Customized BGP Route Selection

Laurent Vanbever, Cristel Pelsser
UCLouvain, Internet Initiative Japan
laurent.vanbever@uclouvain.be, cristel@iij.ad.jp

Pierre François (UCLouvain, BE), Olivier Bonaventure (UCLouvain, BE) and Jennifer Rexford (Princeton, USA)

WIDE Camp
Tuesday, March 9 2010
Customized BGP Route Selection

Introduction and motivation

Implementing CRS

Practical considerations and solutions

Conclusion
Customized BGP Route Selection

Introduction and motivation

Implementing CRS

Practical considerations and solutions

Conclusion
The Internet is a collection of Autonomous Systems (AS)

- An AS is a set of routers managed by a single administrative entity
- Today, there are approximately 30,000 ASes
The Internet is a collection of Autonomous Systems (AS)

- An AS is a set of routers managed by a single administrative entity
- Today, there are approximately 30,000 ASes
The Internet is a collection of Autonomous Systems (AS)

- An AS is a set of routers managed by a single administrative entity
- Today, there are approximately 30,000 ASes
The Internet is a collection of Autonomous Systems (AS)

- An AS is a set of routers managed by a single administrative entity
- Today, there are approximately 30,000 ASes
BGP is the *path-vector, policy-based* interdomain routing protocol.

**UPDATE**
prefix: 192.0.2.0/24
ASPP: E A D

**UPDATE**
prefix: 192.0.2.0/24
ASPP: A D

**UPDATE**
prefix: 192.0.2.0/24
ASPP: C D

**UPDATE**
prefix: 192.0.2.0/24
ASPP: D
BGP is based on *sessions, policies* and a *decision process*

BGP sessions

BGP Adj-RIB-In

Input filters

Attribute Manipulation

Neighor₁

Input filters

Attribute Manipulation

Neighor₂

... 

BGP Decision Process

Best route to each destination

BGP Loc-Rib

All acceptable routes

BGP Adj-RIB-Out

Output filters

Attribute Manipulation

Neighor₁

Output filters

Attribute Manipulation

Neighor₂

... 

BGP sessions

BGP sessions

Input filters

Attribute Manipulation

Neighorₙ
A BGP router selects one best route for each destination.

Globally, AS E knows 4 paths towards D.

Locally, some routers only know one path (C1...C3, R1, R2)
BGP Route Selection: *One-route-fits-all* model

- Many ISPs have a rich path diversity
  - It is common to have 5-10 paths *per prefix*\(^1\)
- Different paths have different properties
  - It could be in terms of security, policies, etc.

---

Clients may want different paths to the same prefix
- If C1 is a competitor of C, he’d prefer to reach D via A or B
- C1 may even want to pay an extra fee for that
BGP Route Selection: *One-route-fits-all* model

- With vanilla BGP, you *can’t* match customers’ preferences to available paths
- Customers of a given PE receive the same path
Under CRS, one router can offer different interdomain routes to different neighbors.

- C1 reaches D via B, C2 reaches D via C

I’d prefer 1

I’d prefer 3

I’d prefer 4
Customized BGP Route Selection

Introduction and motivation

Implementing CRS

Potential issues and solutions

Conclusion
Under CRS, routes are *colorized* based on their properties

- A *color* denotes a set of routes sharing a property
  - *e.g.*, color *red* is associated to all *high-bandwidth* routes learned on *national* peerings
  - one route can have multiple colors

- Colors are “tags” associated to routes
  - we use the well-known BGP community field
What do we need to implement CRS with BGP MPLS VPNs?

- Mechanisms to *disseminate* and *differentiate* paths
  - Multiprotocol BGP is used as dissemination protocol
  - Route Targets (RT) are used to identify colors
  - Route Distinguishers (RD) are used to ensure diversity

- *Customized* route selection mechanisms at ASBR
  - Use of Virtual Routing and Forwarding (VRF) instances

- Traffic forwarding on the chosen paths
  - MPLS tunneling
How do we implement CRS with BGP MPLS VPNs?

- C1 wants to reach D via B, C2 via C
- Define 3 colors: routes learned via A (green), B (red) and C (blue)
- Announce red routes to C1, blue routes to C2
How do we implement CRS with BGP MPLS VPNs?
How do we implement CRS with BGP MPLS VPNs?

- Consider peers as VPNs and put them in VRFs
How do we implement CRS with BGP MPLS VPNs?

- Consider peers as VPNs and put them in VRFs
- Use RT to identify colors
How do we implement CRS with BGP MPLS VPNs?

- Consider peers as VPNs and put them in VRFs
- Use RT to identify colors
- Use different RD to differentiate routes

<table>
<thead>
<tr>
<th>Route Targets</th>
</tr>
</thead>
<tbody>
<tr>
<td>green learned via A</td>
</tr>
<tr>
<td>red learned via B</td>
</tr>
<tr>
<td>blue learned via C</td>
</tr>
</tbody>
</table>
How do we implement CRS with BGP MPLS VPNs?

