LISP Predictive-RLOCs
Mobility with Near-Zero Packet Loss

draft-ietf-lisp-predictive-rlocs-00

LISP Working Group - Prague IETF
July 2017

Dino Farinacci and Padma Pillay-Esnault
Appendix B. Document Change Log

[RFC Editor: Please delete this section on publication as RFC.]

B.1. Changes to draft-ietf-lisp-predictive-rlocs-00.txt
   o Posted June 2017.
   o Make this specification a working group document. It is a copy of
     draft-farinacci-lisp-predictive-rlocs-02.

B.2. Changes to draft-farinacci-lisp-predictive-rlocs-02.txt
   o Posted May 2017 to update document timer.

B.3. Changes to draft-farinacci-lisp-predictive-rlocs-01.txt
   o Posted November 2016 to update document timer.

B.4. Changes to draft-farinacci-lisp-predictive-rlocs-00.txt
   o Initial post April 2016.
Problem Statement

- The mobility problem is simple ;-) 
  - When an EID moves, you send packets to the new location 
- NOT ;-) 
  - Packets already in the network are going to the old location (where the EID is no longer) 
  - EID has arrived at the new location but it is not receiving packets (sender doesn’t know about the move yet) 
- This is not “make-before-break”
Struggling Solutions

• Mobile-IP
  • You can’t send to home agent because it doesn’t know where the new location is

• Host Routes
  • They point to the old and the new location at the same time in different parts of the network
  • Handoffs are slow because the EID host route has to go everywhere

• Locator/ID Separation
  • A good solution if signaling is fast - sender gets new location quickly
Near-Zero Packet Loss

- We really don’t want to drop any packets
- We want handoffs to be instantaneous (atomic)
The Future is Clear

- What if we know all new locations?
- Have source send to all new locations
- We’ll search (and find) where the EID has roamed to
- Exercising a bandwidth/signaling tradeoff
Predictive RLOCs

LISP Mapping Database

'main-street' ->
RLE: RLOC-a, RLOC-b, RLOC-c, RLOC-d

EID-a ->
RLOC: 'main-street'

EID-b ->
RLOC: 'main-street'
Predictive RLOCs

Packet arrives at all xTRs, if EID discovered, forward packet else drop packet.
Predictive RLOCs

No control-plane signaling for $xTR-s$ to stop sending to $RLOC-a$ for $EID-b$

Local $xTR$s can prefetch $RLOC-s$ for server-side $EID$
Intersections

LISP Mapping Database

'wall-street' ->
RLE: RLOC-a, RLOC-b,
(RLE: RLOC-1, RLOC-3),
(RLE: RLOC-2, RLOC-4),
RLOC-c, RLOC-d

When EID-a is at xTR-a, driving straight ahead

'wall-street'

When EID-a is at xTR-b, anticipate turn north/south/ahead

EID-a

xTR-a

xTR-b

xTR-1

xTR-2

xTR-c

xTR-3

xTR-4

xTR-d
LISP Protocol Changes

• None

• Use RLE LCAFs for unicast map-cache entries

• By the way, multicast just works
  • When roaming EID is a receiver
  • When roaming EID is a source, (S-EID, G) cannot be pre-fetched
Quick Demo

Any road-side-unit xTR discovers a roaming EID \[1\text{2.2.2.2}\ldots\]

\[\text{lispers.net}\
\text{Scalable Open Overlay Networking}\
\text{Site name: any, EID-prefix: [1]2.2.2.2/32, registered: yes, dynamic}\
\text{Description:}\
\text{Last registerer: [0]127.0.0.1, xTR-ID: 0xf688382cdf56ea5d, site-ID: 0}\
\text{First registered: 0:00:22, last registered: 0:00:22, auth-type: sha2, registration flags: p-s-I-t-r-m-n}\
\text{Default registration timeout TTL: 180 seconds}\
\text{Forcing proxy Map-Reply: yes}\
\text{Forcing proxy Map-Reply for xTRs behind NATs: no}\
\text{Send drop-action proxy Map-Reply to PITR: no}\
\text{Proxy Map-Reply action: not configured}\
\text{Allowed RLOC-set: any}\
\text{Registered RLOC-set (replacement-semantics):}\
\text{[0]no-address, state: up-state, up/uw/mp/mw: 0/0/255/0, rloc-name: "replicate-to-each-rsu"}\
\text{rle: 10.1.1.1(L0), 10.2.2.2(L0), 10.3.3.3(L0)}\
\text{Individual registrations: none}\
\]

dino-macbook.local

\ldots\text{the RSU or a controller could register the predictive-RLOC mapping}
Quick Demo

ITR has EID [1]2.2.2.2 in its map-cache . . .

. . . replicates to predictive-RLOCs 10.1.1.1, 10.2.2.2, and 10.3.3.3
Plans for -01

- Introduce "RLE Usage Types", when RLE is [A, B, C, D]
- Initially replicate to all of [A, B, C, D], to solicit feedback
- When Usage is:
  - **Directional**: when EID is past B, replicate to [C, D] (default, in -00)
  - **Random**: when EID is at B, replicate to [B] only
  - **Circular**: when EID is at D, replicate to [A, ...], consider counter-clockwise
  - **Back-n-Forth**: when EID is at D, replicate to [C, B, A]
Plans for -01

- Document Pre-Fetch

  - When RSUs get packets from initial replication, do lookup on source-EID to populate map-cache

  - So when roaming-EID appears, there is no packet loss when roaming-EID sources packets to server side EID
Work in Progress

- Use geo-prefixes to reduce replication scope for future RLOCs
- Use overlapping RLEs to reduce replication scope
- Use multiple RLOC-records with shorter RLEs to reduce replication scope
- Use RTRs close to ETR so replication is $O(1)$ over RANs
- Use a level of indirection with distinguished-names for grouping roaming-EIDs to reduce predictive-RLOC duplication in different mappings
- LISP-crypto operation
  - Encrypt for each predictive-RLOC replication (like draft-ietf-lisp-signal-free-multicast)
  - Or encrypt once and replicate many (would have to share keys)
Questions/Comments/Tomatoes?