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 mod p=I |  I -->  |            | p,g,I  |  e   |
+------|--------|------------|---------|------------|--------|------+
|  i   | p,g,I  |            |  <-- E  |g^e mod p=E |  p,g   |  e   |
+------|--------|------------|---------|------------|--------|------+
| i,s  |p,g,I,E |E^i mod p=s |         |I^e mod 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, an 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.

```
     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
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  / |       Source Port = xxxx      |       Dest Port = 4341        |
UDP +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  \ |           UDP Length         |        UDP Checksum           |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
L   |N|L|E|V|I|P|K|K|            Nonce/Map-Version                  |
I \ +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
S / |                 Instance ID/Locator-Status-Bits               |
P   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

     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 exhange 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 adminstrator. Optionally, the ITR can send RLOC-probes to the compromised RLOC to determine if can reach the authoriative 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-Secuirty (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 pubic 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: Informational
Expires: October 8, 2016

L. Iannone
Telecom ParisTech
R. Jorgensen
Bredbandsfylket Troms
D. Conrad
Virtualized, LLC
G. Huston
APNIC - Asia Pacific Network
Information Center
April 6, 2016

LISP EID Block Management Guidelines
draft-ietf-lisp-eid-block-mgmnt-07.txt

Abstract

This document proposes a framework for the management of the LISP EID Address Block. The framework described relies on hierarchical distribution of the address space, granting temporary usage of 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 October 8, 2016.

Copyright Notice

Copyright (c) 2016 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 3
2. Introduction 3
3. Definition of Terms 3
4. EID Prefix Registration Policy 3
5. EID Prefixes Registration Requirements 4
6. EID Prefix Request Template 5
7. Policy Validity Period 6
8. Security Considerations 7
9. IANA Considerations 7
10. Procedures to be followed by RIPE NCC 8
11. Acknowledgments 8
12. References 9
12.1. Normative References 9
12.2. Informative References 9
Appendix A. Document Change Log 10
Authors' Addresses 12

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]) separate 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 organization 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]), but assumes that the reader is familiar with the LISP terminology. [I-D.ietf-lisp-introduction] provides an introduction to the LISP technology, including its terminology. .

4. EID Prefix Registration Policy

The request for 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 MUST be renewed on a regular basis to ensure their use by active participants in the experiment. The registration period is 12 months. A renewal SHOULD NOT cause a change in the EID prefix registered in the previous request. The conditions of registration renewal are the same as the conditions of first EID prefix registration request.

3. It is preferable not to reuse EID prefixes whose registration is expired. 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. 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., [RFC7481], WHOIS, HTTP, or similar access methods).

3. The registry operator MUST permit the delegation of EID prefixes in the reverse DNS space to holders of registered EID prefixes.

4. 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 the case of individuals requesting an EID prefix this section can be left empty)

   (a) Organization Name

   (b) Organization Address

   (c) Organization Phone

   (d) Organization WebSite

2. Contact Person (Mandatory)

   (a) Name

   (b) Address

   (c) Phone

   (d) Fax (optional)

   (e) Email

3. EID Prefix Request (Mandatory)

   (a) Prefix Size

      + Expressed as an address prefix length.

(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 MMMM/YYYY3.

+ All registrations may be renewed by the applicant for further 12 month periods, ending on MMMM/YYYY3.

+ According to the 3+3 year experimentation plan, defined in [I-D.ietf-lisp-eid-block], all registrations MUST end by MMMM/YYYY3, 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 MMMM/YYYY6 and a new policy (to be decided - see Section 7) will apply afterwards.

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 MMMM/YYYY3.

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 MMMM/YYYY6) so as 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 after MMMM/YYYY6.

[RFC Editor: please replace MMMM and all its occurrences in the document with the month of publication of [I-D.ietf-lisp-eid-block] as RFC.]

[RFC Editor: please replace YYYY0 and all its occurrences in the document with the year of publication of [I-D.ietf-lisp-eid-block] as RFC.]

[RFC Editor: please replace YYYY3 and all its occurrences in the document with the year of publication of [I-D.ietf-lisp-eid-block] as RFC plus 3 years, e.g., if published in 2016 then put 2019.]

[RFC Editor: please replace YYYY6 and all its occurrences in the document with the year of publication of [I-D.ietf-lisp-eid-block] as RFC plus 6 years, e.g., if published in 2016 then put 2022.]

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. IANA Considerations

IANA allocated the following IPv6 address block for experimental use as LISP EID prefix [I-D.ietf-lisp-eid-block]:

- o Address Block: 2001:5::/32
- o Name: EID Space for LISP
- o RFC: [I-D.ietf-lisp-eid-block]
- o Further Details at: http://www.iana.org/assignments/iana-ipv6-special-registry/iana-ipv6-special-registry.xhtml

In order to grant requesting organisations and individuals exclusive use of EID prefixes out of such reserved block (limited to the duration of the LISP experiment as outlined in Section 7) there is an operational requirement for an EID registration service.

Provided that the policies and requirements outlined in Section 4, Section 5, and Section 6 are respected, EID prefix registration is accorded based on a "First Come First Served" basis.

There is no hard limit in the number of registrations an organization or individual can submit as long as information described in Section 6 is provided, in particular point 4: "Experiment Description".

For the duration defined in [I-D.ietf-lisp-eid-block] RIPE NCC will manage the the LISP EID prefix as described herein. Therefore, this document has no IANA actions.

10. Procedures to be followed by RIPE NCC

RIPE NCC will provide the registration service following the EID Prefix Registration Policy (Section 4) and the EID Prefix Registration Requirements (Section 5) provided in this document. The request form provided by RIPE NCC will include at least the information from the template in Section 6. RIPE NCC will make publicly available all received requests. While this document does not suggests any minimum allocation size, RIPE NCC is allowed to introduce such minimum size for management purposes.

11. Acknowledgments

Thanks to A. Retana, J. Arko, P. Yee, A. de la Haye, A. Cima, A Pawlik, J. Curran, A. Severin, B. Haberman, T. Manderson, D. Lewis, D. Farinacci, M. Binderberger, D. Saucez, E. Lear, 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.

12. References

12.1. 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-13 (work in progress), February 2016.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, DOI 10.17487/RFC5226, May 2008, <http://www.rfc-editor.org/info/rfc5226>.

12.2. Informative References

[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-13 (work in progress), April 2015.

[RFC2860] Carpenter, B., Baker, F., and M. Roberts, "Memorandum of Understanding Concerning the Technical Work of the Internet Assigned Numbers Authority", RFC 2860, DOI 10.17487/RFC2860, June 2000, <http://www.rfc-editor.org/info/rfc2860>.

[RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, DOI 10.17487/RFC6830, January 2013, <http://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, <http://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, <http://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, <http://www.rfc-editor.org/info/rfc6833>.

[RFC6834] Iannone, L., Saucez, D., and O. Bonaventure, "Locator/ID Separation Protocol (LISP) Map-Versioning", RFC 6834, DOI 10.17487/RFC6834, January 2013, <http://www.rfc-editor.org/info/rfc6834>.

[RFC6835] Farinacci, D. and D. Meyer, "The Locator/ID Separation Protocol Internet Groper (LIG)", RFC 6835, DOI 10.17487/RFC6835, January 2013, <http://www.rfc-editor.org/info/rfc6835>.

[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, <http://www.rfc-editor.org/info/rfc6836>.

[RFC6837] Lear, E., "NERD: A Not-so-novel Endpoint ID (EID) to Routing Locator (RLOC) Database", RFC 6837, DOI 10.17487/RFC6837, January 2013, <http://www.rfc-editor.org/info/rfc6837>.

