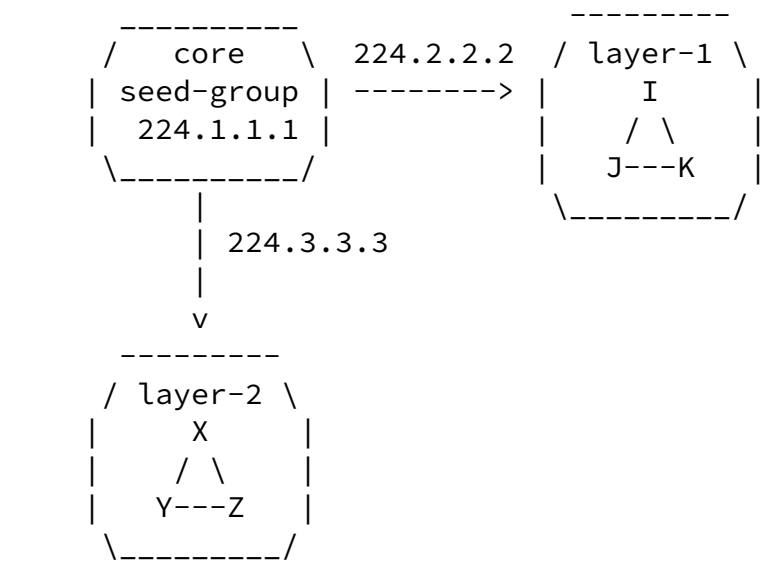
4.4. Core Seed-Group

A core seed-group can be discovered using a multicast group in a push-based system or a Map-Server set of DNS names in a pull-based system (see Section 5 for details).

When using multicast for the mapping system group, a core seed-group multicast group 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 registrations. Then as other xTRs come online, they can add themselves to the core seed-group by joining the seed-group multicast group.

Alternatively or additionally, new xTRs can join a new mapping system multicast group to form another layer of a decentralized mapping system. The group address and members of this new layer seed-group would be registered to the core seed-group address and stored in the core seed-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:



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

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

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

The core seed-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, seed-group multicast group EIDs are being registered only to find other mapping system groups.

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

Registers. The other xTRs J and K in the mapping system 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 seed-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 seed-group mapping system, the data-plane must know of at least one of the RLOCs A, B, and/or C.

5. Pull-Based Mapping System

5.1. Components of a Pulled-Based 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.

Unlike the Push-Based Mapping System, the xTRs do not need to be organized by joining a multicast group. In a Pull-Based Mappig System, a hash function over an EID is used to identify which xTR is used as the Map-Resolver and Map-Server. The Domain Name System (DNS) [RFC1034] [RFC1035] is used as a resource discovery mechanism.

The RLOC addresses of the xTRs will be A and AAAA records for DNS names that map algorithmically from the hash of the EID. A SHA-256 hash function [RFC6234] over the following ASCII formatted EID string is used:

```
[<iid>]<eid>/<ml>
[<iid>]<group>/<gml>-<source>/<sml>
```

Where <iid> is the instance-ID and <eid> is the EID of any EID-type defined in [RFC8060]. And then the Modulus Value <mv> is used to produce the Name Index <index> used to build the DNS lookup name:

```
eid = "[<iid>]<eid>/<ml>"
index = hash.sha_256(eid) MOD mv
```

The Hash Mask is used to select what bits are used in the SHA-256 hash function. This is required to support longest match lookups in the mapping system. The same map-server set needs to be selected when looking up a more-specific EID found in the Map-Request message

with one that could match a less-specific EID-prefix registered and found in the Map-Register message. For example, if an EID-prefix

[0]240.0.1.0/24 is registered to the mapping system and EID [0]240.0.1.1/32 is looked up to match the registered prefix, a Hash Mask of 8 bytes can be used to AND both the /32 or /24 entries to produce the same hash string bits of "[0]240.0".

For (*,G) and (S,G) multicast entries in the mapping system, the hash strings are:

```
sg-eid = "[<iid>]<group>/<gml>-<source>/<sml>"
index = hash.sha_256(sg-eid) MOD mv

starg-eid = "[<iid>]<group>/<gml>-0.0.0.0/0"
index = hash.sha_256(starg-eid) MOD mv
```

The Hash Mask MUST include the string "[<iid>]<group>" and not string <source>. So when looking up [0](2.2.2.2, 224.1.1.1) that will match a (*, 224.1.1.1/32), the hash string produced with a Hash Mask of 12 bytes is "[0]224.1.1.1".

