

25-Feb-16

Accelerating OpenStack “Neutron”

MOIZ ARIF

Presenter Profile

- ▶ Working for more than 2 years on OpenStack
- ▶ Working on Big Data Analytics for over 6 months now
- ▶ Successfully delivered several OpenStack Workshops and talks

What to expect from this Talk

- ▶ We will talk about OpenStack Neutron
- ▶ Explore ways to increase network performance
- ▶ Understand how different technologies can be used with OpenStack

What seems to be the problem Officer?

- ▶ Multi core CPU chips

What seems to be the problem Officer?

- ▶ Multi core CPU chips
- ▶ HW based CPU virtualization

What seems to be the problem Officer?

- ▶ Multi core CPU chips
- ▶ HW based CPU virtualization
- ▶ Hundreds of Gigabytes of RAM

What seems to be the problem Officer?

- ▶ Multi core CPU chips
- ▶ HW based CPU virtualization
- ▶ Hundreds of Gigabytes of RAM
- ▶ Faster Disks

What seems to be the problem Officer?

- ▶ But
- ▶ End users still experience slow application response time
- ▶ Unpredictable application performance



Identify Performance Bottlenecks

- ▶ Driver level bottlenecks



Identify Performance Bottlenecks

- ▶ Driver level bottlenecks
- ▶ Virtualized Environment bottlenecks



Identify Performance Bottlenecks

- ▶ Driver level bottlenecks
- ▶ Virtualized Environment bottlenecks
- ▶ Virtual Machine bottlenecks



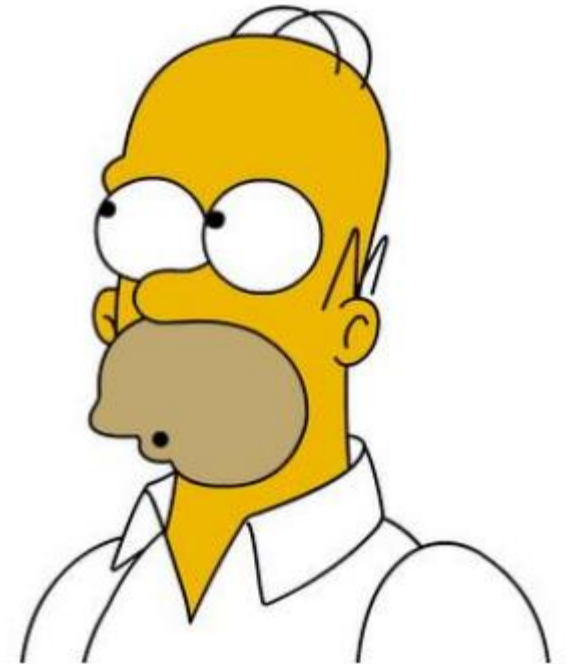
Identify Performance Bottlenecks

- ▶ Driver level bottlenecks
- ▶ Virtualized Environment bottlenecks
- ▶ Virtual Machine bottlenecks
- ▶ Network infrastructure



How to Increase Performance ?

- ▶ Replace linux kernel
- ▶ Enable Hugepages
- ▶ Hyper Threading (HT)
- ▶ CPU Pinning
- ▶ Use opensource/closed source technologies



OpenStack Neutron

- ▶ Neutron handles networking in OpenStack

OpenStack Neutron

- ▶ Neutron handles networking in OpenStack
- ▶ Can handle rich networking topologies and advanced network policies in the cloud

OpenStack Neutron

- ▶ Neutron handles networking in OpenStack
- ▶ Can handle rich networking topologies and advanced network policies in the cloud
- ▶ Pluggable open architecture

OpenStack Neutron

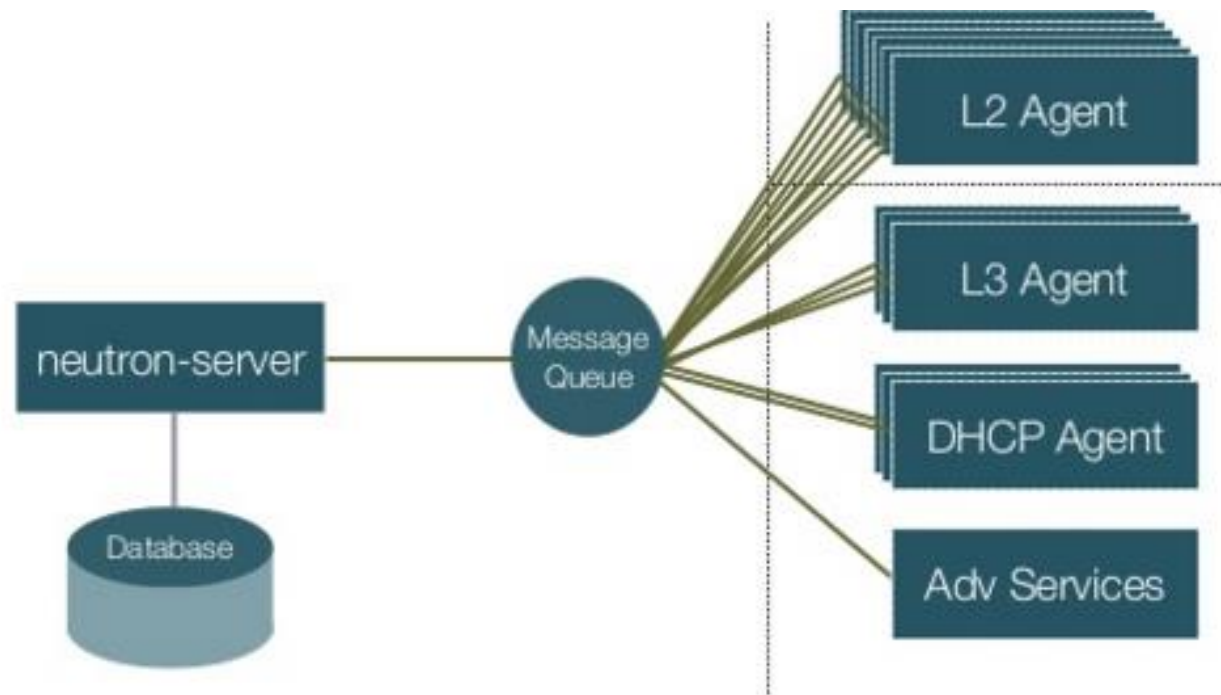
- ▶ Neutron handles networking in OpenStack
- ▶ Can handle rich networking topologies and advanced network policies in the cloud
- ▶ Pluggable open architecture
- ▶ Has a small simplified core