[RFC7020] Housley, R., Curran, J., Huston, G., and D. Conrad, "The Internet Numbers Registry System", RFC 7020, DOI 10.17487/RFC7020, August 2013, <http://www.rfc-editor.org/info/rfc7020>.

[RFC7481] Hollenbeck, S. and N. Kong, "Security Services for the Registration Data Access Protocol (RDAP)", RFC 7481, DOI 10.17487/RFC7481, March 2015, <http://www.rfc-editor.org/info/rfc7481>.

Appendix A. Document Change Log

Version 07 Posted April 2016.

- o Addressed editorial issues raised in Gen-Art review by Peter Yee.

- o Removed "Definition of Terms" section as suggested by Peter Yee in the Gen-Art review.

- o Section "IANA Considerations" has been re-written to fix issue raised by IESG, IANA, and P. Yee.

- o Deleted bullet allowing multiple operators in the requirements section. Due to the limited duration of the experiment one single registration operator (RIPE) is sufficient.

- o Modified the dates, introducing variables, so to allow RFC Editor to easily update dates by publication as RFC.

Version 06 Posted August 2015.

- o Fixed Authors addresses and typo in section 10.

Version 05 Posted July 2015.

- o Added explicit text about RIPE NCC providing the registration service during the temporary experiment.

Version 04 Posted December 2014.

- o Added two clarification sentences to address the comments of E. Lear and D. Saucez during WG LC.

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-mgmnt-03.txt.

Authors' Addresses

Luigi Iannone
Telecom ParisTech
France

Email: ggx@gigix.net

Roger Jorgensen
Bredbandsfylket Troms
Norway

Email: rogerj@gmail.com

David Conrad
Virtualized, LLC
USA

Email: drc@virtualized.org

Geoff Huston
APNIC - Asia Pacific Network Information Center
Australia

Email: gih@apnic.net

Network Working Group
Internet-Draft
Intended status: Informational
Expires: August 1, 2016

D. Saucez
INRIA
L. Iannone
Telecom ParisTech
O. Bonaventure
Universite catholique de Louvain

January 29, 2016

LISP Threats Analysis
draft-ietf-lisp-threats-15.txt

Abstract

This document provides a threat analysis of the Locator/Identifier Separation Protocol (LISP).

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 August 1, 2016.

Copyright Notice

Copyright (c) 2016 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. Threat model 3
2.1. Attacker's Operation Modes 4
2.1.1. On-path vs. Off-path Attackers 4
2.1.2. Internal vs. External Attackers 4
2.1.3. Live vs. Time-shifted attackers 5
2.1.4. Control-plane vs. Data-plane attackers 5
2.1.5. Cross mode attackers 5
2.2. Threat categories 5
2.2.1. Replay attack 5
2.2.2. Packet manipulation 6
2.2.3. Packet interception and suppression 6
2.2.4. Spoofing 6
2.2.5. Rogue attack 7
2.2.6. Denial of Service (DoS) attack 7
2.2.7. Performance attack 7
2.2.8. Intrusion attack 7
2.2.9. Amplification attack 7
2.2.10. Passive Monitoring Attacks 8
2.2.11. Multi-category attacks 8
3. Attack vectors 8
3.1. Gleaning 8
3.2. Locator Status Bits 9
3.3. Map-Version 10
3.4. Routing Locator Reachability 11
3.5. Instance ID 12
3.6. Interworking 12
3.7. Map-Request messages 12
3.8. Map-Reply messages 13
3.9. Map-Register messages 15
3.10. Map-Notify messages 15
4. Note on Privacy 15
5. Threats Mitigation 16
6. Security Considerations 17
7. IANA Considerations 17
8. Acknowledgments 17
9. References 17
9.1. Normative References 17
9.2. Informative References 18
Appendix A. Document Change Log (to be removed on publication) . 19
Authors' Addresses 21

1. Introduction

The Locator/ID Separation Protocol (LISP) is specified in [RFC6830]. This document provides an assessment of the potential security threats for the current LISP specifications if LISP is deployed in the Internet (i.e., a public non-trustable environment).

The document is composed of three main parts: the first defines a general threat model that attackers use to mount attacks. The second part, using this threat model, describes the techniques based on the LISP protocol and LISP architecture that attackers may use to construct attacks. The third part discusses mitigation techniques and general solutions to protect the LISP protocol and architecture from attacks.

This document does not consider all the possible uses of LISP as discussed in [RFC6830] and [RFC7215] and does not cover threats due to specific implementations. The document focuses on LISP unicast, including as well LISP Interworking [RFC6832], LISP Map-Server [RFC6833]), and LISP Map-Versioning [RFC6834]. Additional threats may be discovered in the future while deployment continues. The reader is assumed to be familiar with these documents for understanding the present document.

This document assumes a generic IP service and does not discuss the difference, from a security viewpoint, between using IPv4 or IPv6.

2. Threat model

This document assumes that attackers can be located anywhere in the Internet (either in LISP sites or outside LISP sites) and that attacks can be mounted either by a single attacker or by the collusion of several attackers.

An attacker is a malicious entity that performs the action of attacking a target in a network where LISP is (partially) deployed by leveraging the LISP protocol and/or architecture.

An attack is the action of performing an illegitimate action on a target in a network where LISP is (partially) deployed.

The target of an attack is the entity (i.e., a device connected to the network or a network) that is aimed to undergo the consequences of an attack. Other entities can potentially undergo side effects of an attack, even though they are not directly targeted by the attack. The target of an attack can be selected specifically, i.e., a particular entity, or arbitrarily, i.e., any entity. Finally, an

attacker can aim at attacking one or several targets with a single attack.

Section 2.1 specifies the different modes of operation that attackers can follow to mount attacks and Section 2.2 specifies the different categories of attacks that attackers can build.

2.1. Attacker's Operation Modes

In this document attackers are classified according to their modes of operation, i.e., the temporal and spacial diversity of the attacker. These modes are not mutually exclusive, they can be used by attackers in any combination, and other modes may be discovered in the future. Further, attackers are not at all bound by our classification scheme, so implementers and those deploying will always need to do additional risk analysis for themselves.

2.1.1. On-path vs. Off-path Attackers

On-path attackers, also known as Men-in-the-Middle, are able to intercept and modify packets between legitimate communicating entities. On-path attackers are located either directly on the normal communication path (either by gaining access to a node on the path or by placing themselves directly on the path) or outside the location path but manage to deviate (or gain a copy of) packets sent between the communication entities. On-path attackers hence mount their attacks by modifying packets initially sent legitimately between communication entities.

An attacker is called off-path attacker if it does not have access to packets exchanged during the communication or if there is no communication. In order for their attacks to succeed, off-path attackers must hence generate packets and inject them in the network.

2.1.2. Internal vs. External Attackers

An internal attacker launches its attack from a node located within a legitimate LISP site. Such an attacker is either a legitimate node of the site or it exploits a vulnerability to gain access to a legitimate node in the site. Because of their location, internal attackers are trusted by the site they are in.

On the contrary, an external attacker launches its attacks from the outside of a legitimate LISP site.

2.1.3. Live vs. Time-shifted attackers

A live attacker mounts attacks for which it must remain connected as long as the attack is mounted. In other words, the attacker must remain active for the whole duration of the attack. Consequently, the attack ends as soon as the attacker (or the used attack vector) is neutralized.

On the contrary, a time-shifted attacker mounts attacks that remain active after it disconnects from the Internet.

2.1.4. Control-plane vs. Data-plane attackers

A control-plane attacker mounts its attack by using control-plane functionalities, typically the mapping system.

A data-plane attacker mounts its attack by using data-plane functionalities.

As there is no complete isolation between the control-plane and the data-plane, an attacker can operate in the control-plane (or data-plane) to mount attacks targeting the data-plane (or control-plane) or keep the attacked and targeted planes at the same layer (i.e., from control-plane to control-plane or from data-plane to data-plane).