When the <index> is computed from a unicast or multicast EID, the DNS lookup name becomes:

```
<index>.map-server.domain.com
```

When an xTR does a DNS lookup on the lookup name, it will send Map-Register messages to all A and AAAA records for EID registrations. For Map-Request messages, xTRs MAY round robin EID lookup requests among the A and AAAA records.

5.2. Deployment Example

Here is an example deployment of a pull-based model. Let's say 4 map-server sets are provisioned for the mapping system. Therefore 4 distinct DNS names are allocated and a Modulus Value 4 is used. Each DNS name is allocated Name Index 0 through 3:

```
0.map-server.lispers.net
1.map-server.lispers.net
2.map-server.lispers.net
```

3.map-server.lispers.net

The A records for each name can be assigned as:


```
0.map-server.lispers.net:
    A <rloc1-att>
    A <rloc2-verizon>
1.map-server.lispers.net:
    A <rloc1-bt>
    A <rloc2-dt>
2.map-server.lispers.net:
    A <rloc1-cn>
    A <rloc2-kr>
3.map-server.lispers.net:
    A <rloc1-au>
    A <rloc2-nz>
```

When an xTR wants to register "[1000]fd::2222", it hashes the EID string to produce, for example, hash value 0x66. Using the modulus value 4 (0x67 & 0x3) produces index 0x3, so the DNS name 3.map-server.lispers.net is used and a Map-Regiter is sent to <rloc1-au> and <rloc2-nz>.

Note that the pull-based method can be used for a core seed-group for bootstraping a push-based mapping system where multicast groups are registered.

5.3. Management Considerations

There are no LISP protocol changes required to support the pull-based LISP-Decent set of procedures. However, an implementation SHOULD do periodic DNS lookups to determine if A records have changed for a DNS entry.

When xTRs derive Map-Resolver and Map-Server names from the DNS, they need to use the same Modulus Value otherwise some xTRs will lookup

EIDs to the wrong place they were registered.

The Modulus Value can be configured or pushed to the LISP-Decent xTRs. A future version of this document will describe a push mechanism so all xTRs use a consistent modulus value.

6. 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.

7. IANA Considerations

At this time there are no specific requests for IANA.

8. References

8.1. Normative References

[RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, DOI 10.17487/RFC1034, November 1987, <https://www.rfc-editor.org/info/rfc1034>.

[RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, DOI 10.17487/RFC1035, November 1987, <https://www.rfc-editor.org/info/rfc1035>.

[RFC6234] Eastlake 3rd, D. and T. Hansen, "US Secure Hash Algorithms (SHA and SHA-based HMAC and HKDF)", RFC 6234, DOI 10.17487/RFC6234, May 2011, <https://www.rfc-editor.org/info/rfc6234>.

[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>.

[RFC8378] Moreno, V. and D. Farinacci, "Signal-Free Locator/ID Separation Protocol (LISP) Multicast", RFC 8378, DOI 10.17487/RFC8378, May 2018, <https://www.rfc-editor.org/info/rfc8378>.

8.2. Informative References

[I-D.ietf-lisp-ecdsa-auth] Farinacci, D. and E. Nordmark, "LISP Control-Plane ECDSA

Authentication and Authorization", draft-ietf-lisp-ecdsa-auth-00 (work in progress), September 2018.

[I-D.ietf-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-ietf-lisp-pubsub-02 (work in progress), November 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-24 (work in progress), February 2019.

[I-D.ietf-lisp-sec]
Maino, F., Ermagan, V., Cabellos-Aparicio, A., and D. Saucez, "LISP-Security (LISP-SEC)", draft-ietf-lisp-sec-17 (work in progress), November 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-03

- Posted March 2019.
- Introduce the Hash Mask which is used to grab common bits from a registered prefix and a lookup prefix.

- Spec how multicast lookups are done in the pull-based mapping system.

- Indicate the hash string includes the unicast EID mask-length and multicast group and source mask-lengths.

B.2. Changes to draft-farinacci-lisp-decent-02

- Posted November 2018.

- Changed references from peer-group to seed-group to make the algorithms in this document more like how blockchain networks initialize the peer-to-peer network.

- Added pull mechanism to compliment the push mechanism. The pull mechanism could be used as a seed-group to bootstrap the push mechanism.

B.3. Changes to draft-farinacci-lisp-decent-01

- Posted July 2018.

- Document timer and reference update.

B.4. Changes to draft-farinacci-lisp-decent-00

- Initial draft posted January 2018.

Authors' Addresses

Dino Farinacci
lispers.net
San Jose, CA
USA

Email: farinacci@gmail.com

Colin Cantrell
Nexus
Tempe, AZ
USA

Email: colin@nexus.io