Red Hat

An encounter with OpenvSwitch hardware offload on 100GbE smartNIC!!

OVS/OVN 2019 Fall
10th-11th Dec 2019

Haresh Khandelwal
Principal
Software Engineer

Pradipta Sahoo
Principal
Technical Support Engineer

Context

- To provide an operator's outlook from the territory of OpenvSwitch(TC-HW-Offload) when enabled on smartNIC
- To identify challenges of 100GbE in commodity x86 servers and possible fine-tuning
- To measure the throughput numbers when offload plugged to cloud infrastructure
- To finally share observations/findings

Red Hat

Deployment specifications

Items	Description
Server	Dell PowerEdge R740
CPU	Intel(R) Xeon(R) Silver 4110 CPU @ 2.10GHz 8 CPU Cores * 2 NUMA
RAM	64GB: 8 * 8GB DIMMs * 2 NUMA nodes Type: DDR4 Speed: 2666 MT/s Minimum Voltage: 1.2 V Maximum Voltage: 1.2 V Configured Voltage: 1.2 V
BIOS	BIOS Information Vendor: Dell Inc. Version: 2.3.10 Release Date: 08/15/2019
NIC	Dual Port: Mellanox Technologies MT28800 Family [ConnectX-5 Ex] Subsystem: Mellanox Technologies Device 0026
Cloud Software	Red Hat OpenStack 13 (Queen)
Operating System Virtual Switch	Red Hat Enterprise Linux release 7.7 OpenvSwitch 2.11
Kernel Version	3.10.0-1062.4.1.el7.x86_64
GCC Version	4.8.5 20150623 (Red Hat 4.8.5-39)
Mellanox Firmware Version	16.25.4062 (DEL0000000004)
Mellanox OFED Driver version	MLNX_OFED_LINUX-4.7-1.0.0.1
DPDK Version	18.11
TREX Version	v2.65
Dell Switch System Type	S5048F-ON Dell EMC Real Time Operating System Software

Red Hat

Benchmark methodology & tools

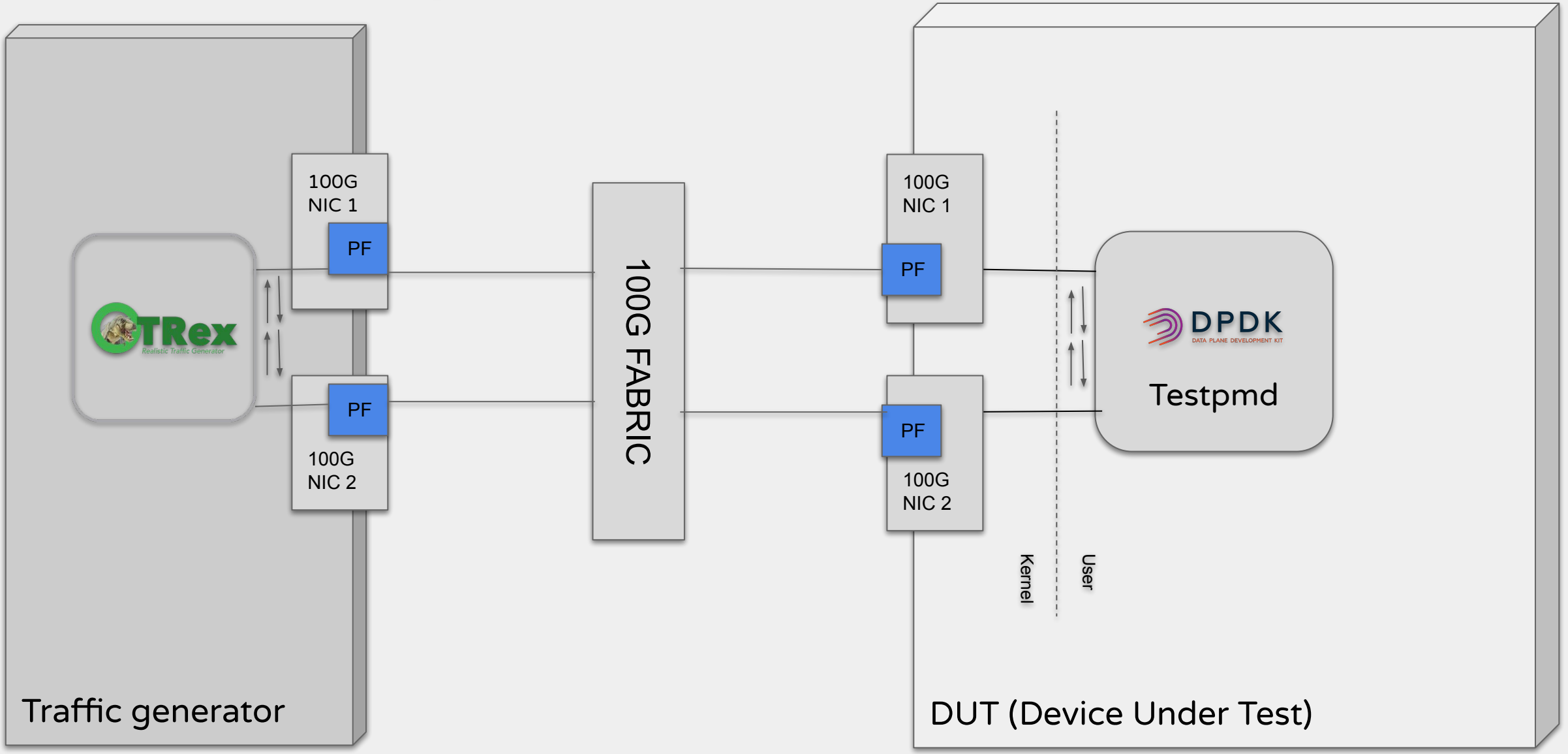
- Open source & low cost
- High scale of realistic traffic up to 200 Gb/sec
- Large scale - Supports up to 20 million packets per second (mpps)
- Multiple streams support
- Ability to change any field inside the packet (e.g. src_ip = 10.0.0.1-10.0.0.255)
- Interactive support - Console, GUI
- Per stream statistics, latency and Jitter