2.1.5. Cross mode attackers

The attacker modes of operation are not mutually exclusive and hence attackers can combine them to mount attacks.

For example, an attacker can launch an attack using the control-plane directly from within a LISP site to which it is able to get temporary access (i.e., internal + control-plane attacker) to create a vulnerability on its target and later on (i.e., time-shifted + external attacker) mount an attack on the data plane (i.e., data-plane attacker) that leverages the vulnerability.

2.2. Threat categories

Attacks can be classified according to the nine following categories. These categories are not mutually exclusive and can be used by attackers in any combination.

2.2.1. Replay attack

A replay attack happens when an attacker retransmits at a later time, and without modifying it, a packet (or a sequence of packets) that

has already been transmitted.

2.2.2. Packet manipulation

A packet manipulation attack happens when an attacker receives a packet, modifies the packet (i.e., changes some information contained in the packet) and finally transmits the packet to its final destination that can be the initial destination of the packet or a different one.

2.2.3. Packet interception and suppression

In a packet interception and suppression attack, the attacker captures the packet and drops it before it can reach its final destination.

2.2.4. Spoofing

With a spoofing attack, the attacker injects packets in the network pretending to be another node. Spoofing attacks are made by forging source addresses in packets.

It should be noted that with LISP, packet spoofing is similar to spoofing with any other existing tunneling technology currently deployed in the Internet. Generally the term "spoofed packet" indicates a packet containing a source IP address that is not the actual originator of the packet. Hence, since LISP uses encapsulation, the spoofed address could be in the outer header as well as in the inner header, this translates to two types of spoofing.

Inner address spoofing: the attacker uses encapsulation and uses a spoofed source address in the inner packet. In case of data-plane LISP encapsulation, that corresponds to spoofing the source EID (End-point IDentifier) address of the encapsulated packet.

Outer address spoofing: the attacker does not use encapsulation and spoofs the source address of the packet. In case of data-plane LISP encapsulation, that corresponds to spoofing the source RLOC (Routing LOCator) address of the encapsulated packet.

Note that the two types of spoofing are not mutually exclusive, rather all combinations are possible and could be used to perform different kinds of attacks. For example, an attacker outside a LISP site can generate a packet with a forged source IP address (i.e., outer address spoofing) and forward it to a LISP destination. The packet is then eventually encapsulated by a PITR (Proxy Ingress

Tunnel Router) so that once encapsulated the attack corresponds to a inner address spoofing. One can also imagine an attacker forging a packet with encapsulation where both inner and outer source addresses are spoofed.

It is important to note that the combination of inner and outer spoofing makes the identification of the attacker complex as the packet may not contain information that allows to detect the origin of the attack.

2.2.5. Rogue attack

In a rogue attack the attacker manages to appear as a legitimate source of information, without faking its identity (as opposed to a spoofing attacker).

2.2.6. Denial of Service (DoS) attack

A Denial of Service (DoS) attack aims at disrupting a specific targeted service to make it unable to operate properly.

2.2.7. Performance attack

A performance attacks aims at exploiting computational resources (e.g., memory, processor) of a targeted node so as to make it unable to operate properly.

2.2.8. Intrusion attack

In an intrusion attack, the attacker gains remote access to a resource (e.g., a host, a router, or a network) or information that it legitimately should not have access. Intrusion attacks can lead to privacy leakages.

2.2.9. Amplification attack

In an amplification attack, the traffic generated by the target of the attack in response to the attack is larger than the traffic that the attacker must generate.

In some cases, the data-plane can be several orders of magnitude faster than the control-plane at processing packets. This difference can be exploited to overload the control-plane via the data-plane without overloading the data-plane.

2.2.10. Passive Monitoring Attacks

An attacker can use pervasive monitoring, which is a technical attack [RFC7258], targeting information about LISP traffic that may or not be used to mount other type of attacks.

2.2.11. Multi-category attacks

Attacks categories are not mutually exclusive and any combination can be used to perform specific attacks.

For example, one can mount a rogue attack to perform a performance attack starving the memory of an ITR (Ingress Tunnel Router) resulting in a DoS (Denial-of-Service) on the ITR.

3. Attack vectors

This section presents attack techniques that may be used by attackers when leveraging the LISP protocol and/or architecture.

3.1. Gleaning

To reduce the time required to obtain a mapping, the optional gleaning mechanism defined for LISP allows an xTR (Ingress and/or Egress Tunnel Router) to directly learn a mapping from the LISP data encapsulated packets and the Map-Request packets that it receives. LISP encapsulated data packets contain a source RLOC, destination RLOC, source EID and destination EID. When an xTR receives an encapsulated data packet coming from a source EID for which it does not already know a mapping, it may insert the mapping between the source RLOC and the source EID in its EID-to-RLOC Cache. The same technique can be used when an xTR receives a Map-Request as the Map-Request also contains a source EID address and a source RLOC. Once a gleaned entry has been added to the EID-to-RLOC cache, the xTR sends a Map-Request to retrieve the actual mapping for the gleaned EID from the mapping system.

If a packet injected by an off-path attacker and with a spoofed inner address is gleaned by an xTR then the attacker may divert the traffic meant to be delivered to the spoofed EID as long as the gleaned entry is used by the xTR. This attack can be used as part of replay, packet manipulation, packet interception and suppression, or DoS attacks as the packets are sent to the attacker.

If the packet sent by the attacker contains a spoofed outer address instead of a spoofed inner address then it can achieve a DoS or a performance attack as the traffic normally destined to the attacker

will be redirected to the spoofed source RLOC. Such traffic may overload the owner of the spoofed source RLOC, preventing it from operating properly.

If the packet injected uses both inner and outer spoofing, the attacker can achieve a spoofing, a performance, or an amplification attack as traffic normally destined to the spoofed EID address will be sent to the spoofed RLOC address. If the attacked LISP site also generates traffic to the spoofed EID address, such traffic may have a positive amplification factor.

A gleaning attack does not only impact the data-plane but can also have repercussions on the control-plane as a Map-Request is sent after the creation of a gleaned entry. The attacker can then achieve DoS and performance attacks on the control-plane. For example, if an attacker sends a packet for each address of a prefix not yet cached in the EID-to-RLOC cache of an xTR, the xTR will potentially send a Map-Request for each such packet until the mapping is installed which leads to an over-utilisation of the control-plane as each packet generates a control-plane event. In order for this attack to succeed, the attacker may not need to use spoofing. This issue can occur even if gleaning is turned off since whether or not gleaning is used as the ITR may need to send a Map-Request in response to incoming packets whose EID is not currently in the cache.

Gleaning attacks are fundamentally involving a time-shifted mode of operation as the attack may last as long as the gleaned entry is kept by the targeted xTR. RFC 6830 [RFC6830] recommends to store the gleaned entries for only a few seconds which limits the duration of the attack.

Gleaning attacks always involve external data-plane attackers but results in attacks on either the control-plane or data-plane.

Note, the outer spoofed address does not need to be the RLOC of a LISP site, it may be any address.

3.2. Locator Status Bits

When the L bit in the LISP header is set to 1, it indicates that the second 32-bits longword of the LISP header contains the Locator Status Bits. In this field, each bit position reflects the status of one of the RLOCs mapped to the source EID found in the encapsulated packet. The reaction of a LISP xTR that receives such a packet is left as operational choice in [RFC6830].

When an attacker sends a LISP encapsulated packet with an illegitimately crafted LSB to an xTR, it can influence the xTR’s

choice of the locators for the prefix associated to the source EID. In case of an off-path attacker, the attacker must inject a forged packet in the network with a spoofed inner address. An on-path attacker can manipulate the LSB of legitimate packets passing through it and hence does not need to use spoofing. Instead of manipulating the LSB field, an on-path attacker can also obtain the same result of injecting packets with invalid LSB values by replaying packets.

