Host Centric Multi6

Christian Huitema
Architect
Windows Networking & Communications
Microsoft Corporation
Principle

- Site is connected to multiple providers
  - Get as many prefixes
- Prefixes are propagated to all site routers
  - Router renumbering?
- Hosts get as many addresses
  - Prefix = provider
  - Subnet number
  - Host ID
- Publish addresses in DNS
Issue: destination address selection

- Fairly common
  - Many hosts are multi-homed.

- Debate whether hosts have sufficient information
  - Hard for small appliances, not enough information
  - Easy for large servers

- It is not unrealistic to expect progress in this area,
  - communication between the hosts and the routers,
  - sharing of experience between hosts,
  - innovative application design
  At worst, a host can always try the proposed addresses one by one, and pick the first one that actually works -- not very elegant, but definitely workable.
Issue: source address selection

- Existing software ties source address selection to interface selection
  - Select outgoing interface
  - Pick one address on interface as source
  - Only consider address scope, and possibly "privacy" status

- Choosing the source address will affect the reverse path of the connection
  - Issue similar to "destination address selection"

- We need some improvement for multi-addressing
Issue: rapid reaction to topology change

- One of site X providers (A) becomes unreachable
- How do we avoid picking a source address A:X?
- How do peers avoid picking a destination address A:X?
Issue: site exit & ingress filtering

X picks source address A:X, dest D:Y
Routing fabric sends packet to exit router RXB
Provider B sees source = A:X, perform ingress filtering, rejects the packet
## Classification of the issues

<table>
<thead>
<tr>
<th>Issue</th>
<th>Classification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Destination address selection</td>
<td>Nice to have.</td>
</tr>
<tr>
<td>Source address selection</td>
<td>Nice to have.</td>
</tr>
<tr>
<td>Reaction to topology changes</td>
<td>Nice to have</td>
</tr>
<tr>
<td>Ingress filtering</td>
<td>MUST SOLVE</td>
</tr>
<tr>
<td></td>
<td>Should at least &quot;retry&quot;</td>
</tr>
</tbody>
</table>
Comparison of ingress filtering solutions

- **Relax address filtering**
  - Requires provider involvement
  - Easy to deploy for large sites

- **Source address dependent routing**
  - Variant = tunnels between exit routers

- **Packet rewriting at exit router**
  - Inferior to “exit tunnel” solution

- **Source address selection by the host**
  - Complement to source dependent routing
  - Requires “exit router” discovery
The “dumb host” requirement

- **Unmodified host**
  - Picks a single source address
  - Must work at least as well as “not multi-homed”

- **Consequence**
  - Ingress filtering must work for all destinations if source provider available
  - Imply either “relaxed filtering” or “per source routing”
Solution’s principle (dumb host)
Solution’s principle (exit discovery)

- Try can be as simple as sending a “ping”, maybe with source address = site local…
Solution’s principle (exit tunnel & redirect)

- There are alternatives, e.g. don’t use update, just a direct tunnel to the “right” exit.
Proposed solution

- Facilitate site exit
  - Site exit “logical” address (for tunnels)
  - Site exit redirect ICMP
  - Tunnel to appropriate exit

- Router advertisements for rapid reaction
  - preferred lifetime > 0 \( \Rightarrow \) source is OK
  - Need router renumbering

- Host improvements
  - Source and destination address selection
  - Exit router discovery (understand site exit ICMP)
  - Binding update / Mobile IPv6 for “reassignment”
Going forward

- Reconcile / merge with Bagnulo’s draft
  - Compare binding update versus advertisement of multiple addresses
- Study possible provider help
  - Some form of tunneling when provider link is broken…
- Get consensus for a narrow scope WG charter, or progress document without a WG