OpenStack Neutron

- ▶ Advanced features of OpenStack Neutron:
 - ▶ Load Balancer as a service (LBaaS)
 - ▶ VPN as a service (VPNaaS)
 - ▶ Firewall as a service (FWaaS)
 - ▶ Distributed Virtual Router (DVR)

OpenStack Neutron

► Architecture



OpenStack Neutron

- ▶ Plugins available for:
 - ▶ Cisco
 - ▶ BigSwitch
 - ▶ Brocade
 - ▶ IBM
 - ▶ Nicira NVP
 - ▶ NEC
 - ▶ PLUMgrid
 - ▶ And many more

SR-IOV Technology

- ▶ SR-IOV stands for Single Root Input/Output Virtualization

SR-IOV Technology

- ▶ SR-IOV stands for Single Root Input/Output Virtualization
- ▶ SR-IOV allows isolation of the PCI Express resources

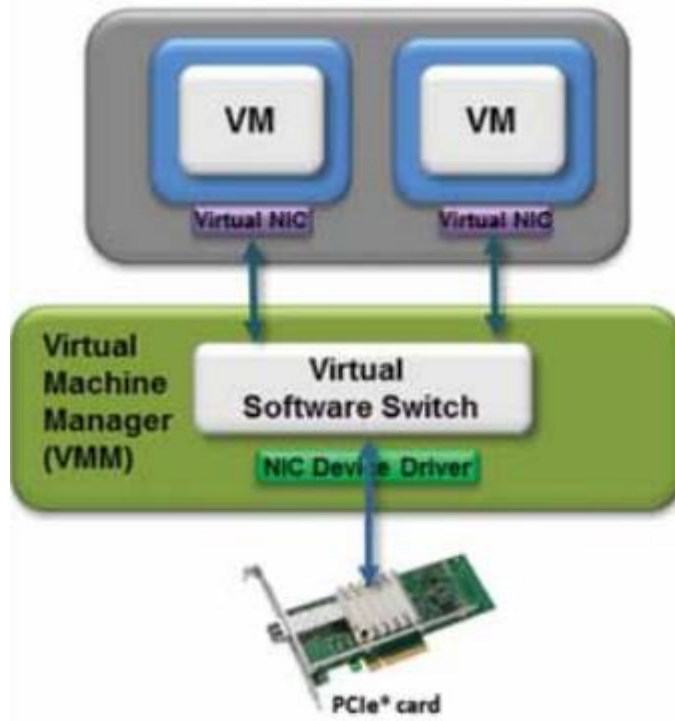
SR-IOV Technology

- ▶ SR-IOV stands for Single Root Input/Output Virtualization
- ▶ SR-IOV allows isolation of the PCI Express resources
- ▶ SR-IOV enables network traffic to bypass the software switch layer of the virtualization stack.

SR-IOV Technology

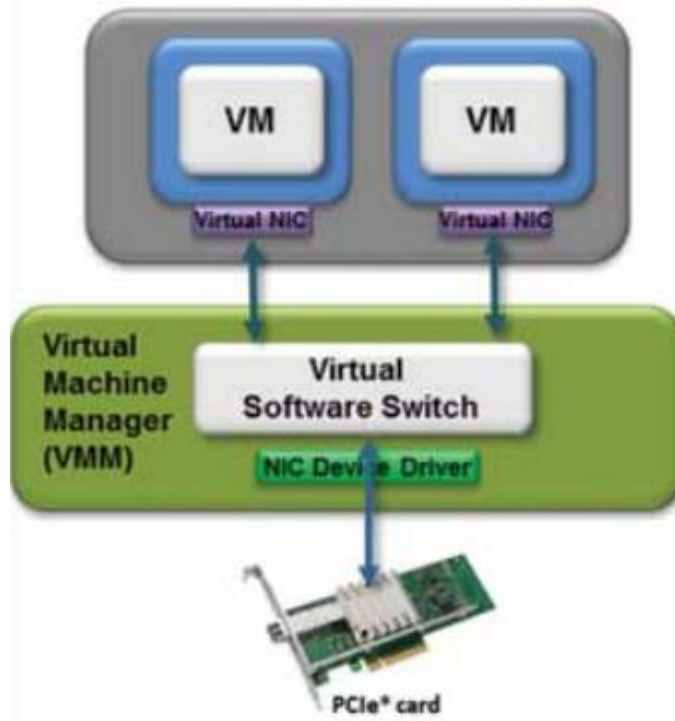
- ▶ SR-IOV stands for Single Root Input/Output Virtualization
- ▶ SR-IOV allows isolation of the PCI Express resources
- ▶ SR-IOV enables network traffic to bypass the software switch layer of the virtualization stack.
- ▶ **Physical Function (PF)** is the physical NIC that has SR-IOV capabilities. **Virtual Functions (VFs)** are created from the physical NIC