The LSB field can be leveraged to mount a DoS attack by either declaring all RLOCs as unreachable (all LSB set to 0), or by concentrating all the traffic to one RLOC (e.g., all but one LSB set to 0) and hence overloading the RLOC concentrating all the traffic from the xTR, or by forcing packets to be sent to RLOCs that are actually not reachable (e.g., invert LSB values).

The LSB field can also be used to mount a replay, a packet manipulation, or a packet interception and suppression attack. Indeed, if the attacker manages to be on the path between the xTR and one of the RLOCs specified in the mapping, forcing packets to go via that RLOC implies that the attacker will gain access to the packets.

Attacks using the LSB are fundamentally involving a time-shifted mode of operation as the attack may last as long as the reachability information gathered from the LSB is used by the xTR to decide the RLOCs to be used.

3.3. Map-Version

When the Map-Version bit of the LISP header is set to 1, it indicates that the low-order 24 bits of the first 32 bits longword of the LISP header contain a Source and Destination Map-Version. When a LISP xTR receives a LISP encapsulated packet with the Map-Version bit set to 1, the following actions are taken:

- o It compares the Destination Map-Version found in the header with the current version of its own configured EID-to-RLOC mapping, for the destination EID found in the encapsulated packet. If the received Destination Map-Version is smaller (i.e., older) than the current version, the ETR should apply the SMR (Solicit-Map-Request) procedure described in [RFC6830] and send a Map-Request with the SMR bit set.

- o If a mapping exists in the EID-to-RLOC Cache for the source EID, then it compares the Map-Version of that entry with the Source Map-Version found in the header of the packet. If the stored mapping is older (i.e., the Map-Version is smaller) than the source version of the LISP encapsulated packet, the xTR should send a Map-Request for the source EID.

A cross-mode attacker can use the Map-Version bit to mount a DoS attack, an amplification attack, or a spoofing attack. For instance if the mapping cached at the xTR is outdated, the xTR will send a Map-Request to retrieve the new mapping which can yield to a DoS attack (by excess of signalling traffic) or an amplification attack if the data-plane packet sent by the attacker is smaller, or otherwise uses fewer resources, than the control-plane packets sent in response to the attacker’s packet. With a spoofing attack, and if the xTR considers that the spoofed ITR has an outdated mapping, it will send an SMR to the spoofed ITR which can result in performance, amplification, or DoS attack as well.

Map-Version attackers are inherently cross mode as the Map-Version is a method to put control information in the data-plane. Moreover, this vector involves live attackers. Nevertheless, on-path attackers do not have specific advantage over off-path attackers.

3.4. Routing Locator Reachability

The Nonce-Present and Echo-Nonce bits in the LISP header are used to verify the reachability of an xTR. A testing xTR sets the Echo-Nonce and the Nonce-Present bits in LISP data encapsulated packets and include a random nonce in the LISP header of packets. Upon reception of these packets, the tested xTR stores the nonce and echoes it whenever it returns a LISP encapsulated data packets to the testing xTR. The reception of the echoed nonce confirms that the tested xTR is reachable.

An attacker can interfere with the reachability test by sending two different types of packets:

1. LISP data encapsulated packets with the Nonce-Present bit set and a random nonce. Such packets are normally used in response to a reachability test.

2. LISP data encapsulated packets with the Nonce-Present and the Echo-Nonce bits both set. These packets will force the receiving ETR to store the received nonce and echo it in the LISP encapsulated packets that it sends. These packets are normally used as a trigger for a reachability test.

The first type of packets are used to make xTRs think that an other xTR is reachable while it is not. It is hence a way to mount a DoS attack (i.e., the ITR will send its packet to a non-reachable ETR when it should use another one).

The second type of packets could be exploited to attack the nonce-based reachability test. If the attacker sends a continuous flow of

packets that each have a different random nonce, the ETR that receives such packets will continuously change the nonce that it returns to the remote ITR, which can yield to a performance attack. If the remote ITR tries a nonce-reachability test, this test may fail because the ETR may echo an invalid nonce. This hence yields to a DoS attack.

In the case of an on-path attacker, a packet manipulation attack is necessary to mount the attack. To mount such an attack, an off-path attacker must mount an outer address spoofing attack.

If an xTR chooses to periodically check with active probes the liveness of entries in its EID-to-RLOC cache (as described in section 6.3 of [RFC6830]), then this may amplify the attack that caused the insertion of entries being checked.

3.5. Instance ID

LISP allows to carry in its header a 24-bits value called Instance ID and used on the ITR to indicate which local Instance ID has been used for encapsulation, while on the ETR the instance ID decides the forwarding table to use to forward the decapsulated packet in the LISP site.

An attacker (either a control-plane or data-plane attacker) can use the instance ID functionality to mount an intrusion attack.

3.6. Interworking

[RFC6832] defines Proxy-ITR and Proxy-ETR network elements to allow LISP and non-LISP sites to communicate. The Proxy-ITR has functionality similar to the ITR, however, its main purpose is to encapsulate packets arriving from the DFZ (Default-Free Zone) in order to reach LISP sites. A PETR (Proxy Egress Tunnel Router) has functionality similar to the ETR, however, its main purpose is to inject de-encapsulated packets in the DFZ in order to reach non-LISP sites from LISP sites. As a PITR (or PETR) is a particular case of ITR (or ETR), it is subject to similar attacks as ITRs (or ETRs).

As any other system relying on proxies, LISP interworking can be used by attackers to hide their exact origin in the network.

3.7. Map-Request messages

A control-plane off-path attacker can exploit Map-Request messages to mount DoS, performance, or amplification attacks. By sending Map-Request messages at high rate, the attacker can overload nodes involved in the mapping system. For instance sending Map-Requests at

high rate can considerably increase the state maintained in a Map-Resolver or consume CPU cycles on ETRs that have to process the Map-Request packets they receive in their slow path (i.e., performance or DoS attack). When the Map-Reply packet is larger than the Map-Request sent by the attacker, that yields to an amplification attack. The attacker can combine the attack with a spoofing attack to overload the node to which the spoofed address is actually attached.

Note, if the attacker sets the P bit (Probe Bit) in the Map-Request, it will cause legitimately sending the Map-Request directly to the ETR instead of passing through the mapping system.

The SMR bit can be used to mount a variant of these attacks.

For efficiency reasons, Map-Records can be appended to Map-Request messages. When an xTR receives a Map-Request with appended Map-Records, it does the same operations as for the other Map-Request messages and so is subject to the same attacks. However, it also installs in its EID-to-RLOC cache the Map-Records contained in the Map-Request. An attacker can then use this vector to force the installation of mappings in its target xTR. Consequently, the EID-to-RLOC cache of the xTR is polluted by potentially forged mappings allowing the attacker to mount any of the attacks categorized in Section 2.2 (see Section 3.8 for more details). Note, the attacker does not need to forge the mappings present in the Map-Request to achieve a performance or DoS attack. Indeed, if the attacker owns a large enough EID prefix it can de-aggregate it in many small prefixes, each corresponding to another mapping and it installs them in the xTR cache by mean of the Map-Request.

Moreover, attackers can use Map Resolver and/or Map Server network elements to relay its attacks and hide the origin of the attack. Indeed, on the one hand, a Map Resolver is used to dispatch Map-Request to the mapping system and, on the other hand, a Map Server is used to dispatch Map-Requests coming from the mapping system to ETRs that are authoritative for the EID in the Map-Request.

