LISP tutorial

Damien Saucez
Olivier Bonaventure
Agenda

- Introduction
- Locator/Identifier Separation Protocol (LISP)
  - LISP in a nutshell
  - LISP data-plane
  - LISP control-plane
- The reachability problem
- Evaluation
- Conclusion
Introduction
The complementary roles of IP addresses

• The IP addresses currently used by endhosts play two complementary roles

  • **Identifier role:** the IP address identifies (with port) the endpoint of transport flows

  • **Locator role:** the IP address indicates the paths used to reach the endhost

• these paths are updated by routing protocols after each topology change
The Locator/Identifier Separation

- Today, changing the locator means changing the identifier, breaking the pending flows
- Separating the locator and the identifier roles to avoid breaking the flows
  - Host-based approach
  - Network-based approach
Host-based Loc/ID split

- **Roles**
  - Translates the packets so that
    - Transport layer always sees only the host identifier
    - IP Routing sublayer sees only locators
  - Manages the set of locators
  - Securely switches from one locator to another upon move or after link failure
  - each host maintains some state
Host-based Loc/ID split

- Roles
  - Translates the packets so that
    - Transport layer always sees only the host identifier
  - IP Routing sublayer sees only locators
  - Manages the set of locators
  - Securely switches from one locator to another upon move or after link failure
    - each host maintains some state
Host-based Loc/ID split

- **Transport layer**
  - Identifier: Ia

- **IP Routing sublayer**
  - Locators: {Ra, Rb}

**Roles**
- Translates the packets so that
  - Transport layer always sees only the host identifier
  - IP Routing sublayer sees only locators
- Manages the set of locators
- Securely switches from one locator to another upon move or after link failure
- Each host maintains some state
Host-based Loc/ID split

- **Roles**
  - Translates the packets so that
    - Transport layer always sees only the host identifier
  - IP Routing sublayer sees only locators
  - Manages the set of locators
  - Securely switches from one locator to another upon move or after link failure
  - each host maintains some state
Network-based Loc/ID split
Network-based Loc/ID split

- Host’s IP stack unchanged
- Each host has one stable IP address
- used as identifier
- not globally routable
Network-based Loc/ID split

- Each edge router owns
  - globally routed addresses used as locators
- **Mapping mechanism** is used to find locators associated to one identifier
- Packets from hosts are modified before being sent on Internet

Host’s IP stack unchanged
- Each host has one stable IP address
- used as identifier
- **not globally routable**

Locators for C/c: a.1.2.3, b.4.5.6

Transport layer
IP Routing layer

Wednesday 31 March 2010
Locator/Identifier Separation Protocol (LISP)
LISP in a nutshell
LISP Main Design Goals

- Minimize required changes to Internet
- No end-systems (hosts) changes
- Be incrementally deployable
- No router hardware changes
- Minimize router software changes
Alternative deployments?

• LISP’s first intention is to improve the global Internet

• What about using LISP in:
  • enterprises environments (VPNs)
  • datacenter
The Locator Identifier Separation Protocol (1/2)

- Define a router-based solution where current IP addresses are separated in two different spaces:
  - **EndPoint Identifiers (EID)**
    - identify end-hosts
    - non-globally routable
    - hosts in a given site are expected to use EIDs in the same prefix
  - **Routing Locators (RLOC)**
    - attached to routers (router interfaces)
    - globally routable
The Locator Identifier Separation Protocol (2/2)

• Follows the Map-and-Encap principle

• A **mapping system** maps EID prefixes onto site routers RLOCs

• Routers **encapsulate** the packets received from hosts before sending them towards the destination RLOC

• Routers **decapsulate** the packets received from the Internet before sending them towards the destination hosts
Terminology

• **Ingress Tunnel Router (ITR):** A router which accepts a packet containing a single IP header. The router maps the destination address of the packet to an RLOC and prepends a LISP header before forwarding the encapsulated packet.

