LISP Predictive-RLOCs
Mobility with Near-Zero Packet Loss

draft-farinacci-lisp-predictive-rlocs-00

LISP Working Group - Berlin IETF
July 2016

Dino Farinacci and Padma Pillay-Esnault
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 xTRs can prefetch RLOC-s for server-side EID

server-side EID

EID-a

xTR-a

RLOC-a

EID-b

xTR-b

RLOC-b

xTR-c

RLOC-c

xTR-d

RLOC-d

'main-street'

Internet

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

Local xTRs can prefetch RLOC-s for server-side EID.
Intersections

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

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

LISP Mapping Database

'wall-street' ->

RLE: RLOC-a, RLOC-b,
(RLE: RLOC-1, RLOC-3),
(RLE: RLOC-2, RLOC-4),
RLOC-c, RLOC-d

'main-street'
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]2.2.2.2 . . .

... 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
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?