3.8. Map-Reply messages

Most of the security risks associated with Map-Reply messages will depend on the 64 bits nonce that is included in a Map-Request and returned in the Map-Reply. Given the size of the nonce (64 bits), if best current practice is used [RFC4086] and if an ETR does not accept Map-Reply messages with an invalid nonce, the risk of an off-path attack is limited. Nevertheless, the nonce only confirms that the Map-Reply received was sent in response to a Map-Request sent, it does not validate the contents of that Map-Reply.

If an attacker manages to send a valid (i.e., in response to a Map-Request and with the correct nonce) Map-Reply to an ITR, then it can perform any of the attacks categorised in Section 2.2 as it can inject forged mappings directly in the ITR EID-to-RLOC cache. For instance, if the mapping injected to the ITR points to the address of a node controlled by the attacker, it can mount replay, packet manipulation, packet interception and suppression, or DoS attacks, as it will receive every packet destined to a destination lying in the EID prefix of the injected mapping. In addition, the attacker can inject a plethora of mappings in the ITR to mount a performance attack by filling up the EID-to-RLOC cache of the ITR. The attacker can also mount an amplification attack if the ITR at that time is sending a large number of packets to the EIDs matching the injected mapping. In this case, the RLOC address associated to the mapping is the address of the real target of the attacker and so all the traffic of the ITR will be sent to the target which means that with one single packet the attacker may generate very high traffic towards its final target.

If the attacker is a valid ETR in the system, it can mount a rogue attack if it uses prefixes over-claiming. In such a scenario, the attacker ETR replies to a legitimate Map-Request message which it received with a Map-Reply message that contains an EID-Prefix that is larger than the prefix owned by the attacker. For example if the owned prefix is 192.0.2.0/25 but the Map-Reply contains a mapping for 192.0.2.0/24, then the mapping will influence packets destined to other EIDs than the one attacker has authority on. With such technique, the attacker can mount the attacks presented above as it can (partially) control the mappings installed on its target ITR. To force its target ITR to send a Map-Request, nothing prevents the attacker to initiate some communication with the ITR. This method can be used by internal attackers that want to control the mappings installed in their site. To that aim, they simply have to collude with an external attacker ready to over-claim prefixes on behalf of the internal attacker.

Note, when the Map-Reply is in response to a Map-Request sent via the mapping system (i.e., not send directly from the ITR to an ETR), the attacker does not need to use a spoofing attack to achieve its attack as by design the source IP address of a Map-Reply is not known in advance by the ITR.

Map-Request and Map-Reply messages are exposed to any type of attackers, on-path or off-path but also external or internal attackers. Also, even though they are control message, they can be leveraged by data-plane attackers. As the decision of removing mappings is based on the TTL indicated in the mapping, time-shifted attackers can take advantage of injecting forged mappings as well.

3.9. Map-Register messages

Map-Register messages are sent by ETRs to Map Servers to indicate to the mapping system the EID prefixes associated to them. The Map-Register message provides an EID prefix and the list of ETRs that are able to provide Map-Replies for the EID covered by the EID prefix.

As Map-Register messages are protected by an authentication mechanism, only a compromised ETR can register itself to its allocated Map Server.

A compromised ETR can over-claim the prefix it owns in order to influence the route followed by Map-Requests for EIDs outside the scope of its legitimate EID prefix (see Section 3.8 for the list of over-claiming attacks).

A compromised ETR can also de-aggregate its EID prefix in order to register more EID prefixes than necessary to its Map Servers (see Section 3.7 for the impact of de-aggregation of prefixes by an attacker).

Similarly, a compromised Map Server can accept an invalid registration or advertise an invalid EID prefix to the mapping system.

3.10. Map-Notify messages

Map-Notify messages are sent by a Map Server to an ETR to acknowledge the reception and processing of a Map-Register message.

Similarly to the pair Map-Request/Map-Reply, the pair Map-Register/Map-Notify is protected by a nonce making it difficult for an attacker to inject a falsified notification to an ETR to make this ETR believe that the registration succeeded when it has not.

4. Note on Privacy

As reviewed in [RFC6973], universal privacy considerations are difficult to establish as the privacy definitions may vary for different scenarios. As a consequence, this document does not aim at identifying privacy issues related to the LISP protocol but the security threats identified in this document could play a role in privacy threats as defined in section 5 of [RFC6973].

Similar to public deployments of any other control plane protocols, in an Internet deployment, LISP mappings are public and hence provide information about the infrastructure and reachability of LISP sites

(i.e., the addresses of the edge routers). Depending upon deployment details, LISP map replies might or might not provide finer grained and more detailed information than is available with currently deployed routing and control protocols.

5. Threats Mitigation

Most of the above threats can be mitigated with careful deployment and configuration (e.g., filter) and also by applying the general rules of security, e.g. only activating features that are necessary for the deployment and verifying the validity of the information obtained from third parties.

The control-plane is the most critical part of LISP from a security viewpoint and it is worth to notice that the LISP specifications already offer an authentication mechanism for mappings registration ([RFC6833]). This mechanism, combined with LISP-SEC [I-D.ietf-lisp-sec], strongly mitigates threats in non-trustable environments such as the Internet. Moreover, an authentication data field for Map-Request messages and Encapsulated Control messages was allocated [RFC6830]. This field provides a general authentication mechanism technique for the LISP control-plane which future specifications may use while staying backward compatible. The exact technique still has to be designed and defined. To maximally mitigate the threats on the mapping system, authentication must be used, whenever possible, for both Map-Request and Map-Reply messages and for messages exchanged internally among elements of the mapping system, such as specified in [I-D.ietf-lisp-sec] and [I-D.ietf-lisp-ddt].

Systematically applying filters and rate-limitation, as proposed in [RFC6830], will mitigate most of the threats presented in this document. In order to minimise the risk of overloading the control-plane with actions triggered from data-plane events, such actions should be rate limited.

Moreover, all information opportunistically learned (e.g., with LSB or gleaning) should be used with care until they are verified. For example, a reachability change learned with LSB should not be used directly to decide the destination RLOC, but instead should trigger a rate-limited reachability test. Similarly, a gleaned entry should be used only for the flow that triggered the gleaning procedure until the gleaned entry has been verified [Trilogy].

6. Security Considerations

This document provides a threat analysis and proposes mitigation techniques for the Locator/Identifier Separation Protocol.

7. IANA Considerations

This document makes no request to IANA.

8. Acknowledgments

This document builds upon the document of Marcelo Bagnulo ([I-D.bagnulo-lisp-threat]), where the flooding attack and the reference environment was first described.

The authors would like to thank Deborah Brungard, Ronald Bonica, Albert Cabellos, Ross Callon, Noel Chiappa, Florin Coras, Vina Ermagan, Dino Farinacci, Stephen Farrell, Joel Halpern, Emily Hiltzik, Darrel Lewis, Edward Lopez, Fabio Maino, Terry Manderson, and Jeff Wheeler for their comments.

This work has been partially supported by the INFSO-ICT-216372 TRILOGY Project (www.trilogy-project.org).

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.

9. References

9.1. 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, <http://www.rfc-editor.org/info/rfc6830>.

[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, <http://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,
<http://www.rfc-editor.org/info/rfc6833>.

[RFC6834] Iannone, L., Saucez, D., and O. Bonaventure, "Locator/ID Separation Protocol (LISP) Map-Versioning", RFC 6834, DOI 10.17487/RFC6834, January 2013, <http://www.rfc-editor.org/info/rfc6834>.

[RFC6973] Cooper, A., Tschofenig, H., Aboba, B., Peterson, J., Morris, J., Hansen, M., and R. Smith, "Privacy Considerations for Internet Protocols", RFC 6973, DOI 10.17487/RFC6973, July 2013, <http://www.rfc-editor.org/info/rfc6973>.

9.2. Informative References

[I-D.bagnulo-lisp-threat]
Bagnulo, M., "Preliminary LISP Threat Analysis", draft-bagnulo-lisp-threat-01 (work in progress), July 2007.