TestPMD

- open source & low cost
- shipped as part of the Data Plane Development Kit.
- used to test the DPDK in a packet forwarding mode
- supports user interactive mode
- supports Multi Core processing

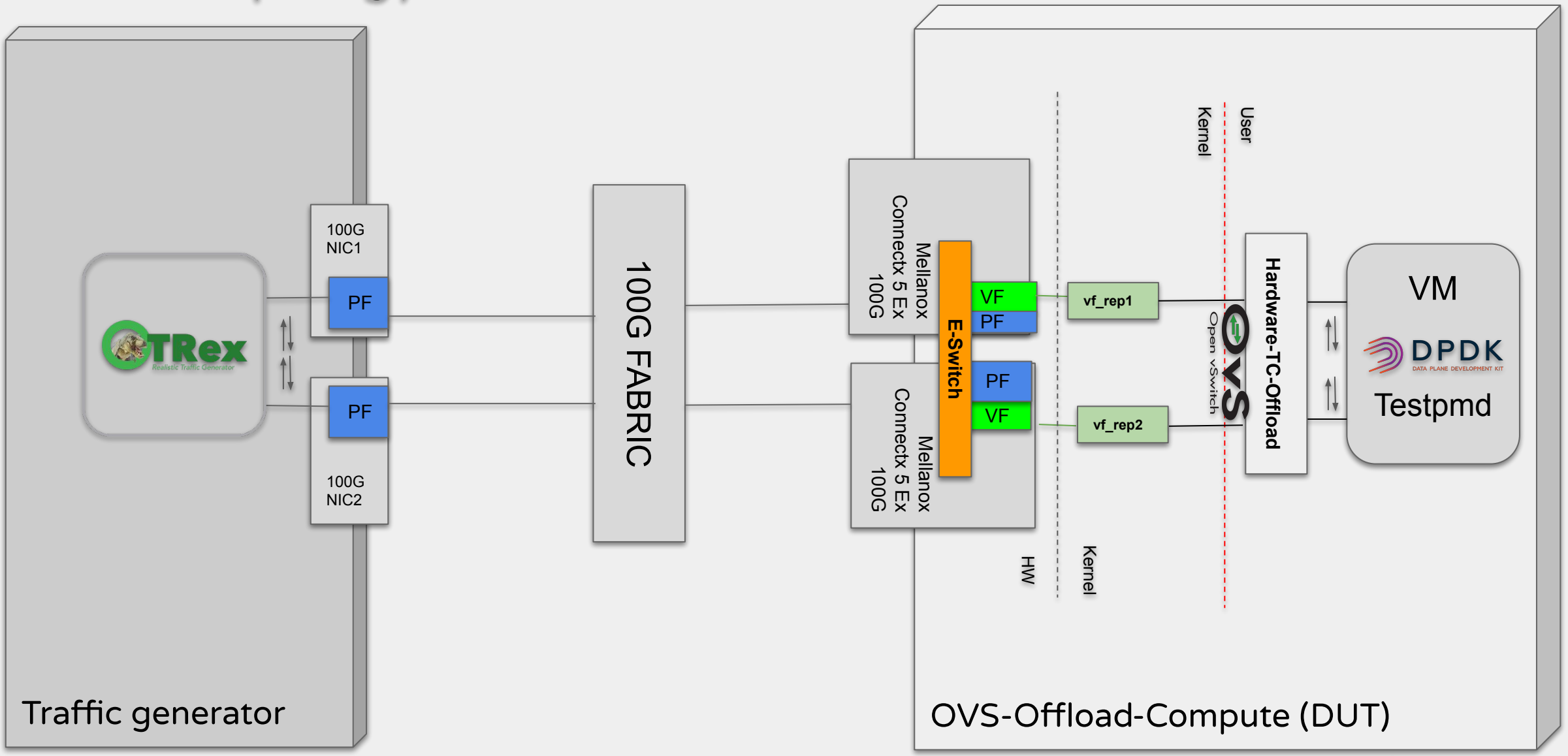
Red Hat

Topology - bare metal network



Red Hat

Topology - OVS hardware offload



Red Hat

Trex Configuration

- Executed in TRex v2.65 with DPDK 18.11
- Tuned the Trex server config file with hugepages and descriptor
- Included all the CPU cores from both socket for packet processing
- Run the trex server without “OFED” dependencies.