• **Egress Tunnel Router (ETR):** A router which accepts a LISP encapsulated packet. The router strips the LISP header and forwards the packet based on the next header.
Terminology

- **EID-to-RLOC Database**: a globally distributed database that contains all known EID-prefix to RLOC mappings

- **LISP Cache**: EID-to-RLOC Database stored at the ITR

- **LISP Database**: EID-to-RLOC Database stored at the ETR
The big picture

EID-to-RLOC db

- **2001:DB8B::/56**
  - 3.2.2.1 1 100%
  - 2.2.2.1 2 100%

- **2001:DB8A::/56**
  - 1.1.1.1 1 75%
  - 2.1.1.1 1 25%

ISP3
- 3/8

ISP1
- 1/8

ISP2
- 2/8

Site 1
- 1.1.1.1
- 2.1.1.1

Site 2
- 3.2.2.1
- 2.2.2.1

2001:DB8A::1234

2001:DB8B::5678

Wednesday 31 March 2010
The big picture

**EID-to-RLOC db**

- **2001:DB8B::/56**
  - 3.2.2.1 1 100%
  - 2.2.2.1 2 100%

- **2001:DB8A::/56**
  - 1.1.1.1 1 75%
  - 2.1.1.1 1 25%

Site 1
- 1.1.1.1

Site 2
- 2.2.2.1
- 3.2.2.1

ISP1
- 1/8

ISP3
- 3/8

ISP2
- 2/8
The big picture

EID-to-RLOC db

2001:DB8B::/56
- 3.2.2.1 1 100%
- 2.2.2.1 2 100%

2001:DB8A::/56
- 1.1.1.1 1 75%
- 2.1.1.1 1 25%

ISP1
1/8

ISP2
2/8

ISP3
3/8

Site 1

Site 2

2001:DB8B::5678

Wednesday 31 March 2010
The big picture

EID-to-RLOC db

2001:DB8B::/56
• 3.2.2.1 1 100%
• 2.2.2.1 2 100%

2001:DB8A::/56
• 1.1.1.1 1 75%
• 2.1.1.1 1 25%

Map-Request: 2001:DB8B::5678?

Wednesday 31 March 2010
The big picture

EID-to-RLOC db

2001:DB8B::/56
- 3.2.2.1 1 100%
- 2.2.2.1 2 100%

2001:DB8A::/56
- 1.1.1.1 1 75%
- 2.1.1.1 1 25%

Map-Reply:
2001:DB8B::/56
- 3.2.2.1 1 100%
- 2.2.2.1 2 100%

ISP3
3/8

ISP1
1/8

ISP2
2/8

Site 1

Site 2

Wednesday 31 March 2010
The big picture

EID-to-RLOC db

2001:DB8B::/56
• 3.2.2.1 1 100%
• 2.2.2.1 2 100%

2001:DB8A::/56
• 1.1.1.1 1 75%
• 2.1.1.1 1 25%

ISP3
3/8

ISP1
1/8

ISP2
2/8

Site 1

Site 2

2001:DB8B::5678

Wednesday 31 March 2010
The big picture

EID-to-RLOC db

2001:DB8B::/56
- 3.2.2.1  1  100%
- 2.2.2.1  2  100%

2001:DB8A::/56
- 1.1.1.1  1  75%
- 2.1.1.1  1  25%

ISP 3
3/8

ISP 1
1/8

ISP 2
2/8

Site 1

Site 2
The big picture

EID-to-RLOC db

2001:DB8B::/56
- 3.2.2.1 1 100%
- 2.2.2.1 2 100%

2001:DB8A::/56
- 1.1.1.1 1 75%
- 2.1.1.1 1 25%

ISP1 1/8

ISP2 2/8

ISP3 3/8

Site 1

Site 2

- 3.2.2.1
- 2.2.2.1

Wednesday 31 March 2010
The big picture

EID-to-RLOC db

2001:DB8B::/56
• 3.2.2.1 1 100%
• 2.2.2.1 2 100%

2001:DB8A::/56
• 1.1.1.1 1 75%
• 2.1.1.1 1 25%

ISP1
1/8

ISP2
2/8

ISP3
3/8

Site 1

Site 2

1.1.1.1

2.1.1.1

3.2.2.1

2.2.2.1

Wednesday 31 March 2010
LISP Data-plane
Data-plane packets

Locator part

Presentation

Control Flags

Identifier Part

Payload
Header details

(IP(UDP(LISP(IP))))
Header details

(IP(UDP(LISP(IP))))
Header details

(IP(UDP(LISP(IP))))

