P4-LISP: A P4-Based High-Performance Router for the Locator/Identifier Separation Protocol

Benjamin Steinert, Marco Häberle, Jan-Oliver Nick, Dino Farinacci, Michael Menth

http://kn.inf.uni-tuebingen.de
What is P4?

P4-LISP router architecture & components
- LISP control plane lispers.net
- Local P4 controller
- P4 data plane

P4-LISP evaluations
Data Plane Programming with P4

- P4 = Programming Protocol-independent Packet Processors
- Programmable header processing
- Deployable on high-performance hardware
- More: see P4 survey [2]

Forward packets to P4 controller where no table-entries / mappings are available

(P)xTR: encapsulate, decapsulate

RTR: re-encapsulate

NAT-Traversal and LISP-NAT

Mobile Node support: double encapsulation

Security measures

- Rate-limit packets to P4 controller to 1pps, to not overload the P4 controller
- Process packets originating only from registered source EIDs
- Process packets destined only to valid destination EIDs
P4-LISP – Example Packet Flow

1. **Host 1**
   - EID: 10.10.1.1
   - **LISP site 1**
     - EID-prefix: 10.10.0.0/24

2. **ITR 1**
   - RLOC: 180.220.54.13

3. **ETR 2**
   - RLOC: 134.2.5.43

4. **Host 2**
   - EID: 20.20.20.2
   - **LISP site 2**
     - EID-prefix: 20.20.0.0/24

5. **LISP Control Plane lispers.net**
6. **P4 Controller**
7. **P4 Data Plane**
8. **LISP router**
9. **Mapping system**

10. **LISP site 1**
11. **LISP site 2**
12. **Mapping system**

Michael Menth | P4-LISP: A P4-Based High-Performance Router for the Locator/Identifier Separation Protocol
P4 Data Plane – Simplified xTR MAT Flowchart

Decapsulation Table
- Destination address: RLOCs of LISP site
- Action: Decapsulation
- No match: No action

Allowed Prefix Table
- Source address, Ingress port: EID-prefix of LISP site
- Action: No action
- No match: Drop

Valid Destination Table
- Destination address: Allowed prefixes
- Action: No action
- No match: Drop

Encapsulation Table (Map-Cache)
- Destination address, Random number: Map-Cache entries
- Action: Encapsulation
- No match: No action

IPv4 Table
- Destination address: Known destinations
- Action: Egress port
- No match: Send to controller

Last request >1s ago or new destination
- no: Drop packet
- yes: Send to controller

Known destination?
- no: Forwarding Table (ARP Table)
- yes: Serialize packet

Forwarding Table (ARP Table)
- Egress port: Set MAC addresses
- Port: Action
- No match: No action
# P4 Data Plane – Simplified MAT Flowchart

## Decapsulation Table
- **Destination address**: Action
- **RLOCs of LISP Site**: Decapsulation
- **No Match**: No Action

## Allowed Prefix Table
- **Source address, Ingress Port**: Action
- **EID Prefix of LISP Site**: No Action
- **No Match**: Drop

## Valid Destination Table
- **Destination address**: Action
- **Allowed Prefixes**: No Action
- **No Match**: Drop

## Encapsulation Table (Map-Cache)
- **Destination address, Random Number**: Action
- **Map-Cache Entries**: Encapsulation
- **No Match**: No Action

## LISP-NAT (inbound)
- **Destination address**: Action
- **Translated EID**: Replace Destination
- **No Match**: No Action

## LISP-NAT (outbound)
- **Source address**: Action
- **Translated EID**: Replace Source
- **No Match**: No Action

## IPv4 Table
- **Destination address**: Action
- **Known Destinations**: Egress Port
- **No Match**: Send To Controller

## Forwarding Table (ARP Table)
- **Egress port**: Action
- **Port**: Set MAC-Addresses
- **No Match**: No Action

---

**Legend:**
- **Table**
- **Conditional**
- **step**
- **Final step**

---

### Michael Menth | P4-LISP: A P4-Based High-Performance Router for the Locator/Identifier Separation Protocol
Evaluations

► Bandwidth evaluation

► Functional unit tests

► Latency analysis

► P4 controller performance
Data Rate Evaluation

- Data rate evaluation with re-encapsulation performs with around 100 Gbit/s
- Evaluations using P4TG [3]

Different scenarios were tested (xTR, PxTR, RTR)

Many extensions are supported
- NAT traversal
- Mobile Node
- LISP-NAT
- …
How large is the latency introduced by different components in P4-LISP?

### TABLE II
Mean and standard deviation (SD) of the latency caused by a cache-miss in the P4-LISP data plane.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Mean (ms)</th>
<th>SD (ms)</th>
<th>Description of the steps</th>
</tr>
</thead>
<tbody>
<tr>
<td>a-i</td>
<td>19.74</td>
<td>0.81</td>
<td>Worst-case latency: First packet sent by source until received by destination.</td>
</tr>
<tr>
<td>a-c</td>
<td>4.45</td>
<td>0.16</td>
<td>P4 controller delay for issuing discovery message.</td>
</tr>
<tr>
<td>d-e</td>
<td>12.61</td>
<td>0.72</td>
<td>Control plane delay.</td>
</tr>
<tr>
<td>f</td>
<td>0.80</td>
<td>0.10</td>
<td>P4 controller delay for processing control plane answer.</td>
</tr>
<tr>
<td>g-i</td>
<td>1.78</td>
<td>0.20</td>
<td>P4 controller delay for inserting table entry, until packet reaches destination.</td>
</tr>
</tbody>
</table>
Send 1000 packets with different destination IP addresses where no mapping is available

- Using nping [4]
- All of these packets are forwarded to the local P4 controller
- Different packet rates are examined

Overload at ~150 pps

- Latency rises
- No packet drops observed

P4-LISP is a high-performance LISP router implementation for data rates ~100 Gb/s

- Open-source: [https://github.com/uni-tue-kn/P4-LISP](https://github.com/uni-tue-kn/P4-LISP)
- Leverages open-source control plane lispers.net
- All LISP tunnel router types are implemented in a single P4 program (ITR, ETR, PITR, PETR, RTR)
- RTR can perform re-encapsulation at line rate
- Double encapsulation also at line rate, e.g., for LISP-MN
- Some security features against DoS attacks

Supports multiple extensions

- LISP Mobile Node
- NAT-Traversal
- Interworking mechanisms (e.g., LISP-NAT)
- Multihoming, load-balancing, and traffic engineering