RIP
(Routing Information Protocol)
RIP (Routing Information Protocol)

- Distance vector algorithm
- Included in BSD-UNIX Distribution in 1982
- Distance metric: # of hops (max = 15 hops)

From router A to subsets:

<table>
<thead>
<tr>
<th>destination</th>
<th>hops</th>
</tr>
</thead>
<tbody>
<tr>
<td>u</td>
<td>1</td>
</tr>
<tr>
<td>v</td>
<td>2</td>
</tr>
<tr>
<td>w</td>
<td>2</td>
</tr>
<tr>
<td>x</td>
<td>3</td>
</tr>
<tr>
<td>y</td>
<td>3</td>
</tr>
<tr>
<td>z</td>
<td>2</td>
</tr>
</tbody>
</table>
RIP advertisements

- Distance vectors: exchanged among neighbors every 30 sec via Response Message (also called advertisement)
- Each advertisement: List of up to 25 destination nets within AS
RIP: Link failure and recovery

If no advertisement heard after 180 sec, neighbor/link declared dead

- Routes via the neighbor are invalidated
- New advertisements sent to neighbors
- Neighbors in turn send out new advertisements (if their tables changed)
- Link failure info quickly propagates to entire net
- Poison reverse used to prevent ping-pong loops (infinite distance = 16 hops)
RIP table processing

- RIP routing tables managed by an application process [Unix: route-d (daemon)]
- Advertisements encapsulated in UDP packets (no reliable delivery required)
- Advertisements are periodically repeated
RIP table example

Router: giroflee.eurocom.fr

- Three attached class C networks (LANs)
- Router only knows routes to attached LANs
- Default router used to “go up”
- Route multicast address: 224.0.0.0
- Loopback interface

<table>
<thead>
<tr>
<th>Destination</th>
<th>Gateway</th>
<th>Flags</th>
<th>Ref</th>
<th>Use</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>127.0.0.1</td>
<td>127.0.0.1</td>
<td>UH</td>
<td>0</td>
<td>26492</td>
<td>lo0</td>
</tr>
<tr>
<td>192.168.2.0</td>
<td>192.168.2.5</td>
<td>U</td>
<td>2</td>
<td>13</td>
<td>fa0</td>
</tr>
<tr>
<td>193.55.114.0</td>
<td>193.55.114.6</td>
<td>U</td>
<td>3</td>
<td>58503</td>
<td>le0</td>
</tr>
<tr>
<td>192.168.3.0</td>
<td>192.168.3.5</td>
<td>U</td>
<td>2</td>
<td>25</td>
<td>qaa0</td>
</tr>
<tr>
<td>224.0.0.0</td>
<td>193.55.114.6</td>
<td>U</td>
<td>3</td>
<td>0</td>
<td>le0</td>
</tr>
<tr>
<td>default</td>
<td>193.55.114.129</td>
<td>UG</td>
<td>0</td>
<td>143454</td>
<td></td>
</tr>
</tbody>
</table>
How to avoid Count-to-Infinity

- Split Horizon
- Split Horizon with Poisoned Reverse
- Triggered Updates
  - After metric change: Send update immediately
- Holddown Timer (Cisco)
  - After invalidation of route:
    For some seconds ignore all updates for route
Routing tasks: RIP

- Neighbor?
  - Discovery
  - Maintenance

- Database?
  - Granularity
  - Maintenance – updates
  - Synchronization

- Routing table?
  - Metric
  - Calculation
  - Update