SR-IOV Technology

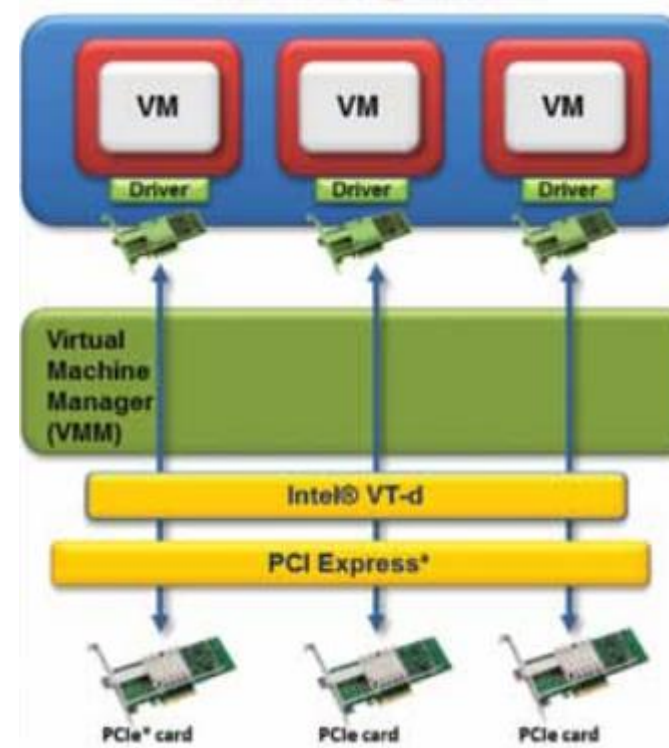


Software based Sharing

SR-IOV Technology

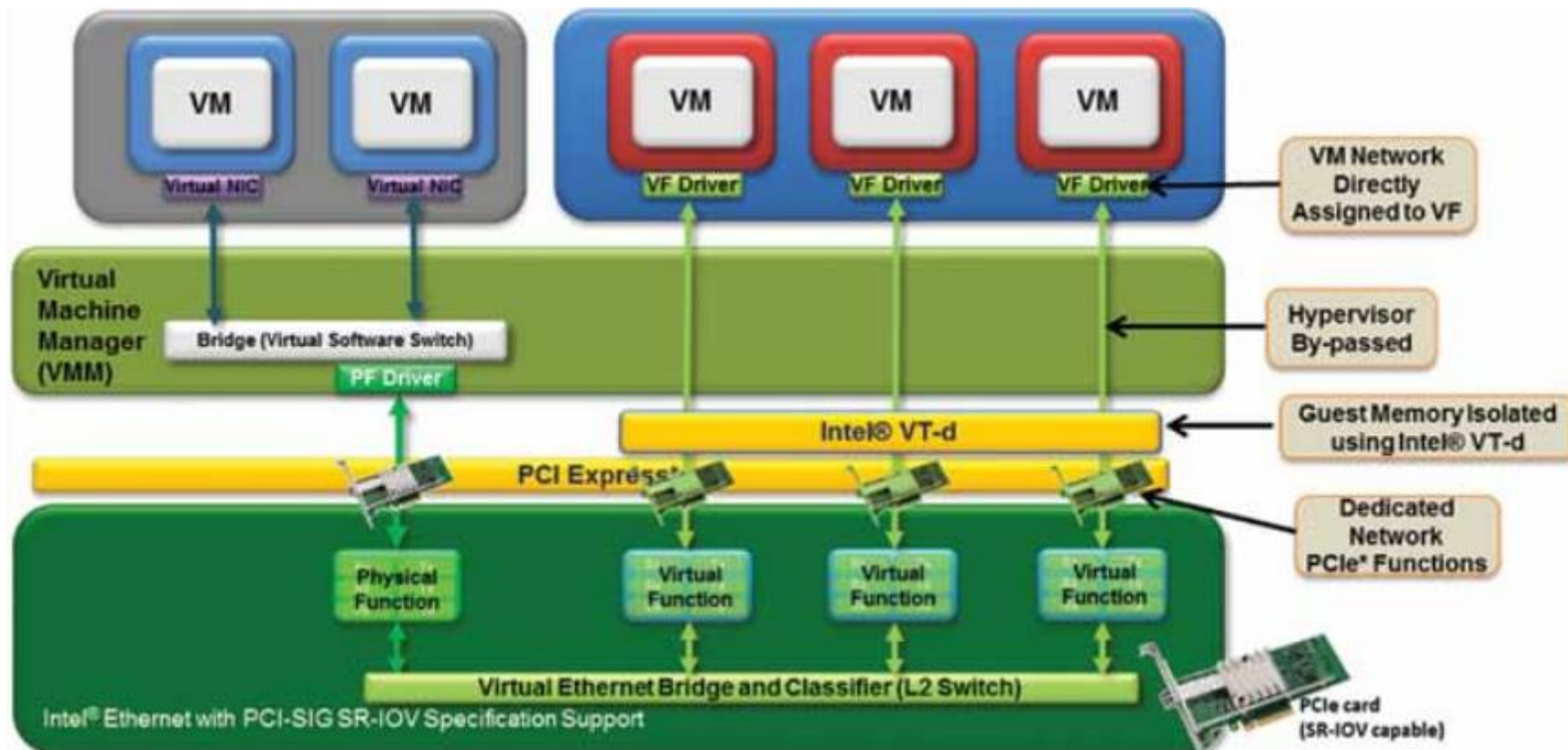


Software based Sharing



Direct Assignment

SR-IOV Technology



SR-IOV Implementation

Manual Integration

- ▶ Ensure that you have a supported SR-IOV capable device

Manual Integration

- ▶ Ensure that you have a supported SR-IOV capable device
- ▶ Ensure the PF driver allocates a number of VFs (e.g. `modprobe igb max_vfs=8`)

Manual Integration

- ▶ Ensure that you have a supported SR-IOV capable device
- ▶ Ensure the PF driver allocates a number of VFs (e.g. `modprobe igb max_vfs=8`)
- ▶ Assign the VF to a guest (see Features/KVM PCI Device Assignment)

Manual Integration

- ▶ Ensure that you have a supported SR-IOV capable device
- ▶ Ensure the PF driver allocates a number of VFs (e.g. `modprobe igb max_vfs=8`)
- ▶ Assign the VF to a guest (see Features/KVM PCI Device Assignment)
- ▶ Load the VF driver in the guest and ensure the device works as expected

SR-IOV Support in OpenStack

- ▶ Before Juno Release SR-IOV had to be manually integrated

SR-IOV Support in OpenStack

- ▶ Before Juno Release SR-IOV had to be manually integrated
- ▶ Official SR-IOV support added to OpenStack from Juno Release

SR-IOV Support in OpenStack

- ▶ Before Juno Release SR-IOV had to be manually integrated
- ▶ Official SR-IOV support added to OpenStack from Juno Release
- ▶ There are two ways that SR-IOV port may be connected:
 - ▶ Directly connected to its VF

SR-IOV Support in OpenStack

