Traffic engineering and GridFTP log analysis

Zhenzhen Yan, Z. Liu, M. Veeraraghavan          Chris Tracy, Chin Guok
University of Virginia                      ESnet
mvee@virginia.edu                           ctracy@es.net

Jan 17, 2013 Project web site:
http://www.ece.virginia.edu/mv/research/DOE09/index.html

Thanks to the US DOE ASCR for grants DE-SC002350 and DE-SC0007341 (UVA)
DOE for DE-AC02- 05CH11231 (ESnet)
NSF for grants, OCI-1127340 , OCI-1038058, and CNS-1116081 (UVA)

Thanks to Brian Tierney, Eric Pouyoul, Tareq Saif, Andy Lake & DOE: testbed

Thanks to Brent Draney, Jason Hick, NERSC, Yee-Ting Li, Wei Yang, SLAC, for
GridFTP DTN login access

Outline

• Hybrid network traffic engineering system (HNTES)
  - alpha-flow identification and redirection
  - alpha flows: high-rate, large sized flows
  - threshold: 1 GB in 1 min
• GridFTP related work
Two key findings

- Throughput on circuits could be lower than on IP-routed paths
  - IDC circuit provisioning includes policing
  - Policing interacts with TCP in a negative way
  - But even with RoCE, throughput limited by circuit rate (typically < link capacity)
- Primary cause of throughput variance
  - CPU and I/O resources of DTNs
  - Application arguments (e.g., -fast)
  - Not too dependent on the net (overprovisioned)

Hybrid network traffic engineering system (HNTES)
- **Intradomain** identification/redirection of alpha flows

- HNTES:
  - offline (e.g., nightly) analysis of NetFlow data at ingress routers
  - requests L3 circuits between ingress-egress router pairs from IDC
  - IDC sets firewall filters to direct alpha-flow packets to L3 circuits
Study router QoS configuration mechanisms

- Used DOE LIMAN testbed
- Hosts: high-performance diskpts

Router configurations

- Firewall filter:
  - alpha-flow based on /24 or /32 address prefixes (src and dst)

- Policing
  - classify out-of-profile packets as scavenger class and send to scavenger queue
  - Weighted Random Early Detection (WRED)

- Scheduling
  - Weighted fair queueing (WFQ)
  - Priority queueing (PQ)
Policing on ingress
scheduling on egress

- Dual goals:
  - reduce impact of alpha flows on real-time
delay-sensitive flows
  - allow alpha flows to enjoy high throughput

Compare 3 configurations

- 1-queue:
  - best-effort
  - all flows directed to same egress-side queue

- 2-queue: alpha and beta
  - scheduling-only (no policing)
    - WFQ + PQ
    - transmitter: shared in work conserving mode (non-strict)
    - buffer: strict partitioning

- 3-queue: alpha, beta, scavenger service (SS)
  - policing: > 1 Gbps sent to SS queue
  - scheduling: WFQ + PQ
Impact of alpha flows on real-time flows

- Impact on ping flow delay
  - significant in 1-queue configuration
  - negligible in 2-queue configuration
- Need separate virtual queue for alpha flow packets

UDP flow
- 3 Gbps

TCP flow
- 6 Gbps

Pings: 1 per sec
- Delay: 60 ms in 1-queue case
- Delay: 2.1 ms in 2-queue case

Impact of policing: 3-queue case causes TCP throughput to drop

- When UDP rate is increased from 0.5 to 1 Gbps, significant drop in TCP throughput
- Why? out-of-sequence packets
- TCP fast-retransmit/fast recovery algorithm causes sending rate to drop by half
- Worst when alpha queue allocation is only 10%
Key findings: HNTES

- Nodes generating alpha flows have static public IP addresses, and create repeated alpha flows
- Can leverage this fact in an offline HNTES design (nightly NetFlow analysis for alpha prefix ID determination)
- Configure ingress routers with firewall filters and WFQ/PQ 2-queue scheduling (alpha and beta); no policing
- ESnet5: WRED solution to policing
- IDC support requested: unspecified-rate circuits
  - implication: no policing

Z. Yan, M. Veeraraghavan, C. Tracy, C. Guok, “On how to provision Quality of Service (QoS) for large dataset transfers,” submitted to CTRQ 2013

Outline

- Hybrid network traffic engineering system
  - GridFTP
    - Usage log analysis software in R
    - GUI
    - Throughput variance:
      - competition for CPU and I/O within DTNs
      - less so for network resources
GridFTP usage log analysis & GUI

- Purpose: analyze and visualize GridFTP performance
- Obtained logs from
  - NERSC-ORNL (Sept. 2010)
  - NERSC-ANL (Mar. 4 - Apr. 22, 2012)
  - NCAR-NICS (2009-2011)
  - SLAC-BNL (Feb. 10 - Apr. 26, 2012)
- Analysis programs: coded in R
  - Session Analysis
  - Throughput Variance Analysis


NERSC - SLAC Experiments
To characterize causes of variance

- Path capacity: 10 Gbps
- At every hour
  - a memory to memory transfer runs from NERSC DTN to SLAC DTN for 30s
  - another one runs from SLAC to NERSC for 30s
  - prior to the two test runs, at the 59th minute every hour, the monitoring tools we developed are run to record CPU usage data and TCP traces
- SNMP data obtained from SLAC, NERSC, ESnet routers
NERSC-SLAC GridFTP mem2mem transfer throughput

There is variance in transfer throughput

GridFTP transfer throughput dependence on CPU time

6 Gbps

Transfer throughput

NERSC CPU % 80%

SLAC CPU % 20%
Network impact low

Correlation coefficients

GridFTP transfer throughput characteristics (shows variance even for mem2mem)

<table>
<thead>
<tr>
<th></th>
<th>Min</th>
<th>1st Qu.</th>
<th>Median</th>
<th>Mean</th>
<th>3rd Qu.</th>
<th>Max</th>
<th>CV</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oct</td>
<td>2196</td>
<td>4327</td>
<td>4744</td>
<td>4683</td>
<td>5065</td>
<td>6577</td>
<td>0.15</td>
</tr>
<tr>
<td>Dec</td>
<td>5316</td>
<td>3408</td>
<td>4463</td>
<td>4202</td>
<td>5056</td>
<td>6290</td>
<td>0.29</td>
</tr>
</tbody>
</table>

Correlation coefficients with CPU usage

<table>
<thead>
<tr>
<th></th>
<th>NERSC CPU</th>
<th>SLAC CPU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oct</td>
<td>0.82</td>
<td>0.90</td>
</tr>
<tr>
<td>Dec</td>
<td>0.95</td>
<td>0.98</td>
</tr>
</tbody>
</table>

Correlation coefficients with SNMP bytes on SLAC router links

<table>
<thead>
<tr>
<th></th>
<th>farm02a</th>
<th>core1</th>
<th>border2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oct</td>
<td>0.99</td>
<td>0.95</td>
<td>0.77</td>
</tr>
<tr>
<td>Dec</td>
<td>0.99</td>
<td>-0.01</td>
<td>0.25</td>
</tr>
</tbody>
</table>

Path changed?
Prototyping

- Our hypothesis is that we need a co-scheduling system for server and network resources to reduce throughput variance
- Suggestions for prior co-scheduling work that can be reused?

Summary

- Policing cuts throughput
  - if CPU and I/O resources are primary determinants of throughput, then do circuits hurt or help?
- Primary cause of throughput variance:
  - CPU and I/O resources of DTNs
  - Not so much the net (overprovisioned)
- Feedback/questions?
  - please email mvee@virginia.edu