- Traffic Profile:
  - Bi directional
  - Fix SRC MAC
  - Fix DST MAC
  - Variable SRC IP
  - Variable DST IP
  - Fix UDP SRC Port
  - Fix UDP DST Port

```
# cat /etc/trex_cfg.yaml
- port_limit: 2
  version: 2
  interfaces: ['5e:00.0', '5e:00.1']
  port_info:
      - dest_mac          :   fa:16:3e:5c:f6:e9
            src_mac       :   xx:xx:xx:xx:xx:xx
            ip            :   192.168.100.3
            default_gw    :   192.168.200.5
      - dest_mac          :   fa:16:3e:2f:08:58
            src_mac       :   yy:yy:yy:yy:yy:yy
            ip            :   192.168.200.5
            default_gw    :   192.168.100.3
  limit_memory: 18432  ### Huge page configuration.
  rx_desc: 4096
  tx_desc: 4096
  port_bandwidth_gb : 100
  platform:
      master_thread_id: 0  ## Non-isolated Cores from NUMA 0
      latency_thread_id: 1 ## Isolated Cores from NUMA 1
      dual_if:
      - socket: 0
      threads: [2,4,6,8,10,12,14,3,5,7,9,11,13,15]   ## Isolated Cores from two NUMA
```

```
./t-rex-64 -i -c 14 --cfg /etc/trex_cfg.yaml -v 7 --no-ofed-check
```

```
-Per port stats table
      ports |               0 |               1
 -----------------------------------------------------------------------------------------
   opackets |      6517638140 |      6517732544
     obytes |    417128841200 |    417134882816
   ipackets |      6514280601 |      6514504457
     ibytes |    416913959964 |    416928285248
    ierrors |               0 |               0
    oerrors |               0 |               0
      Tx Bw |      12.82 Gbps |      12.78 Gbps

-Global stats enabled
 Cpu Utilization : 100.0  %  3.7 Gb/core
 Platform_factor : 1.0
 Total-Tx        :       25.60 Gbps
 Total-Rx        :       25.60 Gbps
 Total-PPS       :       50.00 Mpps
 Total-CPS       :        0.00  cps

 Expected-PPS    :        0.00  pps
 Expected-CPS    :        0.00  cps
 Expected-BPS    :        0.00  bps

 Active-flows    :           0  Clients :        0   Socket-util : 0.0000 %
```

Red Hat

Starting TestPMD …..

- Run TestPMD (DPDK 18.11) with forwarding mode.
- Aligned hugepage memory (--socket-mem) and set the PMD affinity (--nb-cores) to specific NUMA node to avoid the context switching.
- Tested with single and dual cores for for PMD cycle.

```
# dpdk-devbind --status-dev net

Network devices using kernel driver
===================================
0000:00:05.0 'MT28800 Family [ConnectX-5 Ex Virtual Function] 101a' if=ens5 drv=mlx5_core unused=
0000:00:06.0 'MT28800 Family [ConnectX-5 Ex Virtual Function] 101a' if=ens6 drv=mlx5_core unused=
```

```
testpmd -l 0,1 -n 4 --huge-dir=/dev/hugepages -w 00:XX.0 -w 00:XX.0 --socket-mem 16384,0 -- -i --nb-cores=1
--eth-peer=0,xx:xx:xx:xx:xx:xx --eth-peer=1,yy:yy:yy:yy:yy:yy --forward-mode=mac --rxd=2048 --txd=2048 --rxq=1 --txq=1
--socket-num=0 --burst=64 --mbcache=512 -a --rss-udp --no-numa --disable-crc-strip
```

```
testpmd -l 0,1,2 -n 4 --huge-dir=/dev/hugepages -w 00:XX.0 -w 00:XX.0 --socket-mem 16384,0 -- -i --nb-cores=2
--eth-peer=0,xx:xx:xx:xx:xx:xx --eth-peer=1,yy:yy:yy:yy:yy:yy --forward-mode=mac --rxd=2048 --txd=2048 --rxq=2 --txq=2
--socket-num=0 --burst=64 --mbcache=512 -a --rss-udp --no-numa --disable-crc-strip
```

```
# pidstat -t -p `pidof testpmd` 5
Linux 3.10.0-1062.el7.x86_64 (testpmd)     11/18/2019     _x86_64_     (14 CPU)

07:25:53 AM   UID       TGID       TID    %usr    %system  %guest %CPU   CPU  Command
07:25:58 AM 0     3309         -     100.00     0.00    0.00  100.00    1  testpmd
07:25:58 AM 0     -     3309    0.00     0.00    0.00    0.00     1  |__testpmd
07:25:58 AM 0     -     3310    0.00     0.00    0.00    0.00     7  |__eal-intr-thread
07:25:58 AM 0     -     3311    0.00     0.00    0.00    0.00     7  |__rte_mp_handle
07:25:58 AM 0     -     3312  100.00     0.00    0.00  100.00    2  |__lcore-slave-2
07:25:58 AM 0     -     3313  100.00     0.00    0.00  100.00    3  |__lcore-slave-3
```