Source/Destination Locators

Random Source Port

Source Routing Locator

Destination Routing Locator

Source Port = xxxx | Dest Port = 4341

UDP Length | UDP Checksum

Nonce

Locator Status Bits

Source EID

Destination EID
% Header details

(IP(UDP(LISP(IP))))
## Header details

(IP(UDP(LISP(IP))))

```

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 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   | Version | IHL   |   Type of Service   |     Total Length      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     | Identification | Flags | Fragment Offset     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| OH | Time to Live | Protocol = 17 |     Header Checksum   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   |       Source Routing Locator       |   Destination Routing Locator   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   |       Source Port = xxxx       |       Dest Port = 4341       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   | UDP Length | UDP Checksum     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| L   | N | L | E | rflags |   Nonce                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| I \ | Locator Status Bits             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| | 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 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   | Version | IHL   |   Type of Service   |     Total Length      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     | Identification | Flags | Fragment Offset     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| IH | Time to Live | Protocol |      Header Checksum   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   |       Source EID   |   Destination EID   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
Header details

(IP(UDP(LISP(IP))))
Header details

(IP(UDP(LISP(IP))))

N: Nonce present bit
L: Locator Status Bits present bit
E: Echo-Nonce request bit
Header details

(IP(UDP(LISP(IP))))
Header details

(IP(UDP(LISP(IP))))
Src/Dst Locators

- Source locator: IP address of the ITR that performed the encapsulation
- Destination locator: IP address of one ETR responsible for the destination EID’s prefix
- Locators can be in IPv4 or IPv6
Src/Dst EID

- Source EID: IP address of the source end-host
- Destination EID: IP address of destination end-hosts
- EIDs can be in IPv4 or IPv6
- AFI(RLOC) can be different AFI(EID)
- LISP can be an IPv4/IPv6 transition mechanism (but does not support XAFI)
LISP is over UDP

- UDP to traverse firewalls/NAT, limit the impact of ECMP hashing on reordering...
- Source port is random
  - but per-flow source port is recommended
- Destination port is fixed to 4341
- Checksum is important if IPv6 RLOCs
Nonce

• Uniquely identify a packet by setting a nonce while encapsulating (at the ITR)
  • nonce is optional, $N=1$ if present
• Used as passive reachability test when $E=1$
  • ITR asks the ETR to start the Echo-Nonce algorithm
Echo-Nonce Algorithm

• xTR \( I \) wants to know if the RLOC it uses to reach ETR \( E \) is reachable:
  • generate a nonce \( n \) when encap to \( E \)
  • set \( E=I \)

• Next time the \( E \) sends a packet to \( I \), is set the nonce to \( n \)

• If \( I \) receives the nonce within a given time, it considers the RLOC reachable, otherwise \( E \) is considered unreachable
Locator Status Bits

- A vector of 32 bits (L bit set to 1 if present)
- Each **source** locator is mapped to one position in the vector
  
  \[
  \text{if } \text{locator\_status\_bit}(i) = 1
  \]
  
  RLOC \( i \) is reachable

  \[
  \text{else}
  \]
  
  RLOC \( i \) is not reachable

- How to determine the position in the vector?
  
  - Explicit Position (each RLOC has an explicit position in the
    locator status bits)
  
  - Locator Status Bits = non-reachability bloom filter

- How to set the bit? What is its meaning?
LISP Control-plane
LISP mapping

• Possible models for the mapping mechanism
  • Push model
    • LISP ETR routers receive from a protocol to be designed the mapping tables that they need to use to map EIDs onto RLOCs
  • Pull model
    • LISP ETR routers refresh their mapping table by querying the mapping mechanism each time they receive a packet whose mapping is unknown
  • Hybrid models
    • Push is used to place “popular” or “important” mappings on LISP ETR routers and they query for the less important mappings
LISP Mapping messages

- Map-Request
  - request for a mapping

- Map-Reply
  - provides the mapping requested by a Map-Request

- Map-Register
  - an ETR informs the mapping system that it is responsible for an EID prefix
  - provides the RLOCs where the mapping can be found
Map-Request

<table>
<thead>
<tr>
<th>Type=1</th>
<th>0</th>
<th>M</th>
<th>P</th>
<th>S</th>
<th>Reserved</th>
<th>Record Count</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Nonce . .</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>. . . Nonce</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Source-EID-AFI</th>
<th>ITR-AFI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source EID Address . .</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Originating ITR RLOC Address . .</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Reserved</th>
<th>EID mask-len</th>
<th>EID-prefix-AFI</th>
</tr>
</thead>
<tbody>
<tr>
<td>EID-prefix . .</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Map-Reply Record . .</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Mapping Protocol Data</th>
</tr>
</thead>
</table>
Map-Request