[I-D.ietf-lisp-ddt]
Fuller, V., Lewis, D., Ermagan, V., and A. Jain, "LISP Delegated Database Tree", draft-ietf-lisp-ddt-03 (work in progress), April 2015.

[I-D.ietf-lisp-sec]
Maino, F., Ermagan, V., Cabellos-Aparicio, A., and D. Saucez, "LISP-Security (LISP-SEC)", draft-ietf-lisp-sec-09 (work in progress), October 2015.

[RFC4086] Eastlake 3rd, D., Schiller, J., and S. Crocker, "Randomness Requirements for Security", BCP 106, RFC 4086, DOI 10.17487/RFC4086, June 2005, <http://www.rfc-editor.org/info/rfc4086>.

[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, DOI 10.17487/RFC7215, April 2014, <http://www.rfc-editor.org/info/rfc7215>.

[RFC7258] Farrell, S. and H. Tschofenig, "Pervasive Monitoring Is an Attack", BCP 188, RFC 7258, DOI 10.17487/RFC7258, May 2014, <http://www.rfc-editor.org/info/rfc7258>.

[Trilogy] Saucez, D. and L. Iannone, "How to mitigate the effect of scans on mapping systems", Trilogy Future Internet Summer

School., 2009.

Appendix A. Document Change Log (to be removed on publication)

- o Version 15 Posted January 2016.
  - * Few changes to address Stephen Farrel comments as part of the IESG Review.
- o Version 14 Posted December 2015.
  - * Editorial changes according to Deborah Brungard’s (Routing AD) review.
- o Version 13 Posted August 2015.
  - * Keepalive version.
- o Version 12 Posted March 2015.
  - * Addressed comments by Ross Callon on the mailing list (http://www.ietf.org/mail-archive/web/lisp/current/msg05829.html).
  - * Addition of a section discussing mitigation techniques for deployments in non-trustable environments.
- o Version 11 Posted December 2014.
  - * Editorial polishing. Clarifications added in few points.
- o Version 10 Posted July 2014.
  - * Document completely remodelled according to the discussions on the mailing list in the thread http://www.ietf.org/mail-archive/web/lisp/current/msg05206.html and to address comments from Ronald Bonica and Ross Callon.
- o Version 09 Posted March 2014.
  - * Updated document according to the review of A. Cabellos.
- o Version 08 Posted October 2013.
  - * Addition of a privacy consideration note.
  - * Editorial changes

- o Version 07 Posted October 2013.
  - * This version is updated according to the thorough review made during October 2013 LISP WG interim meeting.
  - * Brief recommendations put in the security consideration section.
  - * Editorial changes
- o Version 06 Posted October 2013.
  - * Complete restructuration, temporary version to be used at October 2013 interim meeting.
- o Version 05 Posted August 2013.
  - * Removal of severity levels to become a short recommendation to reduce the risk of the discussed threat.
- o Version 04 Posted February 2013.
  - * Clear statement that the document compares threats of public LISP deployments with threats in the current Internet architecture.
  - * Addition of a severity level discussion at the end of each section.
  - * Addressed comments from V. Ermagan and D. Lewis' reviews.
  - * Updated References.
  - * Further editorial polishing.
- o Version 03 Posted October 2012.
  - * Dropped Reference to RFC 2119 notation because it is not actually used in the document.
  - * Deleted future plans section.
  - * Updated References
  - * Deleted/Modified sentences referring to the early status of the LISP WG and documents at the time of writing early versions of the document.

* Further editorial polishing.

* Fixed all ID nits.

o Version 02 Posted September 2012.

* Added a new attack that combines over-claiming and de-aggregation (see Section 3.8).

* Editorial polishing.

o Version 01 Posted February 2012.

* Added discussion on LISP-DDT.

o Version 00 Posted July 2011.

* Added discussion on LISP-MS>.

* Added discussion on Instance ID.

* Editorial polishing of the whole document.

* Added "Change Log" appendix to keep track of main changes.

* Renamed "draft-saucez-lisp-security-03.txt.

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: ggx@gigix.net

Olivier Bonaventure
Universite catholique de Louvain
Place St. Barbe 2
Louvain la Neuve
Belgium

Email: olivier.bonaventure@uclouvain.be

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.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|N|L|E|V|I|R|K|K|            Nonce/Map-Version                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 Instance ID/Locator-Status-Bits               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

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].

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|N|L|E|V|I|P|K|K|        Nonce/Map-Version      | Next Protocol |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 Instance ID/Locator-Status-Bits               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

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

LISP Working Group
Internet-Draft
Intended status: Informational
Expires: January 5, 2015

A. Rodriguez-Natal
A. Cabellos-Aparicio
Technical University of Catalonia
M. Portoles-Comeras
M. Kowal
D. Lewis
F. Maino
Cisco Systems
July 4, 2014

LISP-OAM (Operations Administration Management): Use cases and requirements
draft-rodrigueznatal-lisp-oam-00

Abstract

This document describes Operations Administration and Management (OAM) use-cases and the requirements that they have towards the LISP architecture.

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 5, 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. Definition of terms 3
3. Use Cases 3
3.1. General LISP operation 3
3.2. MPTCP 5
3.3. Multicast 6
3.4. NFV/SFC 7
4. Requirements 8
5. Acknowledgements 10
6. IANA Considerations 10
7. Security Considerations 10
8. Informative References 10
Authors' Addresses 12

1. Introduction

LISP with its location/ID split in place creates two separated namespaces, the RLOC space where the transit network elements are deployed and the EID space that applies to the end-hosts. This inherently splits the network in an underlay, represented by the RLOC space, and an overlay, represented by the EID space.

However, LISP introduces some drawbacks since relevant details of the underlay network are hidden to the overlay nodes (e.g, xTR). With LISP, an overlay node can learn about the reachability of a path towards a locator and its liveness. In terms of control, it can -by means of priorities and weights- load-balance traffic across different locators and, taking advantage of LISP-TE [I-D.farinacci-lisp-te] and LISP-SR [I-D.brockners-lisp-sr], control how the traffic flows through the underlay topology. However, overlay nodes lack of appropriate knowledge about the characteristics of the paths, such as loss, latency, delay, length in IP/AS hops, etc. Furthermore, LISP nodes have little knowledge about the topological location of the RTRs as well as the characteristics of the underlay paths interconnecting them.

The mechanisms specified by LISP to monitor and control the underlay may not be enough for the complex overlay services that are arising today. Indeed, nowadays there are a plethora of services that require fine-grain control and real-time information of the network state. Such services could take advantage of the programmable

overlay scheme that LISP introduces as long as the appropriate mechanisms to control and monitor the underlay are in place.

LISP can leverage the mapping system to operate, administer, and manage the underlay-overlay relationship. Network devices can push to the Mapping System information about the capabilities and state of the network in order to allow it to take the best network operation and management decisions.

In this document we analyze the most common use-cases of overlay services and the requirements -from an abstract point of view- that they impose on the LISP architecture.

2. Definition of terms

- o OAM: The term OAM is used in this document as the acronym for Operation, Administration and Management. It refers to the set of procedures and mechanism that ensure that a network deployment behaves as expected and adapts properly to new situations.

- o Underlay: In this document, underlay is used to refer to the set of physical devices (i.e. hosts, routers, servers, etc) that support the networking operation in general and the LISP operation in particular. It also refers to the address space on where those devices communicate. The underlay corresponds to the RLOC space.

- o Overlay: The term Overlay is used here to denote the virtual network that sits on top of the underlay thanks to the LISP namespace split. It also refers to the address space that the virtual network uses as well as to the devices that are deployed on that address space. The overlay corresponds to the EID space.

The rest of the terms are defined in their respective documents. See the LISP specification [RFC6830] for most of the definitions, [RFC6832] for PxTR, [I-D.ietf-lisp-lcaf] for LCAF and [I-D.farinacci-lisp-te] for RTR.