Red Hat

Tuning parameters

- **BIOS Settings:**
  - **Processor Setting:**
    - **Logical Processor: Disabled**
    - CPU Interconnect Speed: Maximum Data Rate
    - Dell Controlled Turbo: Enabled
  - System Profile: **Performance**
    - CPU Power Management: Maximum Performance
    - Memory Frequency: Maximum Performance
    - **Turbo Boost: Enabled** **
    - C States: Disabled
    - Write Data CRC: Disabled
    - Uncore Frequency: Maximum
    - Energy Performance Policy: Performance
    - Monitor/Mwait: Disabled
    - CPU Interconnect Bus Link Power Management: Disabled
    - Power-Management: Disabled
  - **Thermal Mode: Performance**

- **Host and Guest CPU Isolation without CPU siblings**

```
# cat proc/cmdline
BOOT_IMAGE=/boot/vmlinuz-3.10.0-1062.4.1.el7.x86_64
root=UUID=b1755736-027f-440c-a0c5-88afa6dce659 ro console=tty0
console=ttyS0,115200n8 crashkernel=auto rhgb quiet default_hugepagesz=1GB
hugepagesz=1G hugepages=44 iommu=pt intel_iommu=on
isolcpus=2,4,6,8,10,12,14,3,5,7,9,11,13,15 spectre_v2=off nopti
--------

# tuned-adm active
Current active profile: cpu-partitioning ?
```

- **Memory Huge-Page allocation**

```
Node 0 AnonHugePages:   86016 kB
Node 0 HugePages_Total: 22
Node 0 HugePages_Free:  2
Node 0 HugePages_Surp:  0
Node 1 AnonHugePages:   36864 kB
Node 1 HugePages_Total: 22
Node 1 HugePages_Free:  2
Node 1 HugePages_Surp:  0
```

** Experimental

Cont...100G NIC Optimization

Resized NIC Ring buffer with max hardware limitation

```
Current hardware settings:
RX:     8192
RX Mini:      0
RX Jumbo:     0
TX:     8192
```

Disabled Advertised pause frame and auto-negotiation

```
Advertised pause frame use: No
Advertised auto-negotiation: No
```

Disabled NIC Flow Control

```
Autonegotiate:    off
RX:     off
TX:     off
```

Switched "Completion Queue Events (CQE)" mode from “**BALANCED**” to "**AGGRESSIVE**"

```
# mlxconfig -d 5e:00.0 set CQE_COMPRESSION=1
```

Change PCI MaxReadReq

```
# setpci -s 5e:00.0 68.w=3xxx
```

Red Hat

Cont… CPU frequency optimization

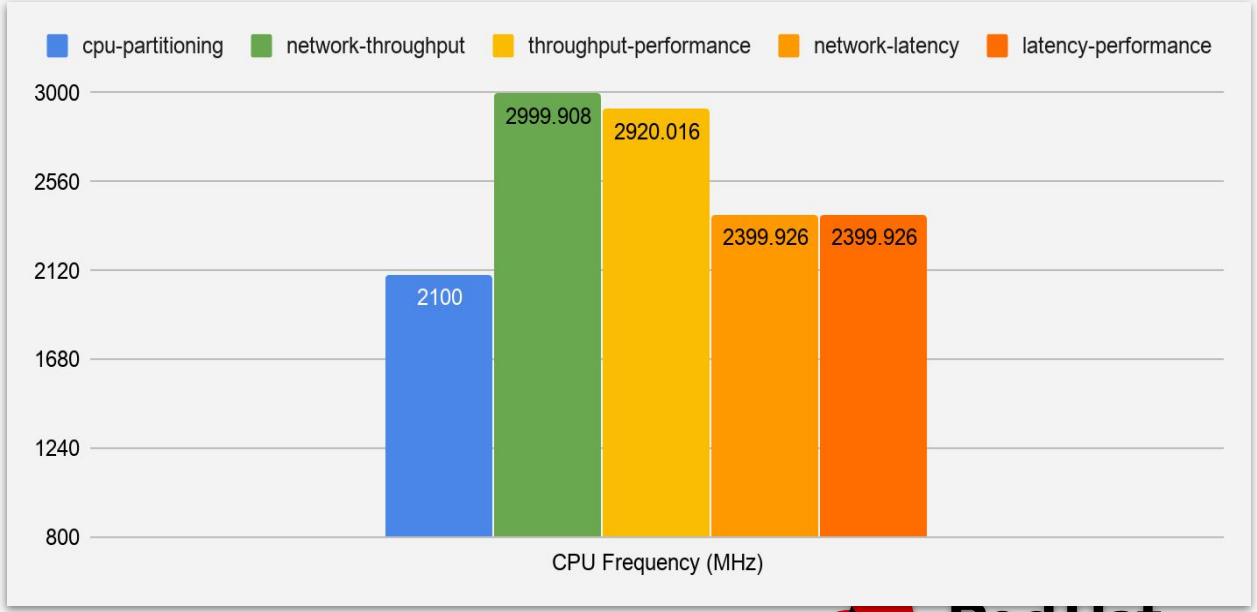
- In host level, “cpu-partitioning” profile gets deterministic frequency rate but not efficient Guest vCPU to process packets from datapath layer.

