IPv6 Current Status and Next Steps

USAGI/WIDE Project
Keio University

Hideaki YOSHIFUJI <yoshfuji@linux-ipv6.org>
Progress since Netconf 2005

- Mobile IPv6 infrastructure [net-2.6.19]
- Stateful Packet Inspection aka nf_conntrack
- Generic Segmentation Offload [net-2.6.19], UDP Fragmentation Offload
- IPv6 Ready Logo Phase-2: Core(Host) and IPsec (End-node)
- Advanced Socket API (RFC3542)
Progress since Netconf 2005 (cont'ed)

- Policy routing [net-2.6.19]
- Default router selection and more-specific routes
- Default source address selection
- Link detection
- DCCPv6
Status of Mobile IPv6 (MIP6)

- MIP6 infrastructure is available in net-2.6.19
  - CN support
  - Base for HA and MN
- HA and MN support will soon be ready as well.
### MIP6: patch outline

<table>
<thead>
<tr>
<th>Category</th>
<th>Status</th>
<th>MIPv6 Roles</th>
<th># of Patches</th>
</tr>
</thead>
<tbody>
<tr>
<td>Policy routing</td>
<td>Already in net-2.6.19</td>
<td>HA(*1), MN</td>
<td>Nearly 20</td>
</tr>
<tr>
<td>XFRM extension</td>
<td>Already in net-2.6.19</td>
<td>CN, HA, MN</td>
<td>44</td>
</tr>
<tr>
<td>Proxy forwarding</td>
<td>Almost ready for net-2.6.19</td>
<td>HA(*1)</td>
<td>Nearly 5</td>
</tr>
<tr>
<td>MN addressing</td>
<td>Almost ready for net-2.6.19</td>
<td>MN</td>
<td>Nearly 6</td>
</tr>
<tr>
<td>IPsec MIGRATE</td>
<td>in preparation</td>
<td>HA, MN</td>
<td>1-3</td>
</tr>
</tbody>
</table>

*1: w/ physical home link support*
MIP6: Proxy forwarding

- Required by HA with physical home link support:
  - Respond to unicast ND for proxy entry
  - Do not learn neighbors in our own Proxy ND list
    - Either its link-layer address is own or such address is off-link.
  - Provide an interface to specify router/host flag to announce NA for each proxy entry respectively.
    - Proxied node may NOT be a router but a host.
MIP6: MN Adressing

• New flags
  – IFA_F_NODAD: DAD avoidance
    • MN does NOT perform DAD for its home address when returning home link
    • IN6_IFF_NODAD in BSDs
  – IFA_F_HOMEADDRESS
    • for the source address selection, rule 4

• Interface to manipulate flags
MIP6: MIGRATE

- Feature to update endpoint addresses (locators) of an IPsec tunnel:
  - Update src/dst addresses of the template of a given SPD entry.
  - Update src/dst addresses of the SAD entry associated with the SPD entry.
- Necessary for the MIP6 operation where an IPsec tunnel is established between MN and HA:
  - A necessary interactions between MIP6 and IPsec/IKE.
  - Can also be used for Mobile VPN (MOBIKE).
- The message can be sent/received via PF_KEY and Netlink (xfrm_user).
- The MIP6 component (mip6d) issues a MIGRATE message targeting a SPD entry.
  - Assumes that the MIP6 component is aware of the security policy configuration that is relevant to MIP6 protocol operation.
MIP6: optional features?

- **Inbound trigger, or “inbound acquire”**
  - MN wants to know non-RO traffic to start (or judge to start) RR (Return Routability; a key exchange mechanism before RO)
  - Currently, xfrm acquire is used for this by adding a last-resort policy, however we have limitation to start it only when MN sends any packet because xfrm acquire is designed for IPsec thus it is applied only for outbound traffic.

- **New route flag as RTPROTO_MIP for mipv6 daemon (like zebra)**
Source Address Selection

• Part of Default Address Selection for IPv6 (RFC3484)

• Basic part has already been implemented
  – static policy table
    • policy table: longest-matching-prefix lookup table
    • additional policy to deal with unique local addresses (RFC4193)
      – fc00::/7
Source Address Selection (cont'ed)

- We SHOULD support configurable address selection; e.g. policy table
  - PREFSRC (rt6i_prefs src)
    - Uneasy to setup, hard to use...
      - Most people want to configure real routes and source address selection rules independently.
    - separate table seems better...
      - Another prefix->data database
IPv6 Host-to-Router Load Sharing (RFC4311)

- Hash based approach SHOULD be used.
  - Our equal-cost routers are held in a list, not in an array. How to achieve this?
- ...traffic for a given destination address will use the same router as long as the Destination Cache Entry is not deleted...(RFC4311)
  - not an issue if we use hash-based approach
  - /128 cache route for off-link destination for optimization
Early inet6_dev registration

• We do not allocate inet6_dev until an address has been assigned on the interface.

• Some per-interface variables are required to be set prior to the corresponding interface is brought up.

• Solution:
  - Call ipv6_find_idev() when we see NETDEV_REGISTER
    • inet6_dev w/o inet6_ifaddr
Autoconfiguration Failure

- A tentative address that is determined to be a duplicate ... MUST NOT be assigned to an interface ... If the address is a link-local address formed from an interface identifier ... IP operation on the interface SHOULD be disabled. (RFC2462bis)

- We keep joining all-node multicast address, thus we do receive unsolicited RAs and try assigning new address(es).
  - not good
Autoconfiguration Failure (cont'ed)

- Solution:
  - input path / forwarding to other interface) path
    - Per-interface variable; e.g. net.ipv6.conf.ethX.enable_ipv6
    - Check it at the entrance of ipv6_rcv()
    - BSDs have similar bits
  - output path / forwarding (from other interface) path
    - ???
Hop-by-Hop Option Processing

- We ALWAYS process Hop-by-Hop options.
  - without checking destination address
    - okay for routers, but NOT GOOD for hosts.
    - Solution: lookup routes in ipv6_rcv()
  - without any netfilter hooks
    - Oops...
Statistics

- Per-interface statistics
  - populated in inet6_dev
- HC (High Capacity) counters
RCUs

- `inet6_dev`
  - important for per-interface statistics
- `inet6_ifaddr`
- `ip_tunnel`
Tunnels

- ipip / ip_gre / sit unification
- IPv4 over IPv6
- Make sit modular
- ISATAP (Intra-Site Automatic Tunnel Addressing Protocol) (RFC4214)
Multicast

- Forwarding
- Specification compliant?
IPv6/Xen

• Does it work well?
• Knowhow / Documentation / Education
Other things we may hit

- HIP (Host Identity Protocol)
  - Serveral implementation available for Linux
    - requested to come up with unified patch
    - general BEET mode
- SHIM6 (Site Multihoming by IPv6 Intermediation)
  - discussion in usagi-users
- XCAST (eXplicit multi-uniCAST)
  - patch available for 2.6.15
TCP MD5 Signature Option (RFC2385)

• Old patch
• TODO
  - port to current tree
  - kill compatible API
Anything Else?