- ▶ Before Juno Release SR-IOV had to be manually integrated
- ▶ Official SR-IOV support added to OpenStack from Juno Release
- ▶ There are two ways that SR-IOV port may be connected:
 - ▶ Directly connected to its VF
 - ▶ Connected with a macvtap device that resides on the host, which is then connected to the corresponding VF

SR-IOV Support in OpenStack

- ▶ In order to enable SR-IOV, the following steps are required:
 - ▶ Create Virtual Functions (Compute)
 - ▶ Whitelist PCI devices in nova-compute (Compute)
 - ▶ Configure neutron-server (Controller)
 - ▶ Configure nova-scheduler (Controller)
 - ▶ Enable neutron sriov-agent (Compute)

SR-IOV Support in OpenStack

- ▶ In order to enable SR-IOV, the following steps are required:
 - ▶ Create Virtual Functions (Compute)
 - ▶ Whitelist PCI devices in nova-compute (Compute)
 - ▶ Configure neutron-server (Controller)
 - ▶ Configure nova-scheduler (Controller)
 - ▶ Enable neutron sriov-agent (Compute)
- ▶ There are 2 ways of configuring SR-IOV:
 - ▶ With the sriov-agent running on each compute node (default)
 - ▶ Without the sriov-agent running on each compute node (Deprecated)

Known Limitations – OpenStack SRIOV

- ▶ Security Group is not supported and the agent is only working with the enabled firewall driver

Known Limitations – OpenStack SRIOV

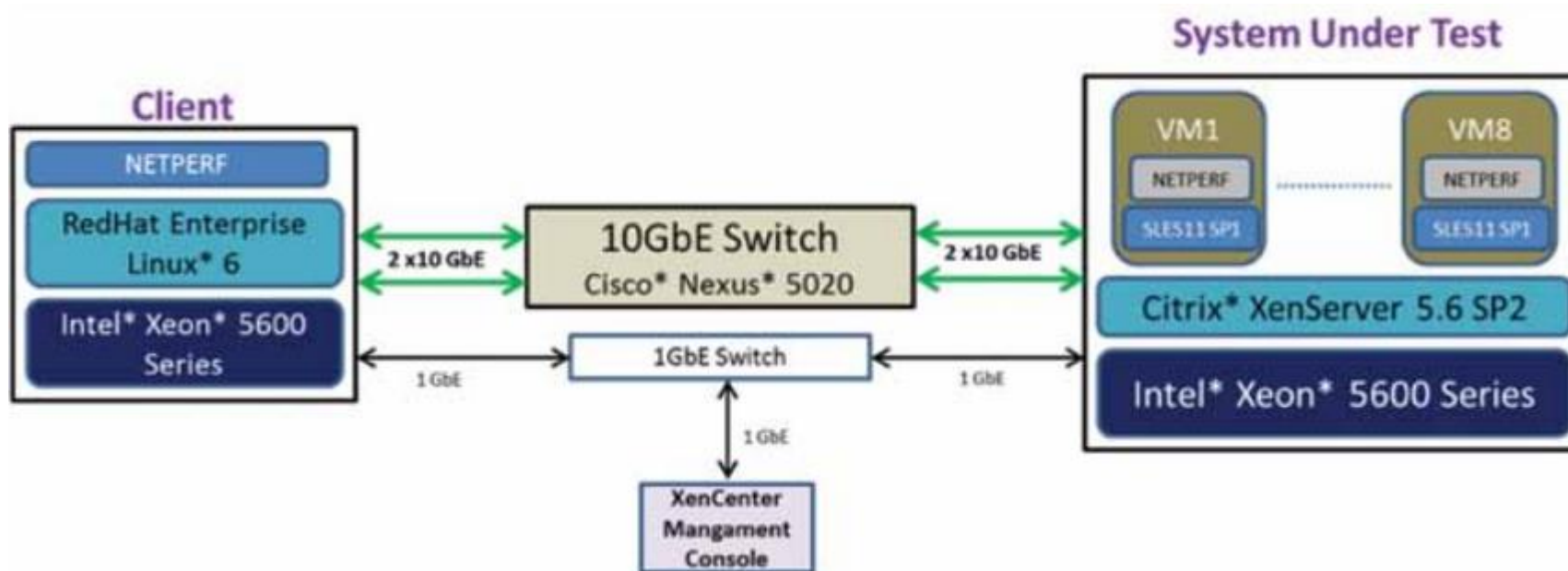
- ▶ Security Group is not supported and the agent is only working with the enabled firewall driver
- ▶ No OpenStack Dashboard integration. Users need to use CLI or API to create neutron SR-IOV ports