- Compute Host, Tuned profile with “network-throughput” **
  - CPU Scaling_Governer : Performance
  - CPU Scaling_Min_Performance: 800 MHz
  - CPU Scaling_Max_Performance: 3.00 GHz

- Deterministic Frequency rate:
  - cpu-partitioning (average)
  - network-throughput (maximum)

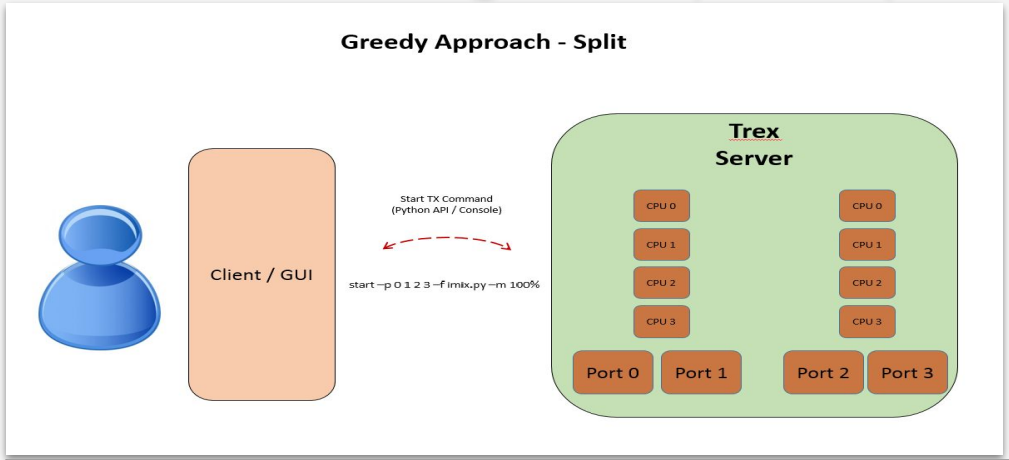
- The other tuned profiles provides better frequency without deterministics clock rate.

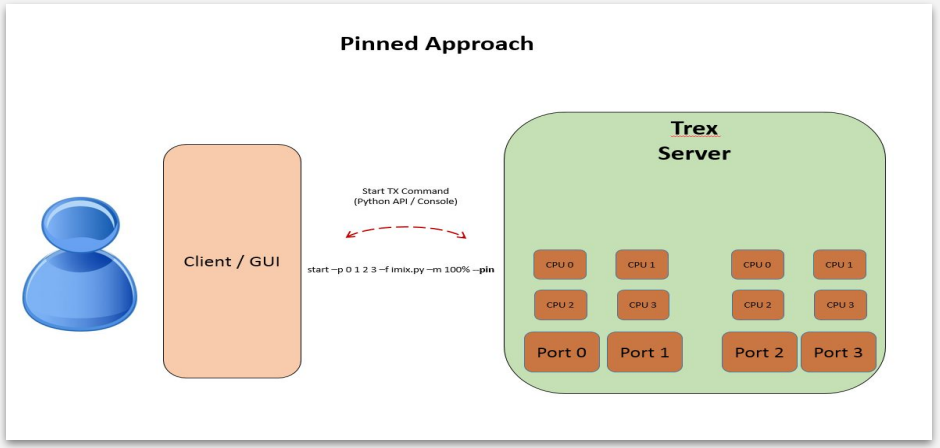
```
# tuned-adm active
Current active profile: network-throughput

# cpupower frequency-info
analyzing CPU 0:
  driver: intel_pstate
  CPUs which run at the same hardware frequency: 0
  CPUs which need to have their frequency coordinated by software: 0
  maximum transition latency:  Cannot determine or is not supported.
  hardware limits: 800 MHz - 3.00 GHz
  available cpufreq governors: performance powersave
  current policy: frequency should be within 800 MHz and 3.00 GHz.
                  The governor "performance" may decide which speed to use
                  within this range.
  current CPU frequency: 2.73 GHz (asserted by call to hardware)
  boost state support:
      Supported: yes
      Active: yes
```



** Experimental

Cont...TRex greedy v/s pinned CPU cycle?






```
start -f stl/offload-bech.py --force -t fsize=64 --total -m 90mpps
```

Global Statistitcs

```
connection   : localhost, Port 4501                total_tx_L2  : 25.56 Gb/sec
version      : STL @ v2.64                         total_tx_L1  : 33.55 Gb/sec
cpu_util.    : 100.0% @ 14 cores (14 per dual port)   total_rx     : 25.56 Gb/sec
rx_cpu_util. : 0.0% / 0 pkt/sec                    total_pps    : 49.93 Mpkt/sec
async_util.  : 0.06% / 1.51 KB/sec                 drop_rate    : 0 b/sec
total_cps.   : 0 cps/sec                           queue_full   : 0 pkts
```

Cpu Util(%)

Thread	Avg	Latest	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10	-11	-12	-13	-14
0 (0,1)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
1 (0,1)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
2 (0,1)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
3 (0,1)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
4 (0,1)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
5 (0,1)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
6 (0,1)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
7 (0,1)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
8 (0,1)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
9 (0,1)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
10 (0,1)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
11 (0,1)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
12 (0,1)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
13 (0,1)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100