```
+----------+----------+----------+----------+----------+----------+----------+----------+
| Type=1   | 0 | M | P | S | Reserved | Record Count |
+----------+----------+----------+----------+----------+----------+----------+----------+
| Nonce ... | . . . Nonce
+----------+----------+----------+----------+----------+----------+----------+----------+
| Source-EID-AFI | ITR-AFI |
| Source EID Address ... |
| Originating ITR RLOC Address ... |
| Reserved | EID mask-len | EID-prefix-AFI |
| Rec | EID-prefix ... |
| Map-Reply Record ... |
| Mapping Protocol Data |
```

M: Map-Reply
P: probing bit
S: Solicitation bit
### Map-Request

<table>
<thead>
<tr>
<th>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</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type=1</td>
</tr>
<tr>
<td>-------------------------------------------------------------</td>
</tr>
<tr>
<td>Nonce . . .</td>
</tr>
<tr>
<td>. . . Nonce</td>
</tr>
<tr>
<td>Source-EID-AFI</td>
</tr>
<tr>
<td>Source EID Address ...</td>
</tr>
<tr>
<td>Originating ITR RLOC Address ...</td>
</tr>
<tr>
<td>Reserved</td>
</tr>
<tr>
<td>EID-prefix ...</td>
</tr>
<tr>
<td>Map-Reply Record ...</td>
</tr>
<tr>
<td>Mapping Protocol Data</td>
</tr>
</tbody>
</table>

Number of records in the request

M: Map-Reply  
P: probing bit  
S: Solicitation bit

Wednesday 31 March 2010
**Map-Request**

<table>
<thead>
<tr>
<th>Type=1</th>
<th>0</th>
<th>M</th>
<th>P</th>
<th>S</th>
<th>Reserved</th>
<th>Record Count</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Nonce . . .</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>. . . Nonce</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Source-EID-AFI</td>
<td>ITR-AFI</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Source EID Address ...</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Originating ITR RLOC Address ...</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Reserved</td>
<td>EID mask-len</td>
</tr>
<tr>
<td>Rec</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EID-prefix ...</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Map-Reply Record ...</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Mapping Protocol Data</td>
<td></td>
</tr>
</tbody>
</table>

**Number of records** in the request

**64-bits Nonce**, uniquely identifying the request

M: Map-Reply  
P: probing bit  
S: Solicitation bit
Map-Request

Number of records in the request

64-bits Nonce, uniquely identifying the request

EID of the source that generated the miss

M: Map-Reply
P: probing bit
S: Solicitation bit

Wednesday 31 March 2010
Map-Request

Number of records in the request

64-bits Nonce, uniquely identifying the request

EID of the source that generated the miss

RLOC of the ITR to reply to

M: Map-Reply
P: probing bit
S: Solicitation bit

Wednesday 31 March 2010
Map-Request

| Type=1 | [0|M|P|S] | Reserved | Record Count |
|--------|-----------|----------|--------------|
|        |           |          |              |
|        |           |          |              |
|        |           |          |              |
|        |           |          |              |
|        |           |          |              |
|        |           |          |              |
|        |           |          |              |

- Type: Type of message
- [0|M|P|S]: Map-Reply or probing bit
- Reserved: Reserved for future use
- Record Count: Number of records in the request

64-bits Nonce, uniquely identifying the request

EID of the source that generated the miss

RLOC of the ITR to reply to

EID prefix to retrieve a mapping for

M: Map-Reply
P: probing bit
S: Solicitation bit

Wednesday 31 March 2010
Map-Reply

