Unstanding Mapping

Dan Jen and Lixia Zhang

RRG @ IETF75

Why This Talk

- Mobility support needs some sort of binding/ mapping
- Scalable routing needs some sort of binding/ mapping too
- Should we kill 2 birds by one stone?
  - Note the word “Should”, not “can”

→ Look before we leap: What are the basic differences between the two, if any?

Mapping/Binding for Mobility

(our observation)

- Mobile (host/subnet): identified by an "ID"
- Packets to Mobile: delivered to an IP address
- Binding: ID ←→ IP address
  - Can be done in different ways/at different layers
    - MIP: binding at IP layer, ID: in form of IP address
    - ILNP: binding through DNS; ID: in form of DNS name
  - Commonality
    - Updates sent to the binding server
    - All senders know exactly where (to get binding) to send packets
      - Know binding prior to data arrival
    - No caching by 3rd party

Mapping/binding for scalable routing

- Reduce RIB/FIB→entries removed from table
- Mapping:
  1. ID←→routed address (e.g. SHIM6, ILNP)
  2. Non-routed address←→routed address (e.g. APT, Ivip, LISP, six/one router)

(1) get mapping from DNS (with its own challenges)

Below we discuss and compare mapping of (2)

Scalable routing by Map-n-Encap

- Done by network entry point; transparent to sending hosts
  - Pre-propagate binding info (NERD, APT DM, VA)
  - find binding info upon data arrival and cache (APT ITR, LISP ITR)

Comparing the Two

- The two mapping systems function in two different and somewhat conflicting ways.
- Mobility mapping systems
  - Holding binding at (logically) one place
  - Granularity: Up to host movement
  - support frequent mapping information changes.
- Scalable routing:
  - Mapping info must be available at large number of data entry points
    - Either pre-distribute out, or
    - demand driven caching
    - Granularity: site

Using one mapping for both purposes

- Can one rely on caching to reduce lookup overhead?
  - Turn the problem to how to deal with stale cache entries
- Can one reduce cache TTL to reduce stale entries for mobiles?
  - Going back to high lookup overhead
- Can this be done?
- Would this make the best design tradeoff?