- In each VRF, prefer certain routes via import filters

<table>
<thead>
<tr>
<th>Route Targets</th>
</tr>
</thead>
<tbody>
<tr>
<td>green learned via A</td>
</tr>
<tr>
<td>red learned via B</td>
</tr>
<tr>
<td>blue learned via C</td>
</tr>
</tbody>
</table>

prefer B routes
import RT: green, red, blue;
from red:
set pref:=200;

prefer C routes
import RT: green, red, blue;
from blue:
set pref:=200;
How do we implement CRS with BGP MPLS VPNs?

- MPLS is used for forwarding
  - Two levels label stack
  - R3 only knows label to reach the PEs

<table>
<thead>
<tr>
<th>Route Targets</th>
</tr>
</thead>
<tbody>
<tr>
<td>green learned via A</td>
</tr>
<tr>
<td>red learned via B</td>
</tr>
<tr>
<td>blue learned via C</td>
</tr>
</tbody>
</table>
Customized BGP Route Selection Using BGP/MPLS VPNs

Introduction and motivation

Implementing CRS

Practical considerations and solutions

Conclusions
Is CRS pushing a M120 to the limit?

Four tables are defined on the Unit Under Test (UUT)

- Each table is fed with one color (one RT)
- In each color, ~300k routes (1 path per route)
- In the end, 1,200,000 routes in RIB & FIB
Is CRS pushing a M120 to the limit?

- UUT was a Juniper M120 [JunOS 9.3R2.8]
- Routing Engine (RE) has 4 GB DRAM
- Forwarding Engine Boards (FEB) have 512 MB DRAM

<table>
<thead>
<tr>
<th></th>
<th>RE</th>
<th>FEB</th>
</tr>
</thead>
<tbody>
<tr>
<td>empty</td>
<td>17%</td>
<td>9%</td>
</tr>
<tr>
<td>fully-loaded (1.200.000 routes)</td>
<td>38%</td>
<td>39%</td>
</tr>
</tbody>
</table>

- FIB could handle more than 2,000,000 routes
- Enough to support a few services *without* modifications
More services? 

*scalability* and...*scalability*

- Routes *dissemination* overhead
  - **All** PEs receive **all** VPN routes

- Routes *storage* overhead
  - **RIB**
    - Modest performance demand
    - Add more DRAM to support CRS?
  - **FIB**
    - CRS’s biggest challenge
    - Sharing between the VRFs in the FIB?
How could we improve CRS FIB’s scaling: *Selective VRF Download*

- By default, *all* VRFs are installed on *all* line cards

<table>
<thead>
<tr>
<th>Slot</th>
<th>State</th>
<th>Temp (C)</th>
<th>CPU Utilization (%)</th>
<th>Memory (MB)</th>
<th>Utilization (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Online</td>
<td>24</td>
<td>1</td>
<td>512</td>
<td><strong>39</strong></td>
</tr>
<tr>
<td>3</td>
<td>Online</td>
<td>28</td>
<td>1</td>
<td>512</td>
<td><strong>39</strong></td>
</tr>
</tbody>
</table>

- Customers ask for the same colors?
  - Connect them on the same line card
  - Download VRFs only to line cards that need them
- It could be a management nightmare...
How could we improve CRS FIB’s scaling: *Cross-VRF Lookup*

- Specific routing for a small set of prefixes?
- Create one small VRF *per color*
- Add default entry towards a default VRF
- The price to pay is 2 IP lookups

<table>
<thead>
<tr>
<th>VRF1</th>
<th>VRF2</th>
</tr>
</thead>
<tbody>
<tr>
<td>*&gt;10/8 via R1</td>
<td>*&gt;10/8 via R2</td>
</tr>
<tr>
<td>0/0 via default</td>
<td>0/0 via default</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
</tr>
<tr>
<td>*&gt;10/8 via R3</td>
</tr>
<tr>
<td>...</td>
</tr>
</tbody>
</table>
How could we improve CRS FIB’s scaling: *Distributed VRF*

- Distribute VRFs among routers which can afford extra load
- PEs do not maintain complete VRFs anymore
- PEs default route traffic towards these routers
- Increase in latency and load
- Distributed version of *Cross-VRF Lookup*

R maintain small VRFs and default rest to R1 or R2

- detour path
- direct path
Customized BGP Route Selection

Introduction and motivation

Implementing CRS

Practical considerations and solutions

Conclusion
**CRS is feasible**

- **Implementable**
  - It can be realized on today’s routers
  - It uses well known BGP MPLS/VPNs techniques

- **Scalable (for a few services)**
  - “Modest” message and storage overhead
  - Lab experiments tend to confirm that
  - Full BGP tables are needed to complete our evaluation

- **Guaranteed interdomain convergence**
  - Extra flexibility does not compromise global routing stability\(^1\)

---

\(^1\) Proof in SIGMETRICS'09 paper by Y. Wang, M. Schapira, and J. Rexford
Customized BGP Route Selection

Questions?

Please, come and see our poster!