```
| Type=2 | P|E| Reserved               | Record Count |
|--------|---|------------------------|--------------|
|        |   |                        |              |
|        |   |                        |              |
| Frame  |   |                        |              |
|        |   |                        |              |
|        |   |                        |              |
|        |   |                        |              |
```

R

```
<table>
<thead>
<tr>
<th>Locator Count</th>
<th>EID mask-len</th>
<th>ACT</th>
<th>A</th>
<th>Reserved</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>---</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>---</td>
<td></td>
</tr>
</tbody>
</table>
```

```
<table>
<thead>
<tr>
<th>Reserved</th>
<th>EID-AFI</th>
</tr>
</thead>
</table>
```

```
|                          EID-prefix                           |
|--------------------------|-----------------------------|
```

```
<table>
<thead>
<tr>
<th>Priority</th>
<th>Weight</th>
<th>M Priority</th>
<th>M Weight</th>
</tr>
</thead>
</table>
```

```
L +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Unused Flags      |R|           Loc-AFI             |
|-----------------------------|-----------------------------|
```

```
|                             Locator                           |
|-----------------------------|-----------------------------|
```

```
|                     Mapping Protocol Data                     |
|---------------------|-----------------------------|
```

30
Map-Reply

P: probing bit
E: Echo-nonce capable

P: probing bit
E: Echo-nonce capable
## Map-Reply

<table>
<thead>
<tr>
<th>Type=2</th>
<th>P</th>
<th>E</th>
<th>Reserved</th>
<th>Record Count</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Nonce . . .

Nonce . . .

Record TTL

<table>
<thead>
<tr>
<th>Locator Count</th>
<th>EID mask-len</th>
<th>ACT</th>
<th>A</th>
<th>Reserved</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Reserved</th>
<th>EID-AFI</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

EID-prefix

<table>
<thead>
<tr>
<th>Priority</th>
<th>Weight</th>
<th>M Priority</th>
<th>M Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Unused Flags | Loc-AFI |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Locator

<table>
<thead>
<tr>
<th>Mapping Protocol Data</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>
### Map-Reply

<table>
<thead>
<tr>
<th>0 1 2 3 4</th>
<th>5 6 7 8</th>
<th>9 0 1 2 3 4 5 6 7 8 9 0 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type=2</td>
<td>P</td>
<td>E</td>
</tr>
<tr>
<td>+---------+---------+-------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nonce . .</td>
<td></td>
<td></td>
</tr>
<tr>
<td>. . . Nonce</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+---------+---------+-------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Record TTL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+---------+---------+-------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Locator Count</td>
<td>EID mask-len</td>
<td>ACT</td>
</tr>
<tr>
<td>+---------+---------+-------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reserved</td>
<td></td>
<td>EID-AFI</td>
</tr>
<tr>
<td>+---------+---------+-------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EID-prefix</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+---------+---------+-------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>/</td>
<td>Priority</td>
<td>Weight</td>
</tr>
<tr>
<td>+---------+---------+-------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Unused Flags</td>
<td></td>
<td>Loc-AFI</td>
</tr>
<tr>
<td>+---------+---------+-------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Locator</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+---------+---------+-------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mapping Protocol Data</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- P: probing bit
- E: Echo-nonce capable

**Number of records in the request**

Copied from the Map-Request

**Wednesday 31 March 2010**
Map-Reply

- **P**: probing bit
- **E**: Echo-nonce capable

