History

• IETF 74th (Episode 1)
  • draft-iannone-lisp-mapping-versioning-00.txt
    • Presented the main idea and mechanism

• IETF 75th (Episode 2)
  • Discussion from the mailinglist
    • Versioning vs. <SMR-bit, Loc-Status-Bits, Nonce>

• IETF 77th (Episode 3)
  • draft-iannone-lisp-mapping-versioning-01.txt
    • Totally transparent to the main spec
    • Header format defined
    • Mapping Versioning vs. Hashing Versioning
Mapping Versioning

- Data-Plane conveys information how the packets have been encapsulated
- Upon a change in the version a Map-Request (or Map-Update-Solicitation) must be sent
- Control Plane “lighter” (no tracking necessary)
- Control Plane just sends Map-Request/Map-Update-Solicitation triggered by the Data Plane
- Internet-wide updates are data-driven (Clock sweep not necessary)
Pros & Cons

- **SMR+Nonce+Reachbits**
  - **PROS**
    - Implemented
  - **CONS**
    - Unidirectional
    - Heavy Control Plane
    - ReachBit have a “loose” meaning

- **Mapping Versioning**
  - **PROS**
    - Versioning
    - Bidirectional
    - Lighter Control Plane
    - Helps in keeping signaling overhead low
    - Mobility
  - **CONS**
    - To be implemented

*Does not mean to get rid of Nonce + SMR + Reachability bit in the Map-Reply

Now implemented in OpenLISP

SMR bit in LISP header not present anymore
Some other changes

- **LISP Header**
  - V-bit for optional presence of mapping versioning

- **Map Record**

- Interoperable with xTRs and MS not supporting versioning
Hashing Versioning design

- To make ETRs aware if PITR are using the correct mapping

- Same can be done with Mapping Versioning
  - If ETR does not cache a mapping for the source RLOC source version number is just ignored
Mobility and source version number: an example

- Assuming bidirectional traffic between a LISP-MN node and another LISP(-MN) node
- Mappings are up-to-date both sides
Mobility and source version number: an example

- MN1 moves to another ISP and updates its mapping
- Traffic to LISP2 still works since the mapping of LISP2 is still valid
- Traffic from LISP2 to MN1 is disrupted until LISP2 refreshes the entry in its cache
Mobility and source version number: an example

- Destination version number does not help here! Because MN1 is not receiving traffic
- Source version number tells LISP2 to update from the very first packet with the new version number
Mobility and source version number: an example

- Without source version number you either:
  - flush MNI cache (you disrupt also the traffic from MNI to LISP2)
    - Is not just the first packet!
  - send SMR to all entries in the cache (could be a problem if there are a lot of entries that are not used anymore but have not yet expired)
Mobility and source version number: an example

- With source version number everything is traffic driven and LISP2 will readily know that needs to update its mapping.
- In this case you need less machinery!
Another example: outage
Hashing

- ITR contacts ISPI via ETR1 and ETR2 at the same time (load balancing)
- ITR is aware of version 1, the last version is 3 but ITR1 does not have converged to this version yet
- We have oscillation
Versioning

- ITR contacts ISPI via ETR1 and ETR2 at the same time (load balancing)
- ITR is aware of version 1, the last version is 3 but ITR1 does not have converged to this version yet
- 3 is the most recent, no oscillation
Mapping Versioning

- Hashing Versioning is a simple application of the more general Mapping Versioning
- Mapping Versioning is more powerful and flexible because of
  - Source Version
  - Ordering
What will be the next Episode?
(Ready for a new season?)