3. Use Cases

3.1. General LISP operation

The overlay introduced by LISP provides an abstract view of the network that simplifies the deployment and operation of the network and its services. However this abstraction also hides the details of the underneath physical topology. While the overlay deployment can be fully defined at a logical level, the underlay is permanently subject to physical state changes that can affect the overall performance. Any LISP deployment has to deal with both the overlay

and underlay management and with underlay issues that can impact the overlay operation. In this context, the overlay needs to be aware of the underlay state in order to adapt itself to the current network conditions.

A LISP deployment where the overlay has detailed information of the underlay presents several advantages. First it can help troubleshooting the deployment. For instance, when a problem is detected, it is easy to know if it is due to misconfiguration on the LISP overlay, or rather from a physical problem on the underlay. Second, the underlay information can be used to influence policy decisions such as dynamically adapting the locators' priority and weight values based on the network state observed on the underlay. Finally, it can serve to automate the configuration of certain parts of the overlay deployment.

This is the case when underlay topological information is used to automatically select on a xTR which PxTR to use. Nowadays, PxTRs are generally manually configured, PITRs are provisioned with the EID prefixes they announce and the PETR to use is fixed on xTR boxes. With the proper overlay-underlay information exchange, these settings can be adapted over time. For instance, the PITR that is announcing an EID prefix can change to a secondary PITR in order to reduce round-trip time (RTT) if the EID prefix moves to a different RLOC, or the PETR used by a certain xTR can be replaced with a new one when the PETR goes down or the underlay network conditions change (e.g. the delay increases or the throughput decreases).

In order to provide the ability to operate with knowledge of the underlay, the LISP protocol could be extended to allow collection of underlay metrics that could then be pushed to the overlay. In terms of collected metrics, there are a few that would improve LISP operations. Some of these metrics could be extracted from the network state, by passive measurement or active probing, such as locator reachability, delay and throughput for a path, packet loss and MTU for a link, etc. Those metrics can be directly applied to the LISP policies (e.g. announcing a locator as down if it is not reachable anymore), can incrementally modify the policies (e.g. changing dynamically LISP weight values based on the observed delay or throughput), or can be applied after a threshold has been reached (e.g. setting a locator as down if the packet loss goes above a certain value). In addition to network state, it would be useful to keep track of LISP operation statistics, such as the size of the Map Cache or the last time a locator status changed. This would give more context of the underlay state and help the overlay to make better decisions.

3.2. MPTCP

Multipath TCP (MPTCP) [RFC6824] introduces several sub-flows in a single end-to-end TCP session while keeping a legacy TCP interface to the applications. This provides both resilience and bandwidth aggregation to hosts with multiple interfaces. MPTCP capabilities are negotiated between end-systems, which includes the capability of falling back to legacy TCP if negotiation is not possible. If the other end supports MPTCP, the original TCP flow is split into several sub-flows which are then forwarded over the different available links. Each of these sub-flows behaves as a legacy TCP flow and hence, from the network point of view, each sub-flow is a different TCP session. The network conditions over the different paths the sub-flows follow affect the whole MPTCP session. Since MPTCP has to keep the aggregate session consistent, each aggregated flow can perform as good as the worst of the sub flows it integrates.

As a consequence of this, MPTCP is really sensitive to unbalanced conditions on different links. Moreover, in an ideal scenario, the multiple sub-flows should follow disjoint paths, in order to ensure the maximum network utilization and the best link backup scenario. However, there is no way to ensure that the sub-flows will not cross paths beyond sending them through different interfaces from the end-point. On the other hand, legacy hosts do not support MPTCP and, in that case, proxies should be provisioned for them. All of these constraints make the overlay architecture proposed by LISP a suitable scenario for MPTCP deployments. Assuming the appropriate LISP-OAM mechanisms in place, MPTCP traffic over LISP should work as follows. Consider that a MPTCP capable source sends traffic towards a non-MPTCP capable destination. The LISP overlay has relevant information about the underlay and thus knows the best topology to deliver the traffic. It enforces this topology on the underlay by defining the points the flows will go through and where the flows will just be forwarded or balanced over different links. Since the destination is not MPTCP capable, all of the flows will be eventually be gathered at a proxy that will collapse them into a single flow that is forwarded to the destination. To handle the reply traffic, the single flow will first go through the proxy MPTCP and then the MPTCP subflows will be balanced again on the underlay via overlay management.

With LISP in place, and the MPTCP sub-flows being routed on the overlay, it is possible to adapt the overlay topology to match one that offers better performance for the MPTCP session. Disjoint and balanced paths may be enforced by means of using RTRs on the underlay. MPTCP proxies can be deployed at xTRs or RTRs and the traffic then routed to/from them using LISP. In order to compute this suitable topology, the Mapping System needs to be provided with several pieces of information regarding the network components

themselves: which prefixes should use MPTCP for their communications, which among them are not MPTCP enabled and thus have to go through a proxy, where are these proxies located and which RTRs can be used to create the topology. The Mapping System would need to know the state of the underlay network to create the best paths among the devices. Some metrics that would be of interest to retrieve, in terms of MPTCP, are the bandwidth among the xTRs, the RTRs and the proxies, the latency observed on their connections, etc. Finally, the Mapping System needs a way to tell the participants of the overlay what to do with the traffic, i.e. it needs to tell a MPTCP proxy which EID prefixes flows should be split or merged, it needs to indicate an RTR how to balance the different sub-flows it receives among the different paths that are available, etc.

3.3. Multicast

LISP defines several options to handle multicast operation between LISP sites. [RFC6831] describes how LISP interacts with traditional multicast protocols, i.e. how multicast traffic generated and managed by multicast specific protocols are handled by LISP devices. The multicast distribution tree creation and the multicast interaction with the network is leveraged on those legacy multicast protocols. "LISP Control-Plane Multicast Signaling" [I-D.farinacci-lisp-mr-signaling] proposes an alternative method to support multicast operation among LISP sites fully supported by the LISP control-plane. It covers the signaling to build the multicast distribution tree, however how it computes the tree topology is not within the scope of the document. "Signal-Free LISP Multicast" [I-D.farinacci-lisp-signal-free-multicast] proposes to connect multicast capable LISP sites through a non-multicast capable transit network. The replication is done at the LISP edge devices and the packets are forwarded via unicast on the core network. In that proposal, there is no multicast tree built on the transit network. Finally, "LISP Replication Engineering" [I-D.coras-lisp-re] describes a mechanism to build multicast distribution trees over a unicast-only transit network by means of using RTRs as multicast replication points.

In general, multicast traffic management relies on building a multicast distribution tree where the multicast source is the root and the multicast receivers are the leaves. The multicast traffic is forwarded according to that distribution tree and replicated when needed. The topology of the tree impacts both the performance of the multicast deployment and the quality of service of multicast traffic delivery. In order to provide the best service, the multicast algorithm can use the overlay capabilities of LISP to build an optimized tree for the multicast participants based on their underlay topological location and the dynamic network conditions.

LISP-OAM mechanisms can be applied to build and maintain an optimized multicast tree. In a similar fashion to what is done in LISP-RE, underlay information can be pushed to the overlay management. In LISP-RE, the RTRs involved in the multicast process register themselves in the Mapping System, letting it know that they may be used to build the distribution tree. Beyond multicast-capable device discovery, a LISP-OAM architecture could potentially feed the Mapping System with underlay information relevant to the multicast tree computation, such as the replication capacity in the underlay devices or the latency among them. Also, the multicast policies can be enforced in detail from the Mapping System, for instance setting up some nodes for only forwarding while keeping others for both forwarding and replication.

