Considerations for
Benchmarking Virtual Networks
draft-bmwg-nvp-00

Samuel Kommu, skommu@vmware.com

Jacob Rapp, jrapp@vmware.com

Ben Basler, bbasler@vmware.com

July 2016 IETF 96 – Berlin

Considerations for
Benchmarking Network Virtualization Platforms - Overview

draft-bmwg-nvp-00

Why : Physical vs Virtual Network Platforms - Differences

MTU limited packets vs Higher Level Segments

Scope

Hypervisor Based Network Virtualization Platforms only – Not NFV

Considerations

Application Layer Benchmarks

Working closer to application layer segments and not low level packets

Server Hardware

Support for HW offloads (TSO / LRO / RSS)
Other Hardware offload benefits – Performance Related Tuning
Frame format sizes within Hypervisor

Scale Testing for New Application Architectures

New micro-Service type architectures

Documentation

System Under Test vs Device Under Test
Intra-Host (Source and destination on the same host)
Inter-Host (Source and Destination on different hosts – Physical Infra providing connectivity is part of SUT)

Hardware Switch vs Software Switch

Hardware Switching	Logical Switch/Logical Router etc.,
Works at lower layer packets	Works closer to application layer segments
Limited by ASIC/SoC	Limited mostly by CPU and Memory (only LB) • which is not really a limit with today's processor capabilities and memory capacity/speeds
Packet size limited by supported MTU • General Max supported is 9K	Packet size a function of RSS, TSO & LRO etc., • By default 65K
Multiport – often 48 or more	Generally 2 Ports/Server
Extending functionality through additional ASIC / FPGAs and Hardware	NIC Offloads Intel DPDK / Latest Drivers etc., SSL Offload with AES-NI (Intel and AMD)