Known Limitations – OpenStack SRIOV

- ▶ Security Group is not supported and the agent is only working with the enabled firewall driver
- ▶ No OpenStack Dashboard integration. Users need to use CLI or API to create neutron SR-IOV ports
- ▶ Live migration is not supported for instances with SR-IOV ports

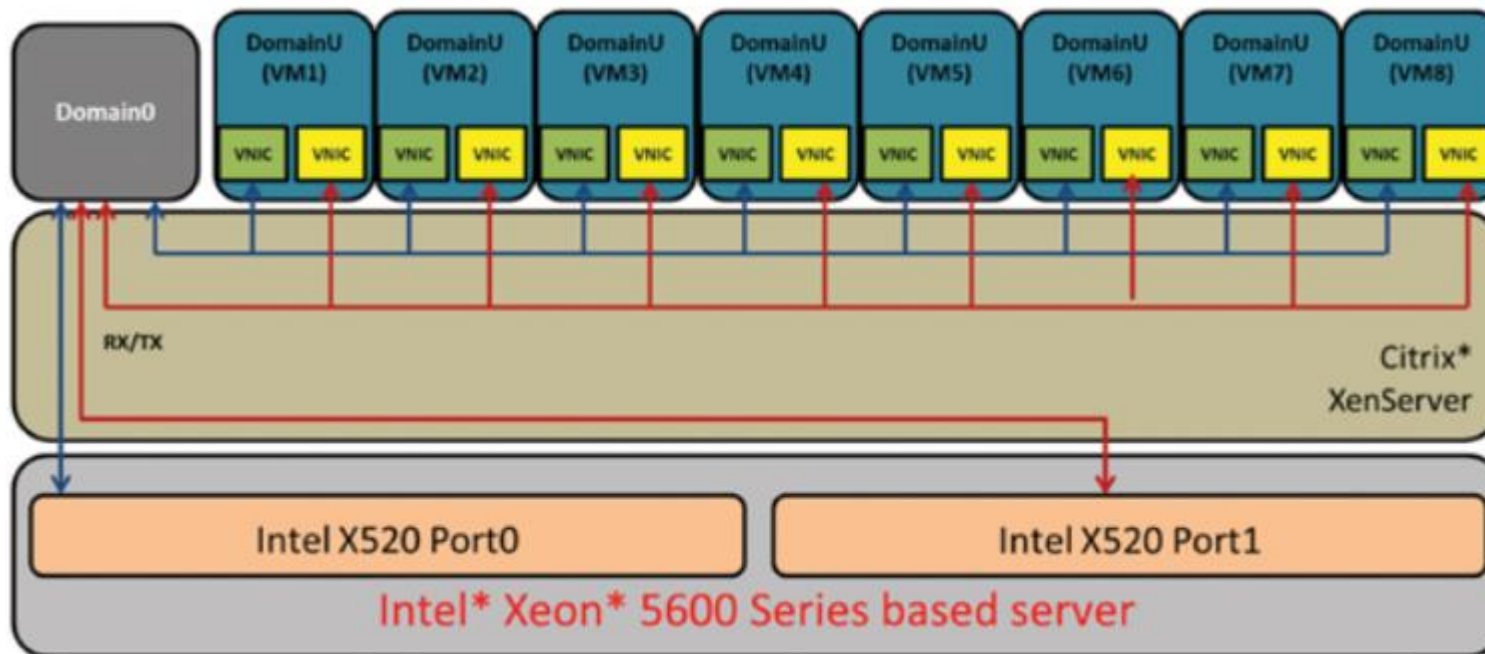
Intel Case Study

► Testbed setup



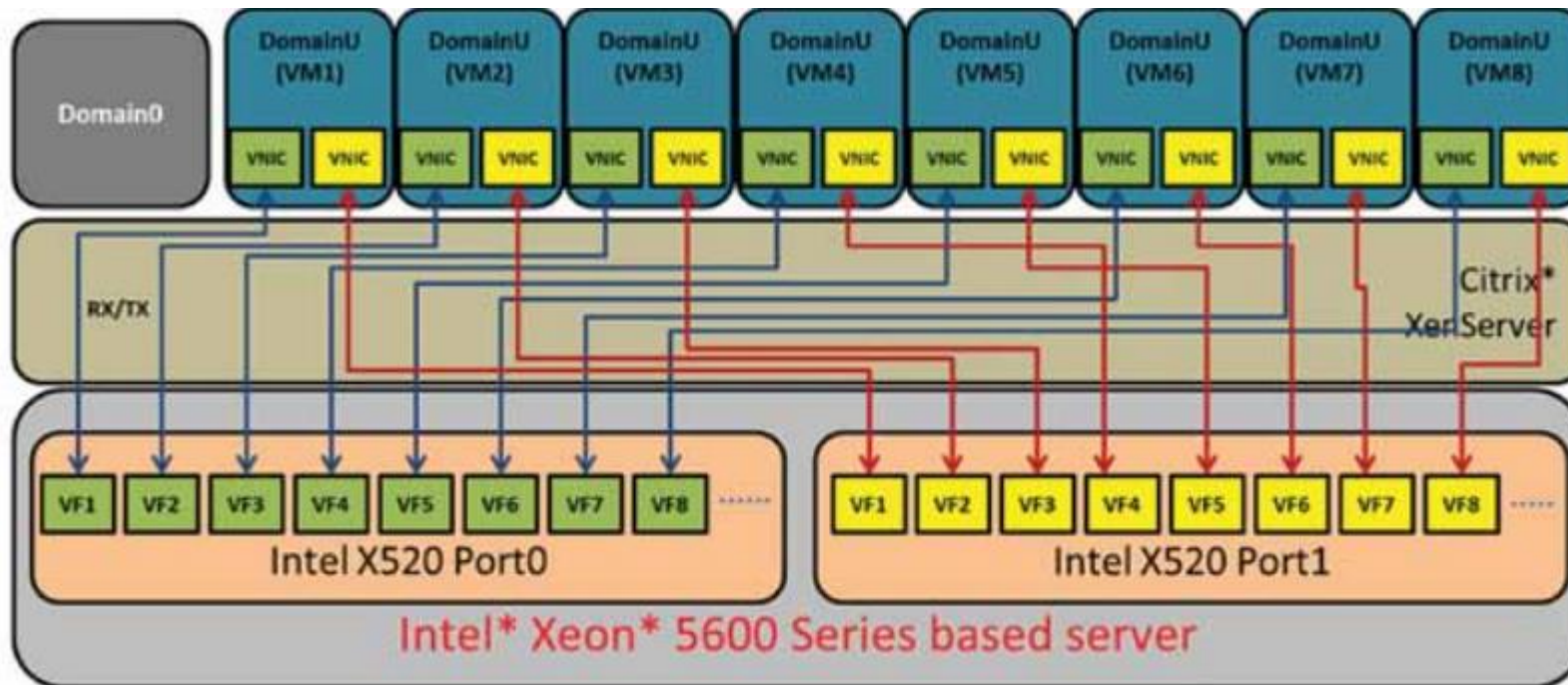
Intel Case Study

► Non-SRIOV network topology



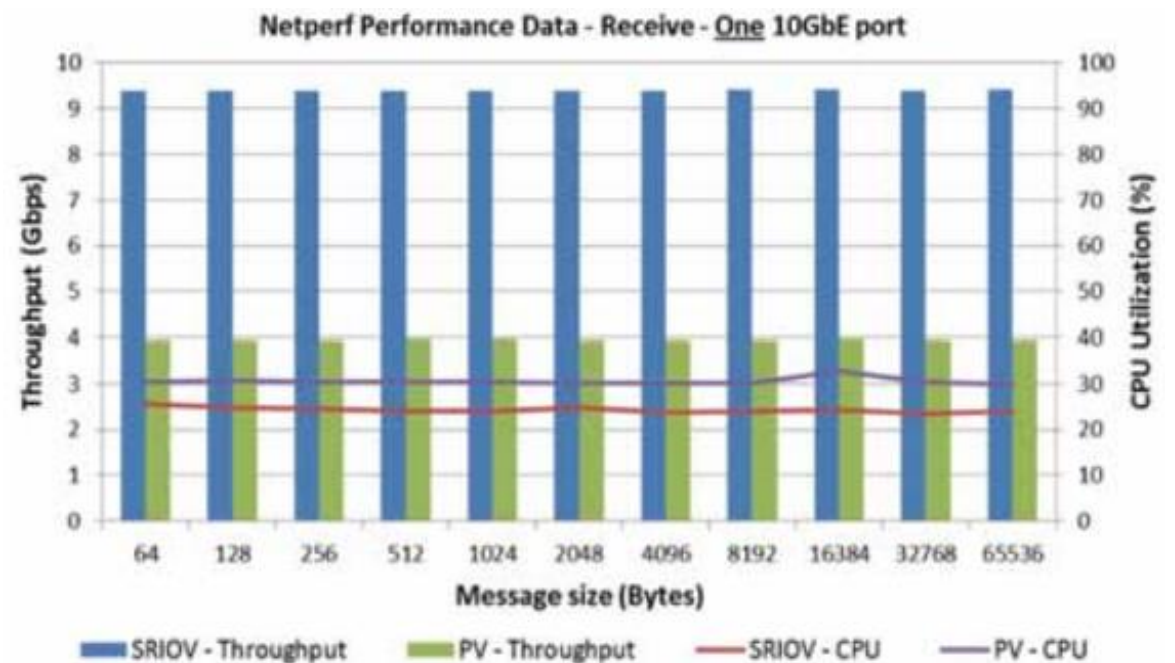
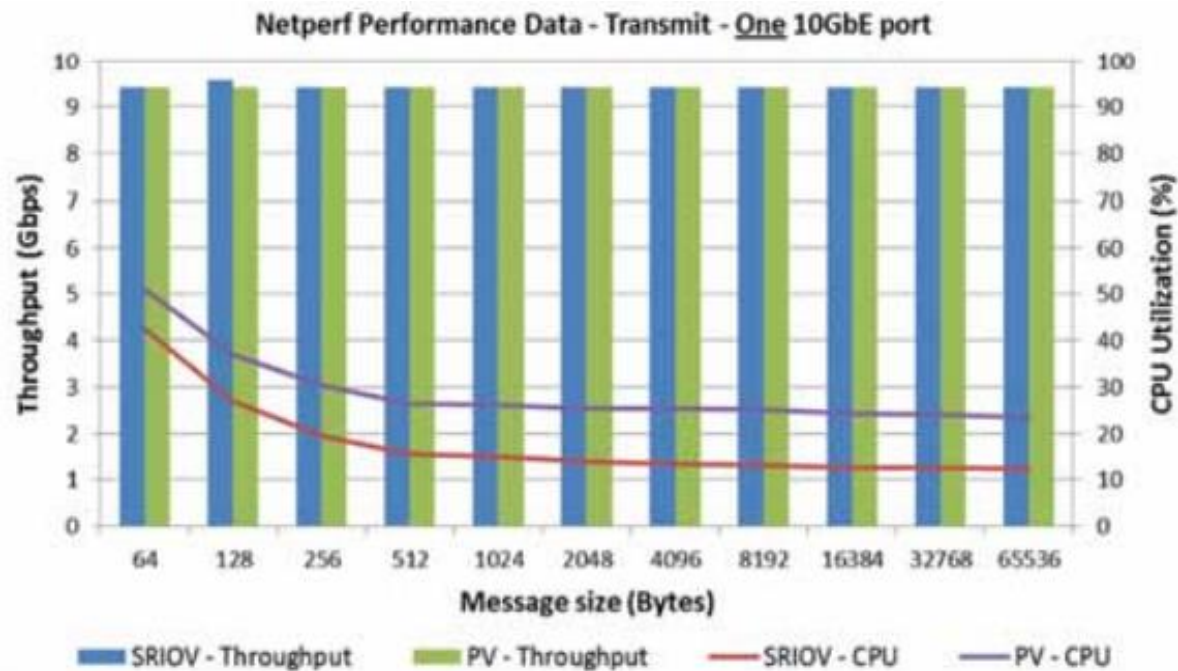
Intel Case Study