Mbuf Util

	64b	128b	256b	512b	1024b	2048b	4096b	9kb	RAM(MB)
Total:	229320	114660	90090	90090	90090	82030	14336	14336	546
Used:									
Socket 0:	404	0	0	0	0	8320	0	0	17
Percent:	0%	0%	0%	0%	0%	10%	0%	0%	

```
start -f stl/offload-bech.py --force -t fsize=64 --total -m 90mpps --pin
```

Global Statistitcs

```
connection   : localhost, Port 4501                total_tx_L2  : 49.89 Gb/sec
version      : STL @ v2.64                         total_tx_L1  : 65.48 Gb/sec
cpu_util.    : 100.0% @ 14 cores (14 per dual port)   total_rx     : 26.96 Gb/sec
rx_cpu_util. : 0.0% / 0 pkt/sec                    total_pps    : 97.44 Mpkt/sec
async_util.  : 0.05% / 1.55 KB/sec                 drop_rate    : 22.93 Gb/sec
total_cps.   : 0 cps/sec                           queue_full   : 609,143,787 pkts
```

Cpu Util(%)

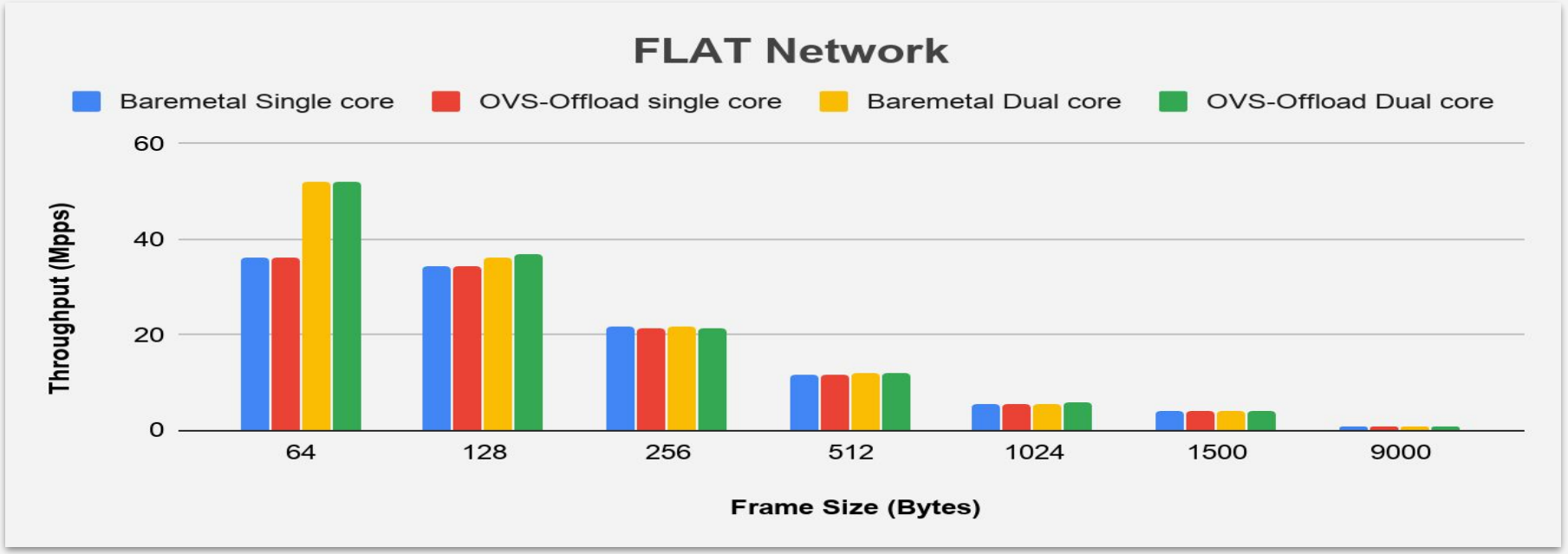
Thread	Avg	Latest	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10	-11	-12	-13	-14
0 (0)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
1 (1)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
2 (0)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
3 (1)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
4 (0)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
5 (1)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
6 (0)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
7 (1)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
8 (0)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
9 (1)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
10 (0)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
11 (1)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
12 (0)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
13 (1)	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100