**Number of records in the request**

**Copied from the Map-Request**

**Lifetime of the record in min**

- **P**: probing bit
- **E**: Echo-nonce capable

<table>
<thead>
<tr>
<th>Type=2</th>
<th>P</th>
<th>E</th>
<th>Reserved</th>
<th>Record Count</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Nonce . . .</th>
</tr>
</thead>
<tbody>
<tr>
<td>. . . Nonce</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Record TTL</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Locator Count</th>
<th>EID mask-len</th>
<th>ACT</th>
<th>A</th>
<th>Reserved</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>EID-AFI</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>EID-prefix</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Priority</th>
<th>Weight</th>
<th>M Priority</th>
<th>M Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Unused Flags</th>
<th>R</th>
<th>Loc-AFI</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Locator</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Mapping Protocol Data</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

**Wednesday 31 March 2010**
Map-Reply

Number of records in the request

Copied from the Map-Request

Lifetime of the record in min

Authoritative bit

P: probing bit
E: Echo-nonce capable

Wednesday 31 March 2010
Map-Reply

Number of records in the request
Copied from the Map-Request
Lifetime of the record in min
Authoritative bit
Negative reply action

P: probing bit
E: Echo-nonce capable

Wednesday 31 March 2010
# Map-Reply

There are several fields in the Map-Reply, each with a specific purpose:

- **Number of records in the request**: Indicates the number of records in the request.
- **Copied from the Map-Request**: Indicates whether the fields are copied from the Map-Request.
- **Lifetime of the record in min**: Represents the lifetime of the record in minutes.
- **Authoritative bit**: Indicates if the record is authoritative.
- **Negative reply action**: Specifies the action to be taken in case of a negative reply.
- **Priority**: Represents the priority of the RLOCs. RLOCs with lower priority are preferred. If several have the same priority, load balance among them.
- **Weight**: Represents the percentage of traffic to this RLOC when load balancing is active. (M for multicast)

### Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type=2</td>
<td>Reserved</td>
</tr>
<tr>
<td>Nonce ..</td>
<td>.. Nonce</td>
</tr>
<tr>
<td>Record TTL</td>
<td></td>
</tr>
<tr>
<td>Locator Count</td>
<td>EID mask-len</td>
</tr>
<tr>
<td>Record</td>
<td>Reserved</td>
</tr>
<tr>
<td>Locator Count</td>
<td>EID-mask-len</td>
</tr>
<tr>
<td>Record</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Probing Bits and Echo-Nonce Capable

- **P**: Probing bit
- **E**: Echo-nonce capable

---

**Wednesday 31 March 2010**
Map-Reply

Number of records in the request

Copied from the Map-Request

Lifetime of the record in min

Authoritative bit

Negative reply action

Priority: RLOCs with lower priority are preferred. If several have the same priority, load balance among them when load balancing is active. (M for multicast)

Is record is reachable from responder’s viewpoint?

P: probing bit
E: Echo-nonce capable

Wednesday 31 March 2010
How to control incoming traffic?

- LISP site can control incoming traffic with Weight and Priority
How to control incoming traffic?

- LISP site can control incoming traffic with Weight and Priority

- A Primary, B Backup

  - 2001:DB8B::/56
    - 3.2.2.1, prio: 1, weight: 100
    - 2.2.2.1, prio: 99, weight: 100
How to control incoming traffic?

- LISP site can control incoming traffic with Weight and Priority

- A Primary, B Backup
  - 2001:DB8B::/56
    - 3.2.2.1, prio: 1, weight: 100
    - 2.2.2.1, prio: 99, weight: 100

- A 60%, B 40%
  - 2001:DB8B::/56
    - 3.2.2.1, prio: 1, weight: 60
    - 2.2.2.1, prio: 1, weight: 40

LISP ITR will load balance layer 4 flows by using hash as in ECMP
How to design the deployment?

- Should we allow more specific EID prefixes?
- How EID must be allocated?
  - Per interface? Per hosts? Per service? Per site?
- How to build the mappings?
  - 1 RLOC per EID? Several RLOC per EID?