► SRIOV network topology



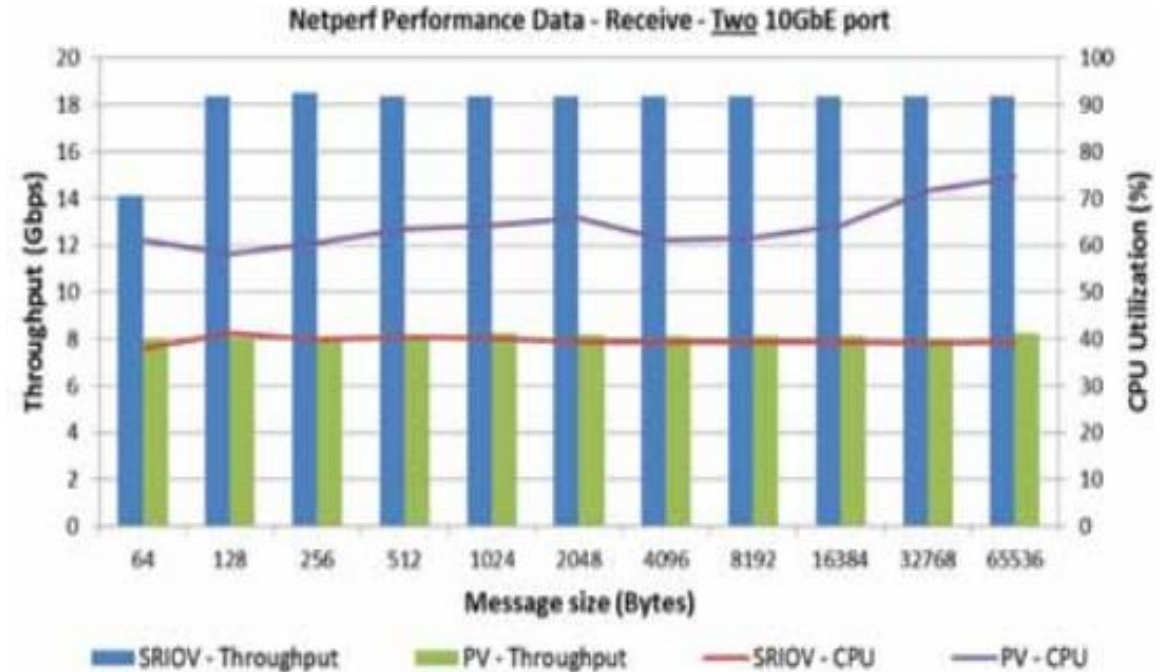
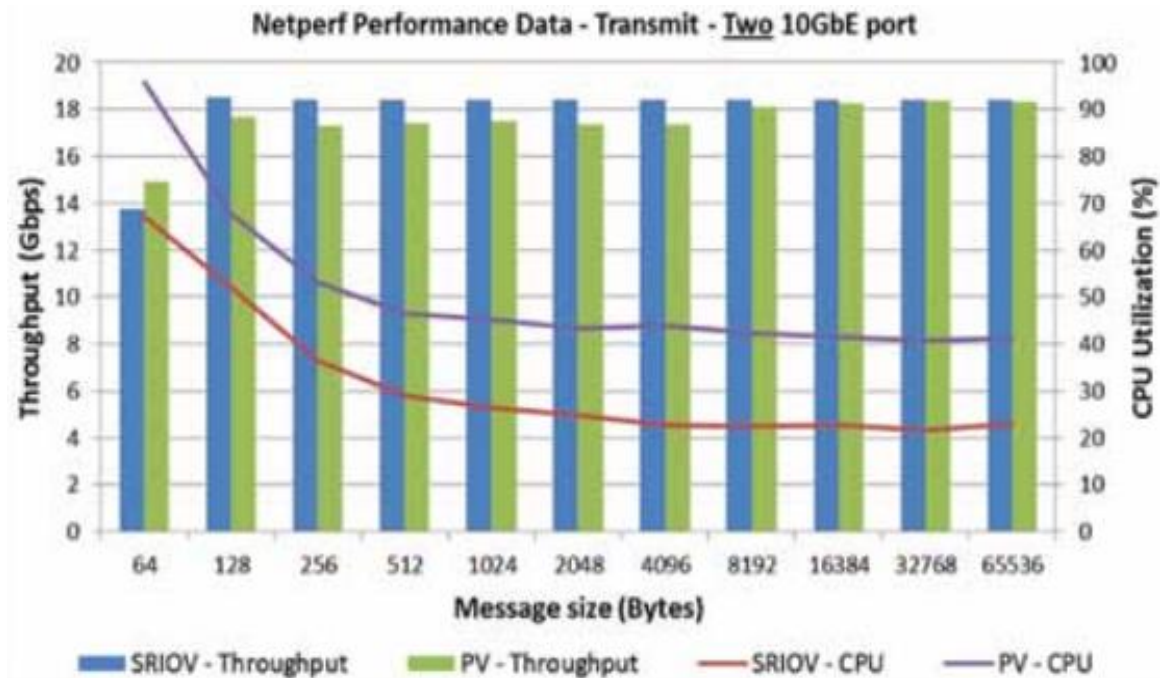
Intel Case Study

