BEYOND THE BEST: REAL-TIME NON-INVASIVE COLLECTION OF BGP MESSAGES
Interdomain Routing = BGP

- BGP is the Internet glue
  - de-facto standard for interdomain routing
- BGP decides traffic forwarding in the Internet
  - BGP has a non-negligible economic impact on the business of the ISPs
- BGP monitoring is crucial for ISPs
  - several applications, from troubleshooting [Roughan04] to traffic engineering [Balon08] and SLA compliance [Feamster04]
Overview

- We identify the basic requirements for an ideal monitoring system
  - cost-effective system for the collection of all BGP messages as sent by neighboring ISPs
- We proposed a monitoring infrastructure
  - routers are mandated to copy TCP segments and an ad-hoc software collect and store them
    - exploit an already available feature
  - easily extendable to other protocols
- We experimentally evaluate our solution
for each destination, BGP routers receive a set of announcements
- each BGP router autonomously selects the best route among them
  - best routes control traffic flow
- ... and propagates it to its neighbors
Monitoring BGP Best Routes

- monitor BGP messages
  - quality
  - SLA
  - history
- check egress traffic flow
- ... but only on the primary link
Monitoring All BGP Routes

- What if link with A goes down?
- What if I change local-pref of some messages?
- What is the effective redundancy provided by B?
- What is the quality of announcements from B?

- monitor BGP messages on both links
  - quality
  - SLA
  - history

- X is enabled to analyze what-if scenarios, check SLA compliance for A and B, perform other value-added activities
An Ideal Monitoring System

- Collection of all the BGP routes
- Policy independent data
- Real-time collection
- Low impact on router resources
- Cost-efficient deployment
Existing Monitoring Systems

- A collector maintains iBGP peerings with routers that push data to it
  - Open source daemons (Quagga, Pyrt, ...)
  - Not possible to collect all the messages and policy independent data

- A separate management protocol can be used to pull information from routers
  - SNMP, screen scraping
  - Heavy impact on routers, can not be real-time

- BMP (comparison in the following)
Proposed Architecture

TCP segments with BGP data

clones all the TCP segments containing BGP data and sends them to the route collector

reconstruct the TCP stream, decodes BGP packets and store BGP packets in MRT.

cloned packets

cloned packets

cloned packets

TCP segments with BGP data
Border Routers

- border routers have to selectively clone incoming traffic to a destination
  - supported by major vendors on most routers
    - RITE/ERSPAN (Cisco), port mirroring (Juniper)
    - originally designed for supporting IDSes
  - cloned packets can typically be sent to the collector via VLANs or IP tunnels
  - management overhead is limited
Configuring Border Routers

```
access-list 100 permit tcp any any eq bgp

ip traffic-export profile <pr-name> 
    interface <vlan-interface> 
    incoming access-list 100 
    mac-address <addr>

interface <src-interface> 
    ip traffic-export apply <pr-name>
```
Route Collector

- the route collector has to **reconstruct the TCP stream** and to decode and store BGP messages
  - TCP segments are reordered and duplicated packets are silently ignored
  - prototype based on two Perl scripts
    - the first script reconstructs the TCP stream
    - the second script decodes and stores BGP packets in MRT
Testbed

Smartbits 600B

Cisco 7201 (DUT)

Bidirectional traffic flows

Route collector

medium-sized Italian ISP

Stream of Internet BGP updates

BGPd

BGPd

BGPd

BGPd

BGPd
Evaluation of our Solution

- We checked solution for correctness
  - no cloned packet was dropped
  - BGP messages were always correctly reconstructed and stored on disk
- We also evaluate performance of both border routers and route collector
  - throughput
  - CPU usage
  - latency
Evaluation: Border Routers

The diagram shows the performance degradation induced by SPC as a function of the packet rate (% of maximum packet rate). The graph includes data on frame loss, average latency, and average CPU usage. The performance degradation is indicated on the y-axis, ranging from -3% to 4%. The x-axis represents the packet rate, ranging from 20% to 60%.

Key observations:
- Frame loss: A sharp drop is observed at around 30% packet rate, reaching a minimum at around 40% before increasing again.
- Average latency: Remains relatively constant across the range of packet rates.
- Average CPU usage: Also remains relatively constant.

The graph indicates that the maximum packet rate without frame loss is achieved at around 40% packet rate.
Evaluation: Route Collector

- Transfer of five full BGP RIBs is replayed using tcpreplay at top speed

<table>
<thead>
<tr>
<th>elapsed time</th>
<th>original transfer</th>
<th>tcpreplay</th>
<th>stream reconstruction</th>
<th>BGP decoding and storage</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>&gt; 2 min</td>
<td>3.38 sec</td>
<td>2.6 sec</td>
<td>1.7 sec</td>
</tr>
</tbody>
</table>

- A single route collector can handle hundreds of border routers
  - processing a single prefix took about 5 μsec

- Performance can be further improved
## Comparison with Related Work

<table>
<thead>
<tr>
<th></th>
<th>BGP daemons (Quagga, Pyrt)</th>
<th>SNMP screen scraping</th>
<th>Our Approach and BMP</th>
</tr>
</thead>
<tbody>
<tr>
<td>non-best collection</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>policy independency</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>real-time</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>impact on router resources</td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>cost efficiency</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>
## Detailed Comparison with BMP

- **Our solution pushes complexity to the collector side**

<table>
<thead>
<tr>
<th></th>
<th>BMP</th>
<th>Our Approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>solution deployability</td>
<td>Internet draft, not widely supported yet</td>
<td>readily deployable</td>
</tr>
<tr>
<td>reliable delivery to the collector</td>
<td>yes, TCP connection</td>
<td>only check for lost packets</td>
</tr>
<tr>
<td>router performance</td>
<td>additional daemon, routers maintain a state</td>
<td>leverage optimized switching mechanisms</td>
</tr>
<tr>
<td>extendability to other protocols</td>
<td>extensions require software changes</td>
<td>easily extendable</td>
</tr>
</tbody>
</table>
Conclusions and Future Work

- what is the impact on production networks?
  - we exploit optimized packet copying mechanisms
  - experimental results are promising
  - a couple of companies already contacted us

- we plan to
  - deploy this solution in real networks
  - extend the approach to monitor all the control plane
  - integrate with iBGPlay: [www.ibgplay.org](http://www.ibgplay.org)
Thank you!!

- Questions?