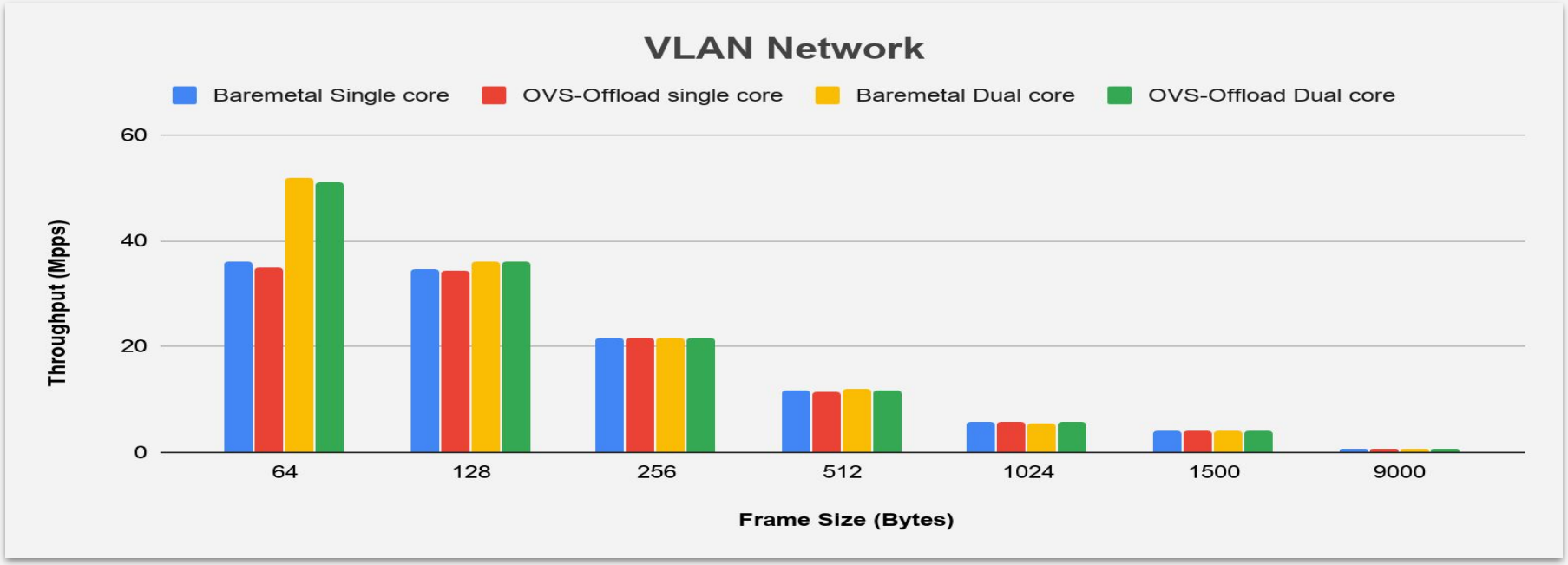
Mbuf Util

	64b	128b	256b	512b	1024b	2048b	4096b	9kb	RAM(MB)
Total:	229320	114660	90090	90090	90090	82030	14336	14336	546
Used:									
Socket 0:	256	0	0	0	0	8320	0	0	17
Percent:	0%	0%	0%	0%	0%	10%	0%	0%	

Red Hat

Performance benchmark





Red Hat

Observations/Findings

- Stable offload flows with 53 mpps on 64 bytes frame size for 18 hours and with **no traffic disruption**
- **~24.5ms** latency observed from end to end for first packet. Further it took ~200 us
- Flow counters get **overrun**
- OVS offload debuggers doesn't help i.e. **“netdev-tc-offload”** and also doesn’t generate any event log. Similar case with vendor drivers
- Feature harness i.e. security group rules
- Limited design/architect documentation in public forum
- CPU clock rate is critical while designing 100 GbE bandwidth network

Cont… flow inconsistency in TC classifier

- With Unidirectional test, TC offload flow break in (5-7 min) irrespective to any line rate. (~2+Mpps)
- Datapath flow missed from TC flower classifier and switch the flag to "**not_in_hw**"
- This issue has appeared when the neutron port bind with security group (iptables_hybrid).
- Sometimes the flow break only from ingress path.
- No events has triggered neither ovs-vswitchd and system logs.
- Raised and working on this bug: https://bugzilla.redhat.com/show_bug.cgi?id=1776136

```
# tc -s filter show dev p3p1 ingress
filter protocol LLDP pref 2 flower chain 0
filter protocol 802.1Q pref 3 flower chain 0
filter protocol 802.1Q pref 3 flower chain 0 handle 0x1
  vlan_id 501
  vlan_prio 1
  vlan_ethtype ip
  dst_mac fa:16:3e:5c:f6:e9
  src_mac xx:xx:xx:xx:xx:xx
  eth_type ipv4
  ip_flags nofrag
  in_hw
       action order 1:  vlan pop pipe
...
```

```
# tc -s filter show dev p3p1 ingress
filter protocol LLDP pref 2 flower chain 0
filter protocol 802.1Q pref 3 flower chain 0
filter protocol 802.1Q pref 3 flower chain 0 handle 0x1
  vlan_id 501
  vlan_prio 1
  vlan_ethtype ip
  dst_mac fa:16:3e:5c:f6:e9
  src_mac xx:xx:xx:xx:xx:xx
  eth_type ipv4
  ip_flags nofrag
  not_in_hw
       action order 1:  skbedit ptype host pipe
...
```

