Introduction

- Personal view as a person on the equipment vendor side.
- BGP design decisions.
- Frequent discussion topics:
  - How much hierarchy?
  - Where to place route reflectors.
  - Implications of MEDs and damping.
  - Next-hop self.
  - Advertising multiple paths in BGP.
An example

- 10 locations; 2 core routers each.
- Route reflection from core to access.
- Goal: keep traffic away from E-W links.
What is wrong with this picture?

- IGP metrics control which exit point gets selected.
- Top level of hierarchy unnecessary to meet requirement.
- Adds significant amount of complexity.
What does BGP do well?

- Database transfer of external routing information (bulk).
  - Designed for networks with 100s of iBGP mesh peers, millions of paths.
  - With rudimentary policy selection.
- It is not an IGP. Doesn’t care which internal links are up or down; doesn’t need to follow link topology.
  - Using BGP for internal traffic eng. is generally a bad idea.
Confederations <-> Reflection

- “You’re right! No need to use confederations. We will use 2 levels of route reflection instead”.
- Same beast by a different name.
- Confederations are equivalent to Reflection w/ no-client-to-client (as per spec).
- Difference: boundary on the link, or on the system.
Goal: Reduce routing information.
Otherwise you can end up with 2k copies of the routing table.
Non-goals: configuration management; scaling # TCP sessions.
Assume \{a, b\} reflectors for \{c, d, e\}

Without client-reflection: only c is used as exit point from d.

Beyond the cluster: lost path to e.
Configuration management

- In practice, many use RR as a configuration management tool.
- It is the wrong tool for the job: “side effects” of path selection are not usually understood.
- Solutions?
  - Automated scripts / provisioning system;
  - draft-raszuk-idr-ibgp-auto-mesh-00.txt;
Information hiding

- Confed per continent or top level RRs on both sides of the pond.
- Vs all major locations on top level mesh.
## Trade-offs

<table>
<thead>
<tr>
<th>Confed per continent</th>
<th>Large top level mesh</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1 path per inter-continent link.</strong></td>
<td><strong>1 off-continent path per city (worse case).</strong></td>
</tr>
<tr>
<td>Less info for choosing exit point.</td>
<td>More ability to do intra-domain TE.</td>
</tr>
<tr>
<td>Convergence depends on 2 RR hops.</td>
<td>Choice of remote exit point via IGP metric.</td>
</tr>
<tr>
<td>Ability to do policy.</td>
<td>No policy.</td>
</tr>
</tbody>
</table>
How RRs achieve efficiency

- **Statement:** BGP can do 100s of iBGP mesh peers or rr-clients.
- **Under what conditions is this true?**
- **BGP efficiency depends on peer-groups.**
  - Select which routes should be advertised once per group;
  - Format updates once per group;
  - Copy the update to N sockets;
- **Means BGP is as efficient w/ 1 peer or 100 per group (minus TCP processing).**
Caveat

- We left flow-control out of the previous equation (which is per peer).
- Revise: work is done per set of peers in the group which have approx. same flow-control state.
  - Implementation dependent: select updates to send once per group (or sub-group). JunOS only formats messages per sub-group.
- Particularly for an RR (sending full routes) the Round Trip Time distribution to clients does matter.
Recommendations

◆ Keep It Simple.
  - Engineering: find the lowest cost solution that satisfies the problem.

◆ Avoid loosing information in the core.
  - Keep your multiple city to city choices available.

◆ Avoid centralization.
  - Distribution improves resiliency and performance.
◆ Customer pays ISP to transport incoming traffic to selected location.

◆ From London POV: w/o MED 2 available paths; w/ MED only one.
Implications of cold-potato

- AMS router prefers MIL; and refrains from advertising its own path.
- Less information; only best overall path is known.
- Convergence: withdrawal of MIL path will cause AMS to advertise its alternate; LON will probably see MIL -> unreach -> AMS.
- JunOS has hidden knob to force advertisement of “best-external” route.
Cold-potato (continued).

- Likely-hood of MED oscillation problems: proportional to the number of hierarchies in the network.

- Simplest case:
  - In A: \( p_1 < p_2; p_2 < p_3 < p_1 \)
  - In B: \( p_2 < p_3; p_3 < p_1 \)
To “next-hop self”

... Or not to “next-hop self”.

◆ Advantages of external next-hop addresses:
  ▶ Metric of external link can be used to influence decision.
  ▶ Convergence in terms of IGP propagation.
    ◆ Assumes efficient detection of resolution changes by remote peer.

◆ Disadvantages:
  ▶ Need to configure external link as passive in IGP.
Damping

- Goal: eliminate noise generated by flapping tail circuit.
- Problem: it cannot distinguish between that case and changes caused by transit ASes (example: MED change).
- Current implementations create more problems than it solves.
- If you must: crank up suppress; low half-life so that only continuous flapping prefixes are suppressed.
Routing Views

- “Can BGP advertise more than one path?”
- RFC 2547
  - Route Distinguisher qualifies IP prefix.
  - Route Target community used to control which routes are imported into which forwarding tables.
- JunOS
  - Input firewall filter can specify which routing-instance to use for forwarding lookup.
- Use of tunneling (mpls, ip) in the core.
Policy: customer Ca uses upstream 1; other customers use best of all internet routes.
[edit routing-options]
rib-groups rg-isp1 {
    import-rib [inet.0 isp1.inet.0];
    /* optional import-policy */
}

[edit protocols bgp group isp1]
family inet unicast rib-group rg-isp1;
[edit routing-instances isp1]
instance-type vrf;
vrf-target target:10458:1; /* identify table */
Configuration - pe1

[edit routing-instances isp1]
instance-type vrf;
vrf-target target:10458:1; /* identify table */
[edit interfaces so-0/0/1.0 family inet]
filter input fbf;
[edit firewall filter fbf]
term a {
    from /* some criteria */
    then routing-instance isp1;
}
Limitations

- # entries in forwarding tables.
- Can selectively discard forwarding table state.
- No forwarding entries needed for diagnostic applications.
- Scaling of BGP: depends mostly on the number of events processed rather than number of total entries.
Recent JunOS BGP behavior changes

◆ 6.3
   √ Incoming interface check on EBGP sessions.
   √ Policy from aggregate-contributor.

◆ 7.0
   √ No EBGP poison reverse to neighbor-as.
   √ policy next-hop [discard | reject].
   √ TCP path mtu discovery (knob).
Thank You

http://www.juniper.net