TSO for VXLAN Traffic

NIC Based TSO

CPU Based TSO

VM

MAC	IP	TCP	Payload

MAC	IP	TCP	Payload

65K

MTU

MAC	IP	UDP	VXLAN	MAC	IP	TCP	Payload

MAC	IP	UDP	VXLAN	MAC	IP	TCP	Payload

MAC	IP	UDP	VXLAN	MAC	IP	TCP	Payload

| MAC	IP	UDP	VXLAN	MAC	IP	TCP	Payload

Physical Fabric
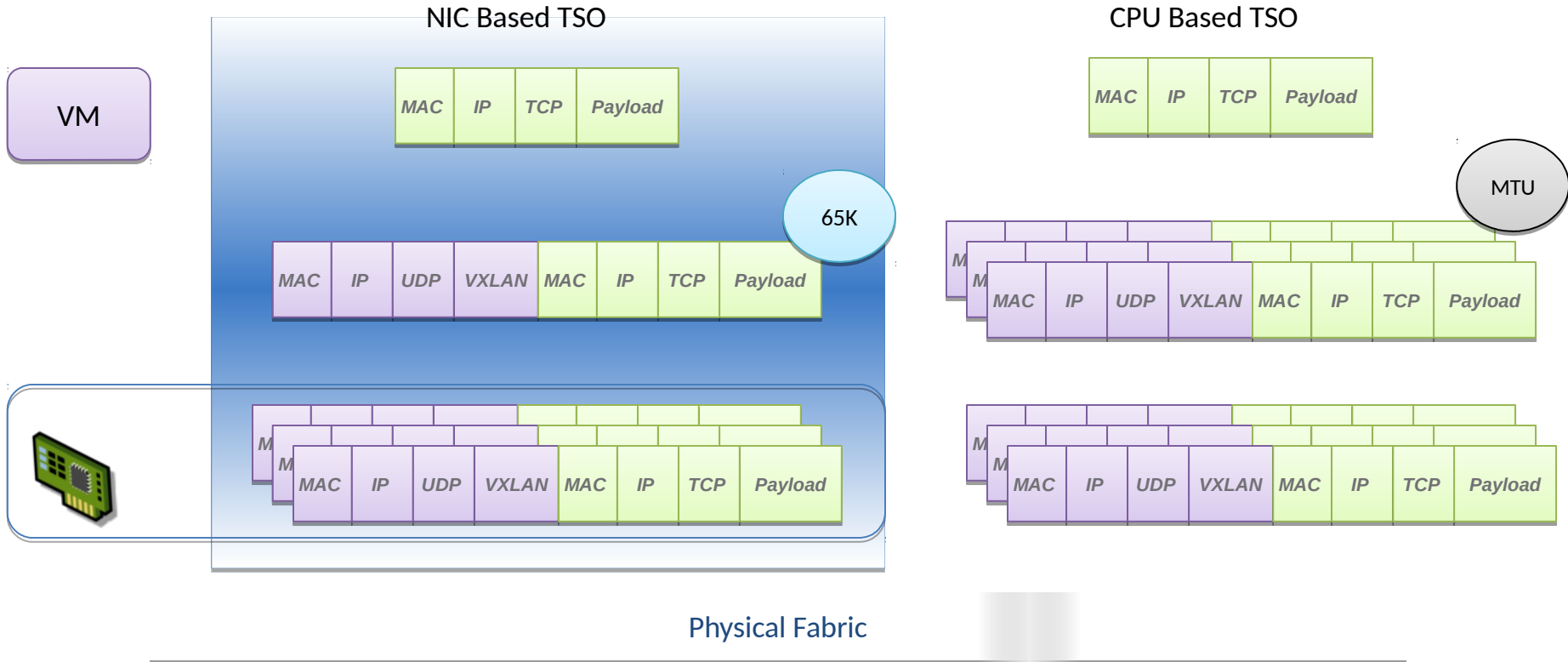
Receive Side Scaling (RSS)
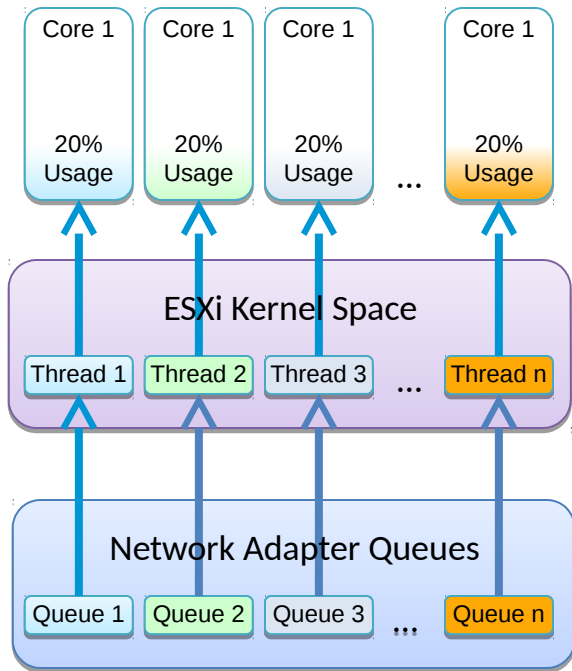
- With Receive Side Scaling Enabled

  - Network adapter has multiple queues to handle receive traffic

  - 5 tuple based hash (Src/Dest IP, Src/Dest MAC and Src Port) for optimal distribution to queues

  - Kernel thread per receive queue helps leverage multiple CPU cores

Page Size and Response Times

Average Page Size   2MB

                                               http://httparchive.org/trends.php

Average HTML Content   56KB

Web Response Times   200ms   https://developers.google.com/speed/docs/insights/Server

Memcached Response Time   Sub 1ms   https://code.google.com/p/memcached/wiki/NewPerformance

Example Test Methodology

- Application level throughput using Apache Benchmark
  - ~2m file sizes based on http://httparchive.org/trends.php
    - Images tend to be larger
    - Page content tends to be smaller
- Application latency with Memslap
  - Standard settings
- iPerf
- Avalanche