Mapping Systems
NERD: A Not-so-novel EID to RLOC Database
NERD

• The only proposed push model
• Composed of 4 parts
  • a network database format;
  • a change distribution format;
  • a database retrieval/bootstrapping method;
  • a change distribution method

• Principles
  • An authority computes the mapping database based on the stored registrations
  • The database signed by the authority is stored on servers
  • ITR poll regularly the database servers to update their own mapping database
LISP+ALT
LISP+ALT

• An pull model mapping system

• A mapping mechanism that relies on an alternate topology to distribute mapping requests to mapping servers

• LISP ITR routers sending mapping request messages to ALT routers

• ALT routers forward those mapping messages between themselves on an overlay topology built by using GRE tunnels
LISP+ALT

- BGP announces *where* the mappings can be found
- Map-Requests are forwarded on the ALT
- Map-Replies are forwarded on the legacy Internet (directly sent to the ITRs’ RLOC)
- ! BGP does not give the mappings !
LISP+ALT issues

- Complex system with tunnels, BGP protocol (no discussion about policies), ...
- Still relies on lots of error-prone manual configuration
- Scalability will depend on whether aggregation will be possible
- If mapping requests are lost due to congestion, difficult to diagnose the problem or send them via another path
- Security needs to be studied
LISP+ALT, big question

• How to deploy it?
  • aggregation vs recovery vs TE
• how to cache the mappings?
The reachability problem
The reachability problem

• Today, preserving the prefixes reachability is mainly performed locally

• In LISP, the legacy Internet is EID agnostic
The reachability problem in today’s Internet

- In today’s Internet, routing protocols converge after a link failure to ensure that multihomed prefixes such as A/a remain reachable.
The reachability problem in today’s Internet

- In today’s Internet, routing protocols converge after a link failure to ensure that multihomed prefixes such as A/a remain reachable.
The reachability problem in a LISP-based Internet

- Upon failure of ETR1, A continues to advertise A/a via BGP
- How can ITR notice that ETR1 failed and that ETR2 should be used instead?
The reachability problem in a LISP-based Internet

- Upon failure of ETR1, A continues to advertise A/a via BGP
- How can ITR notice that ETR1 failed and that ETR2 should be used instead?
Solving the reachability problem with the locator status bits

- ETR2 notices the failure and informs all ITRs to which it is sending LISP encapsulated packets by setting the reachability bit of ETR1 to 0.
Solving the reachability problem with the locator status bits

ETR2 notices the failure and informs all ITRs to which it is sending LISP encapsulated packets by setting the reachability bit of ETR1 to 0.

- ETR2 notices the failure and informs all ITRs to which it is sending LISP encapsulated packets by setting the reachability bit of ETR1 to 0.
Solving the reachability problem with the locator status bits

- ETR2 notices the failure and informs all ITRs to which it is sending LISP encapsulated packets by setting the reachability bit of ETR1 to 0.
Solving the reachability problem with the locator status bits

ETR2 notices the failure and informs all ITRs to which it is sending LISP encapsulated packets by setting the reachability bit of ETR1 to 0.

- ETR2 notices the failure and informs all ITRs to which it is sending LISP encapsulated packets by setting the reachability bit of ETR1 to 0.

Wednesday 31 March 2010
Solving the reachability problem with the locator status bits

ETR2 notices the failure and informs all ITRs to which it is sending LISP encapsulated packets by setting the reachability bit of ETR1 to 0.

- ETR2 notices the failure and informs all ITRs to which it is sending LISP encapsulated packets by setting the reachability bit of ETR1 to 0.
ETR2 notices the failure and informs all ITRs to which it is sending LISP encapsulated packets by setting the reachability bit of ETR1 to 0.
Solving the reachability problem with the locator status bits

