Performance SDN

Yatish Kumar
CTO Corsa Technology

yatish@corsa.com
All you need is white box hardware and SDN to solve any networking problem!

It’s a switch
It’s a router
It’s a load balancer
It’s a scalable fabric
It’s powered by Linux!
Performance Fundamentals

The purpose of all networking boxes is to inspect packets and forward them to an output port.
Performance Fundamentals

The root of all evil is when we over subscribe a port causing congestion.
Performance Fundamentals

So we either add buffers to hold packets for a while, or we start throwing away packets.

The root of all evil is when we over subscribe a port causing congestion.
Performance Fundamentals

Throw Bandwidth to fix the problem
Does it really work for R&E traffic profile?

- Throughput is important, not bandwidth
- Throughput is an end-to-end problem. \textit{Excess} WAN bandwidth has limited impact

With loss, high performance beyond metro distances is essentially impossible

So we either add buffers to hold packets for a while, or we start throwing away packets

The root of all evil is when we over subscribe a port causing congestion

What would these guys do?
Keep packets or throw them away?
Performance Fundamentals

- If we have buffers we need a scheduler to pull packets out.
- So we either add buffers to hold packets for a while, or we start throwing away packets.
- The root of all evil is when we over subscribe a port causing congestion.
Performance Fundamentals

- and policers
- If we have buffers we need a scheduler to pull packets out
- and shapers

- So we either add buffers to hold packets for a while, or we start throwing away packets
- The root of all evil is when we over subscribe a port causing congestion
Performance Fundamentals

and policers

and meters

If we have buffers we need a scheduler to pull packets out

and shapers

So we either add buffers to hold packets for a while, or we start throwing away packets

Network Element

The root of all evil is when we over subscribe a port causing congestion

Does NOT have this stuff

Has this stuff
Performance Fundamentals

- OpenFlow MAT
- OpenFlow MAT

If we have buffers we need a scheduler to pull packets out

So we either add buffers to hold packets for a while, or we start throwing away packets

The root of all evil is when we over subscribe a port causing congestion
Just one more thing !!!
At 100 Gbits per second
According to Inder Monga we need 100 ms of buffering
Which means we need 10Gbits (Roughly 1 GigaByte) of buffering
Per 100G port!

WAN Buffers have to use DDR3

Each DDR3 roughly supports 50Gbits of read+write traffic
So we need 2 per 100G port in order to have sustained bandwidth
A 600 Gig WAN switch needs 12 DDR3 modules
12 Modules x 240 pins = 2880
Is a LOTTA PINS!

WAN Buffers have to use distributed processing

ToR ASIC
64 x10G Ports
10 Mbytes
What about TCAMs?
ToR White Box switches support roughly 10k TCAM entries
WAN routers need 1M IPv4 routes
Now we need external TCAMs too!

WAN Lookups have to use distributed processing
ToR discards packets
small internal TCAM
Head Of Queue Blocking

N inputs per module
M Modules
Q QoS
Q x N x M

8 QoS x 4 boxes x 50 egress ports = 1600 Queues
NOT 8 Queues !!
Leading to Papers in Sigcomm 2015

Congestion Control for Large-Scale RDMA Deployments

Their solution signals back to the servers for congestion control. Unfortunately this doesn't work in the WAN!!

So taking ToR CLOS fabrics and building WAN routers doesn't work very well!

ABSTRACT

Modern datacenter applications demand high throughput (40Gbps) and ultra-low latency (< 10 µs per hop) from the network, with low CPU overhead. Standard TCP/IP stacks cannot meet these requirements, but Remote Direct Memory Access (RDMA) can. On IP-routed datacenter networks, RDMA is deployed using RoCEv2 protocol, which relies on Priority-based Flow Control (PFC) to enable a drop-free network. However, PFC can lead to poor application performance due to problems like head-of-line blocking and unfairness. To alleviate these problems, we introduce DC-QCN, an end-to-end congestion control scheme for RoCEv2.
Match Action Hardware Implications

OpenFlow MAT

Network Element

What Is In the ASIC Matters Too

OOPS!

AHEM!

WAN
Inside The Corsa Dataplane

- Standard OpenFlow 1.3+ API
- P4 (and other) Compiled Pipelines
- Massive Sea of Gates to build 100G line rate hardware engines
- WAN not ToR Hardware Features
Key Attributes to Shop For

- Multiple match/action tables
- Millions of flow entries
- Large scale packet buffers
- Metering and QoS
- 100-Gigabit ports with full OpenFlow 1.3
- Extremely fast flow modifications per second 60k+ flow entries per second.
  (as opposed to 100 flow mods / sec on white box ToR)
Key Bottlenecks To Understand

- How big are the match tables?
- How much search bandwidth?
- What DoS Conditions Exist?
  - Too many packets?
  - Not enough control plane mips?
  - Tail drop under load?
- Match Action
  - MIPS
  - How many Packets Per Second?
- Buffer Memory Size and Bandwidth
  - How many ms?
  - What is the read/write bandwidth?
- How is QoS maintained in the fabric?
Thank You
Skip the cool aid. Get a stiff drink.