3.4. NFV/SFC

Network Function Virtualization (NFV) is a methodology that brings the advantages of traditional server virtualization to network functions. Virtual Network Functions (VNFs) are no longer tied to the hardware and can be dynamically instantiated, moved, and modified on demand. On the other hand, Service Function Chaining (SFC) is a proposal to provide a framework to manage and orchestrate chains of service functions that are applied to traffic across the network. In both proposals, LISP can play a role, since the overlay it provides can be used to deploy or improve deployments of NFV and/or SFC. An architecture of LISP for NFV is already described in [I-D.barkai-lisp-nfv]. The applicability of LISP to support SFC is discussed in [I-D.farinacci-lisp-te] and in [I-D.ietf-sfc-problem-statement]

The network functions (virtualized or not), of a LISP-based NFV or SFC deployment, will be deployed on LISP devices on the underlay (either xTRs or RTRs) and the data traffic will be managed over the overlay. The Mapping System will store the functions chains that should be applied to specific traffic and traffic engineering policies, such as the ones described in [I-D.farinacci-lisp-te], will be used to ensure that traffic goes through the network functions.

Deploying NFV or SFC solutions on top of LISP, in order to leverage its overlay, requires a bi-directional communication among the underlay devices and the overlay. The overlay must discover the underlay devices that provide network functions and understand how they are connected. It also needs to know the state of both the underlay network and the underlay devices in terms of latency or bandwidth among the devices as well as current load per device. In the NFV/SFC use-case, it is particularly important that the devices are able to announce the functions (virtual or not) that they provide, or that they are capable of providing. On the other hand, a

LISP-OAM architecture for NFV/SFC must be able to program the appropriate service chains in the Mapping System and to instantiate and manage on demand VNFs in the capable devices.

4. Requirements

The use-cases presented in Section 3 show the importance of including OAM mechanisms into the LISP protocol to make a better use of the overlay-underlay architecture. Based on those use-cases, this section proposes a set of requirements that should be fulfilled by a LISP-OAM solution. These requirements may be modified and/or extended in the future based on further use-cases discussion or experimental experience. Note that each requirement is meant to cover a specific need, all of them are independent and can be individually added to LISP. However, the more requirements addressed, the better the overlay can leverage the underlay.

- o Device discovery: The overlay needs to know the LISP devices (xTR, PxTR and RTR) that are available and that can be used to handle traffic. This is solved for xTRs by sending Map Register messages. A similar approach can be followed to automatically discover PxTRs and RTRs.

- o Capability discovery: The overlay must be aware of the capabilities of the nodes participating in the overlay, although LISP functionality is assumed in all LISP devices, the OAM mechanisms need further information. Based on the use-cases discussed in this document the capabilities to be announced by the devices are:

  - * Support for MPTCP flow balancing

  - * Network functions implemented on the device

  - * VNFs that the device can instantiate

  - * Capacity to replicate packets

  The capabilities should be encoded on a specific format (e.g a YANG model in XML, a new LCAF, JSON data, etc) and submitted to the overlay using LISP signaling (e.g. including capabilities information on the Map Registers) or leveraging on other existing protocols.

- o Underlay state access: The overlay needs as much underlay information as possible to make the best topology and policy decisions. Underlay devices have to implement ways to collect, store and offer this information to the overlay. According to the

use-cases described in this document the metrics to be collected are:

* Latency

* Packet loss

* Path length (IP/AS hops)

* MTU

* LISP state (map-cache, locator status, etc)

* System load

* Replication capacity

* VNFs instantiated

The metrics have to be encoded (e.g. YANG, LCAF, JSON, etc) and communicated to the overlay. The way to communicate them can be either a push mechanism (e.g. Map Register) that would simplify operation but requires a central administration entry, or a pull approach (e.g Map Request) that would allow the overlay to retrieve only on-demand information. The pull mechanism also serves as a way to specify which information is relevant for the overlay and to trigger metric collection if it was not already ongoing. In any case, the underlay device may decide to limit the information that it shares with the overlay.

o Forwarding actions: Some use-cases require that the overlay defines actions on how to process packets. According to the use-cases analyzed in this document the actions are:

* Forwarding: the basic forwarding action as defined in LISP.

* Replicate: Replicate an EID packet and forward it to a set of RLOCs.

* Balance flows: Distribute EID flows across different RLOCs. The flows are identified by a source/destination tuple, a 5-tuple, etc.

* Apply NF: Apply a (virtual or not) network function to the EID traffic.

These actions can be implemented as extensions to the current specifications of LISP-TE or LISP-SR or be defined by means of a

new LCAF. Some use-cases will narrow down actions via options, i.e. to define the algorithm to balance flows, the specific network function to be applied, etc.

Some of the required LISP extensions to support OAM may be offloaded to existing solutions, for instance using configuration protocols such NETCONF to get the PETR address on an xTR, build a YANG model to express devices capabilities or instantiate VNFs via NFV specific protocols.

5. Acknowledgements

6. IANA Considerations

This memo includes no request to IANA.

7. Security Considerations

In certain environments, multiple components of the LISP architecture may be managed in a distributed fashion (i.e., a Map Server, an ITR, and an ETR may be managed each individually by three separate organizations). When including capabilities to allow for the discovery of devices and its capabilities, as well as the collection of metrics regarding the underlay and the local device itself, it should be taken into consideration that proper controls are put in place to enforce strict policies as to which devices can access what type(s) of information.

8. Informative References

[I-D.barkai-lisp-nfv]
Barkai, S., Farinacci, D., Meyer, D., Maino, F., Ermagan, V., Rodriguez-Natal, A., and A. Cabellos-Aparicio, "LISP Based FlowMapping for Scaling NFV", draft-barkai-lisp-nfv-04 (work in progress), February 2014.

[I-D.brockners-lisp-sr]
Brockners, F., Bhandari, S., Maino, F., and D. Lewis, "LISP Extensions for Segment Routing", draft-brockners-lisp-sr-01 (work in progress), February 2014.

[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.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.farinacci-lisp-signal-free-multicast]
Moreno, V. and D. Farinacci, "Signal-Free LISP Multicast", draft-farinacci-lisp-signal-free-multicast-01 (work in progress), June 2014.

[I-D.farinacci-lisp-te]
Farinacci, D., Lahiri, P., and M. Kowal, "LISP Traffic Engineering Use-Cases", draft-farinacci-lisp-te-04 (work in progress), January 2014.

[I-D.ietf-lisp-lcaf]
Farinacci, D., Meyer, D., and J. Snijders, "LISP Canonical Address Format (LCAF)", draft-ietf-lisp-lcaf-04 (work in progress), January 2014.

[I-D.ietf-sfc-problem-statement]
Quinn, P. and T. Nadeau, "Service Function Chaining Problem Statement", draft-ietf-sfc-problem-statement-07 (work in progress), June 2014.

[RFC6020] Bjorklund, M., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, October 2010.

[RFC6241] Enns, R., Bjorklund, M., Schoenwaelder, J., and A. Bierman, "Network Configuration Protocol (NETCONF)", RFC 6241, June 2011.

[RFC6824] Ford, A., Raiciu, C., Handley, M., and O. Bonaventure, "TCP Extensions for Multipath Operation with Multiple Addresses", RFC 6824, January 2013.

[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.

[RFC7159] Bray, T., "The JavaScript Object Notation (JSON) Data Interchange Format", RFC 7159, March 2014.

Authors' Addresses

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

Marc Portoles-Comeras
Cisco Systems
170 Tasman Drive
San Jose, CA
USA

Email: mportole@cisco.com

Michael Kowal
Cisco Systems
111 Wood Avenue South
ISELIN, NJ
USA

Email: mikowal@cisco.com

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

Email: darlewis@cisco.com

Fabio Maino
Cisco Systems
170 Tasman Drive
San Jose, CA
USA

Email: fmaino@cisco.com