- Transmit & Receive Benchmarking (One 10GbE port)



Intel Case Study

- Transmit & Receive Benchmarking (TWO 10GbE port)



Intel Case Study

- ▶ Observations:
 - ▶ SR-IOV drivers can achieve nearly line rate or utilize more than 90 percent of available network bandwidth while using less CPU resources in comparison with para-virtualized drivers

Intel Case Study

- ▶ Observations:
 - ▶ SR-IOV drivers can achieve nearly line rate or utilize more than 90 percent of available network bandwidth while using less CPU resources in comparison with para-virtualized drivers
 - ▶ Para-virtualized drivers can only drive approximately 50 percent of the available bandwidth while using more CPU resources.

Benefits of SRIOV

- ▶ Achieves near native I/O performance by eliminating the SW overhead

Benefits of SRIOV

- ▶ Achieves near native I/O performance by eliminating the SW overhead
- ▶ Reduces the system overhead incurred by the VMM for I/O processing

Benefits of SRIOV

- ▶ Achieves near native I/O performance by eliminating the SW overhead
- ▶ Reduces the system overhead incurred by the VMM for I/O processing
- ▶ Can use VFs instead of multiple physical I/O devices to separate I/O functions

Benefits of SRIOV

- ▶ Achieves near native I/O performance by eliminating the SW overhead
- ▶ Reduces the system overhead incurred by the VMM for I/O processing
- ▶ Can use VFs instead of multiple physical I/O devices to separate I/O functions
- ▶ Reduces HW costs, simplifies cabling and simplifies management

Benefits of SRIOV

- ▶ Achieves near native I/O performance by eliminating the SW overhead
- ▶ Reduces the system overhead incurred by the VMM for I/O processing
- ▶ Can use VFs instead of multiple physical I/O devices to separate I/O functions
- ▶ Reduces HW costs, simplifies cabling and simplifies management
- ▶ SR-IOV is beneficial in workloads with high packet or low latency requirements

Customizing OpenStack

- ▶ OpenStack might not do everything you need it to do out of the box

Customizing OpenStack

- ▶ OpenStack might not do everything you need it to do out of the box
- ▶ Add new features in OpenStack

Customizing OpenStack

- ▶ OpenStack might not do everything you need it to do out of the box
- ▶ Add new features in OpenStack
- ▶ Solve real world problems faced by your company

Customizing OpenStack

- ▶ OpenStack might not do everything you need it to do out of the box
- ▶ Add new features in OpenStack
- ▶ Solve real world problems faced by your company
- ▶ Write a plugin and enables your product to work with OpenStack

Closing Remarks

- ▶ We briefly discussed OpenStack Neutron
- ▶ Analyzed the performance benefits of SR-IOV technology
- ▶ SR-IOV support in OpenStack
- ▶ What can you do to improve performance?

References

- ▶ <https://wiki.openstack.org/wiki/Neutron>
- ▶ http://www.intelcloudbuilders.com/docs/Intel_Cloud_Builders_Unified_Networking_Citrix_March_2012.pdf
- ▶ <https://fedoraproject.org/wiki/Features/SR-IOV>
- ▶ <http://docs.openstack.org/liberty/networking-guide/adv-config-sriov.html>