- ETR2 notices the failure and informs all ITRs to which it is sending LISP encapsulated packets by setting the reachability bit of ETR1 to 0.
Solving the reachability problem with the locator status bits

- ETR2 notices the failure and informs all ITRs to which it is sending LISP encapsulated packets by setting the reachability bit of ETR1 to 0
Solving the reachability problem with the locator status bits

- ETR2 notices the failure and informs all ITRs to which it is sending LISP encapsulated packets by setting the reachability bit of ETR1 to 0.
Solving the reachability problem
with the locator status bits

- ETR2 notices the failure and informs all ITRs to which it is sending LISP encapsulated packets by setting the reachability bit of ETR1 to 0
Solving the reachability problem with the locator status bits

- ETR2 notices the failure and informs all ITRs to which it is sending LISP encapsulated packets by setting the reachability bit of ETR1 to 0.
ETR2 notices the failure and informs all ITRs to which it is sending LISP encapsulated packets by setting the reachability bit of ETR1 to 0.
Solving the reachability problem with the SMR bit

- ETR1 has been decommissioned and ETR2 wants that all the sites currently sending encapsulated data to itself update the mapping.

Wednesday 31 March 2010
Solving the reachability problem with the SMR bit

- ETR1 has been decommissioned and ETR2 wants that all the sites currently sending encapsulated data to itself update the mapping.

Wednesday 31 March 2010
Solving the reachability problem with the SMR bit

• ETR1 has been decommissioned and ETR2 wants that all the sites currently sending encapsulated data to itself update the mapping.
Solving the reachability problem with the SMR bit

• ETR1 has been decommissioned and ETR2 wants that all the sites currently sending encapsulated data to itself update the mapping.
Solving the reachability problem with the SMR bit

- ETR1 has been decommissioned and ETR2 wants that all the sites currently sending encapsulated data to itself update the mapping.

Wednesday 31 March 2010
Solving the reachability problem with the SMR bit

- ETR1 has been decommissioned and ETR2 wants that all the sites currently sending encapsulated data to itself update the mapping

C/c
>a.1.2.3, p=1
b.1.2.3, p=2

nonce = 1234
Map-Request

Wednesday 31 March 2010
Solving the reachability problem with the SMR bit

- ETR1 has been decommissioned and ETR2 wants that all the sites currently sending encapsulated data to itself update the mapping
Solving the reachability problem with the SMR bit

- ETR1 has been decommissioned and ETR2 wants that all the sites currently sending encapsulated data to itself update the mapping.

nonce = 1234

Map-Reply

Wednesday 31 March 2010
Solving the reachability problem with the SMR bit

- ETR1 has been decommissioned and ETR2 wants that all the sites currently sending encapsulated data to itself update the mapping.

nonce = 1234

Map-Reply

C/c > b.1.2.3, p=1

Wednesday 31 March 2010
Solving the reachability problem with the SMR bit

- ETR1 has been decommissioned and ETR2 wants that all the sites currently sending encapsulated data to itself update the mapping.
Evaluation
Evaluation of the cost of using LISP mappings

- Full Netflow (v7) on border router
- 1 Gigabit link to Belnet
- ~10000 users (/16 prefix block)
- Analysis: flow-tools + custom software
- /BGP Granularity of mappings
- iPlane data set

LISP Cache size

![Graph showing LISP Cache size over time with different timeout periods: 3 Min Timeout, 30 Min Timeout, 300 Min Timeout. The graph indicates the number of entries over time, with peaks and troughs at various intervals.](image)
Hit ratio (full PULL)

Hit Ratio (%)

00h 24h 12h 90 92 94 96 98 100

3 Min Timeout                    30 Min Timeout                 300 Min Timeout

Hit Ratio (%)

90 92 94 96 98

00h 12h 24h

3 Min Timeout 30 Min Timeout 300 Min Timeout

Wednesday 31 March 2010
Conclusion
Conclusion

- Network based Locator/Identifier separation proposal
- limited impact on the router
- limited impact on the traffic
- How to recover from failures?
- How to secure the whole system?