Red Hat

Cont… Port statistics get over-run

- OVS offload datapath counter usually set with 10 digit. "**packets:2125346404**"
- During the test (e.g., 53+mpps), the datapath packet counter statistics get over run once it reached out 5-Billion (~5300000000) and reset after that
- We would expect the counter should reach to 9-Billion (~9999999999) before reset.
- Also, need to think on ideal size of packet counter when it is running over 100G network bandwidth.
- Raised following and we are working on it: https://bugzilla.redhat.com/show_bug.cgi?id=1776816

```
# ovs-dpctl dump-flows -m type=offloaded
2019-11-16T19:19:00Z|00001|dpif_netlink|INFO|The kernel module does not support meters.
..... Output omitted ......
ufid:200a1be6-a48f-453c-9208-a07f2c132475,
skb_priority(0/0),skb_mark(0/0),in_port(p3p1),packet_type(ns=0/0,id=0/0),eth(src=xx:xx:xx:xx:xx:xx,dst=fa:16:3e:5c:f6:e9),eth_type(0x8100),vlan(vid
=501,pcp=1),encap(eth_type(0x0800),ipv4(src=0.0.0.0/0.0.0.0,dst=0.0.0.0/0.0.0.0,proto=0/0,tos=0/0,ttl=0/0,frag=no)), packets:2126050235,
bytes:3498690562028, used:0.470s, offloaded:yes, dp:tc, actions:pop_vlan,p3p1_1
..... Output omitted ......
ufid:20d44421-d6c0-4909-ab4a-402e2f4cfa3e,
skb_priority(0/0),skb_mark(0/0),in_port(p3p1_1),packet_type(ns=0/0,id=0/0),eth(src=fa:16:3e:5c:f6:e9,dst=xx:xx:xx:xx:xx:xx),eth_type(0x0800),ipv4(s
rc=0.0.0.0/0.0.0.0,dst=0.0.0.0/0.0.0.0,proto=0/0,tos=0/0,ttl=0/0,frag=no), packets:2124747635, bytes:3730409641488, used:0.470s, offloaded:yes,
dp:tc, actions:push_vlan(vid=501,pcp=0),p3p1
..... Output omitted ......
```

Red Hat

Cont...How critical is PCIe lane for performance?

- The maximum possible PCIe bandwidth is calculated by multiplying the PCIe width and speed.
- From that number we reduce ~1Gb/s for error correction protocols and the PCIe headers overhead.

Maximum PCIe Bandwidth = SPEED x WIDTH x (1 - ENCODING) - 1Gb/s.

Current PCIe Bandwidth = 8 x 8 x (1 - 2/130) - 1G = 64G x 0.985 - 1G = ~62Gb/s

```
PCI Device Name: 5e:00.0 Ethernet controller: Mellanox Technologies MT28800 Family [ConnectX-5 Ex]
Status: Warning
Current Firmware Version: Warning the current Firmware- 16.25.4062, is not latest - N/A
PSID: XXXXXXXXX
Desired PCIe Generation: 4
Current PCIe Generation: 3
Desired Speed: 8.0
Current Speed: 8.0
Desired Width: x16.0
Current Width: x8.0
Desired Payload Size: 256.0
Current Payload Size: 256.0
Desired Max Read Request: 4096.0
Current Max Read Request: 1024.0
```

```
Capabilities: [60] Express (v2) Endpoint, MSI 00
..... Output omitted ......
 LnkCap:    Port #0, Speed 16GT/s, Width x16, ASPM not supported
       ClockPM- Surprise- LLActRep- BwNot- ASPMOptComp+
 LnkCtl:    ASPM Disabled; RCB 64 bytes Disabled- CommClk+
       ExtSynch- ClockPM- AutWidDis- BWInt- AutBWInt-
 LnkSta:    Speed 8GT/s, Width x8, TrErr- Train- SlotClk+ DLActive- BWMgmt- ABWMgmt-
..... Output omitted ......
 LnkCtl2: Target Link Speed: 16GT/s, EnterCompliance- SpeedDis-
        Transmit Margin: Normal Operating Range, EnterModifiedCompliance- ComplianceSOS-
        Compliance De-emphasis: -6dB
..... Output omitted ......
```

Red Hat

Reference Resources: templates,software ...etc.

OpenStack OVS Offload Templates:

- https://github.com/HareshKhandelwal/RHOSP13_Offload_VxLAN_VLAN

Software Details:

- Red Hat Enterprise Linux Server 7.7
- Red Hat OpenStack Platform release 13
- MLNX_OFED_LINUX-4.7-1.0.0.1
- Trex v2.65
- DPDK 18.11
- OpenvSwitch 2.11

Trex Traffic Profile: https://github.com/pradiptapks/nfv-sdn-troubleshooting/blob/master/trex/offload-bech.py

CLI list: https://beta.etherpad.org/p/ovs-con-2019-offload

Reference link:

- https://fast.dpdk.org/doc/perf/DPDK_18_11_Mellanox_NIC_performance_report.pdf
- https://trex-tgn.cisco.com/trex/doc/trex_stateless.html#_performance_tweaking
- https://community.mellanox.com/s/article/understanding-pcie-configuration-for-maximum-performance
- https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/6/html/power_management_guide/tuned-adm
- https://trex-tgn.cisco.com/trex/doc/trex_manual.html#_platform_yaml_cfg_argument
- https://trex-tgn.cisco.com/trex/doc/trex_faq.html
- http://doc.dpdk.org/guides/nics/mlx5.html#mlx5-offloads-support

Thank you !!
&
Questions..

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.


linkedin.com/company/red-hat


youtube.com/user/RedHatVideos


facebook.com/redhatinc


twitter.com/RedHat