On Demand Mobility Management

draft-yegin-dmm-ondemand-mobility-00

A. Yegin, K. Kweon, J. Lee, J. Park
Samsung Electronics
Mobile IP

• Provides a persistent IP address (HoA) to MN
  – Cost: HA CAPEX/OPEX, triangular routing latency

• All applications on the MN subjected to same treatment all the time

• But not all apps really need that!
  – Typical client app does not need persistent IP address
  – Higher-layer solutions (e.g., MPTCP, SIP or app-mobility) are more efficient and may serve session continuity needs, when available
Approach

• Let the applications indicate their need, and IP stack engage IP-layer mobility accordingly
# Types of IP Addresses
(with respect to IP-layer mobility management)

<table>
<thead>
<tr>
<th>Type</th>
<th>Belongs to prefix managed by:</th>
<th>Persistency</th>
<th>Supports IP address reachability?</th>
<th>Supports IP session continuity?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Home Network Anchored Address</td>
<td>Centrally-located Home Agent</td>
<td>Fixed all the time</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Access Network Anchored Address</td>
<td>Serving or previous Access Router</td>
<td>Released after IP session(s) terminate</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Unanchored Address</td>
<td>Serving Access Router</td>
<td>Released upon IP handover</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>
RFC 5014

• “IPv6 Socket API for Source Address Selection”
• Defines IPV6_ADDR_PREFERENCES socket flags to influence source address selection
  – IPV6_PREFER_SRC_HOME
  – IPV6_PREFER_SRC_COA
• Not sufficient because
  – Home vs. CoA distinction not sufficient to capture 3 different types of IP addresses
  – Selects among available addresses, but on-demand configuration is needed too
    • Example use: Never configure the rarely-used Home Network Anchored Address until it’s requested by an app.
    • Indication of “Prefer(ence)” is not sufficient, we need “require(ment)”
Solution

• New IPV6_ADDR_PREFERENCES flags
  – IPV6_REQUIRE_HOME_ANCHORED
  – IPV6_REQUIRE_ACCESS_ANCHORED
  – IPV6_REQUIRE_UNANCHORED

• Works with “IPv6 Socket API for Source Address Selection (RFC 5014)” framework

• If the requested type is not already configured, then the IP stack attempts to dynamically configure one
Policy

• Following are policy matters:
  – The type of IP addresses configured on the host at the boot time.
  – Permission to grant various types of IP addresses to a requesting application.
  – Determination of a default address type when an application does not use the API.
Questions and comments?