Considerations for Benchmarking Network Performance in Containerized Infrastructure

draft-ietf-bmwg-containerized-infra-00

Minh-Ngoc Tran (Soongsil University), Sridhar Rao (The Linux Foundation),

Jangwon Lee, Younghan Kim (Soongsil University)

Scope

- Previous **NFV benchmarking** related RFCs
  - RFC 8172: Considerations for Benchmarking Virtual Network Functions and Their Infrastructure
  - RFC 8204: Benchmarking Virtual Switches in the Open Platform for NFV (OPNFV)

- The primary scope of this document is to fill in **the gaps** of these works

when applying to containerized NFV infrastructure.

- **The consideration gaps are:**
  - Different **network models/topologies configured by Container Network Interfaces** (including the extended Berkeley Packet Filter model which was not mentioned in previous documents)

  - **Resources configuration for containers.**

Networking Models

Different with VM-based infrastructure, network in Container-based infrastructure

Is configured by a specific Container Network Interface (CNI) Plugin

- It inserts an interface in container network namespace and connects to the host network namespace
- Network configuration methods are different between CNI Plugins.

Different CNI Plugins create Different kinds of containerized networking model

For each model, this document discusses

- **Model architecture differences**
- **Notable model examples**
- **Model repeatable environment setup guidance** (newly added based on previous meeting comment)

Networking Models

Model Architecture

- **Kernel-space non Acceleration**
  - Packet processing through the switching/routing component at the host's Kernel-space
  - Notable examples: Flannel, Calico (non-ebpf), OpenvSwitch,…
- **User-space Acceleration**
  - Packet processing bypass Kernel-space's network stack, traverse through User-space vSwitch
  - Notable examples: OVS-DPDK, VPP user-space vSwitch
- **eBPF Acceleration**
  - Packet processing bypass Kernel via eBPF programmed path
  - Notable examples: Cilium, AFXDP plugin, OVS-DPDK/VPP vswitch with AFXDP support
- **Smart-NIC Acceleration**
  - Packet processing bypass Kernel via Smart-NIC's features (Virtual function / offloaded eBPF)
  - Notable examples: SR-IOV, Cilium (offloaded eBPF mode)
- **Model Combination**
  - Packet processing bypass Kernel via combining SR-IOV and User-space vSwitch

Networking Models

Repeatable Environment Setup recommendations

- General: Use Kubernetes as container platform, Multus CNI to enable multiple network interfaces at container
- **Kernel-space non Acceleration**
  - Simply install corresponding CNI plugin yaml configuration file
- **User-space Acceleration**
  - DPDK libraries installation, binding NIC port to DPDK Poll Mode Driver
  - vSwitch vhost-user/memif ports configuration, vSwitch traffic path mapping between ports
  - Userspace CNI plugin is required to be installed and configured
- **eBPF Acceleration**
  - Install corresponding CNI (Cilium / AFXDP CNI plugin)
  - In case of vSwitch with AFXDP-ebpf support, Requires Userspace CNI and above vSwitch configurations
- **Smart-NIC Acceleration**
  - Requires DPDK libraries, SR-IOV capable NIC card and BIOS Virtual Function support feature
  - Then, created Virtual Functions should be bound to DPDK driver
  - SR-IOV CNI plugin / Cilium CNI plugin installation depends on the chosen model
- **Model Combination**
  - Follows above SR-IOV and user-space vSwitch
  - Configures North/South traffic for SR-IOV and East/West traffic for user-space vSwitch

Resources Configuration

Compared with previous RFCs, *Additional Parameters* ***are required*** *for container network*

- **CPU Isolation / NUMA Affinity**
  - Selected CPU Isolation level/technique
  - NUMA cores allocation to pod
- **Pod Hugepages**
  - Pod's Hugepage size
- **Pod CPU Cores and Memory allocation**
  - Pod's CPU cores allocation
  - Pod's RAM allocation
- **Service Function Chaning**
  - Number of CNFs/pod
  - Selected CNI plugin

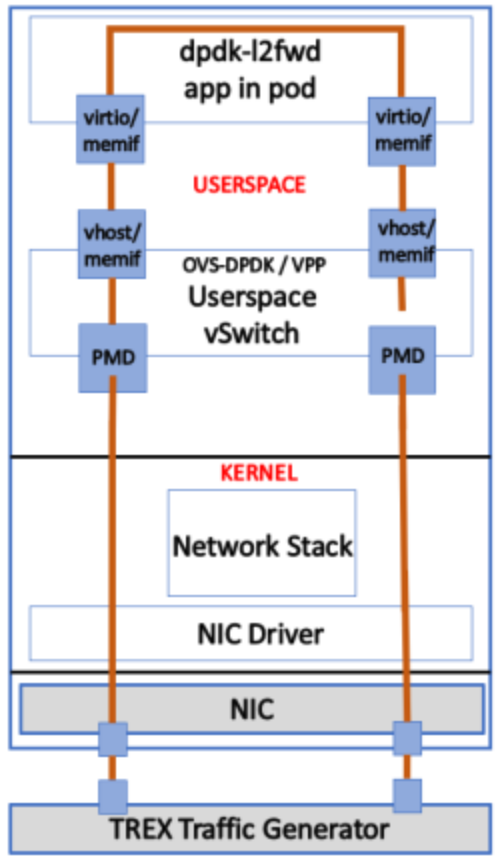
Summary

- From adoption comments, we added brief key environment setup guidance for each container networking models
- Fixed some grammartical issues.

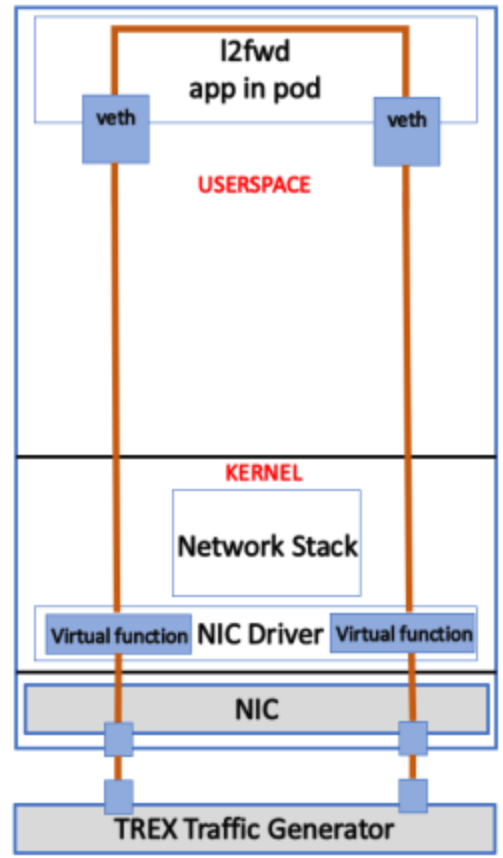
- Additional reviews and comments are welcomed to polish the document after adoption.

Backup Slides

- Different networking models based on packet acceleration techniques

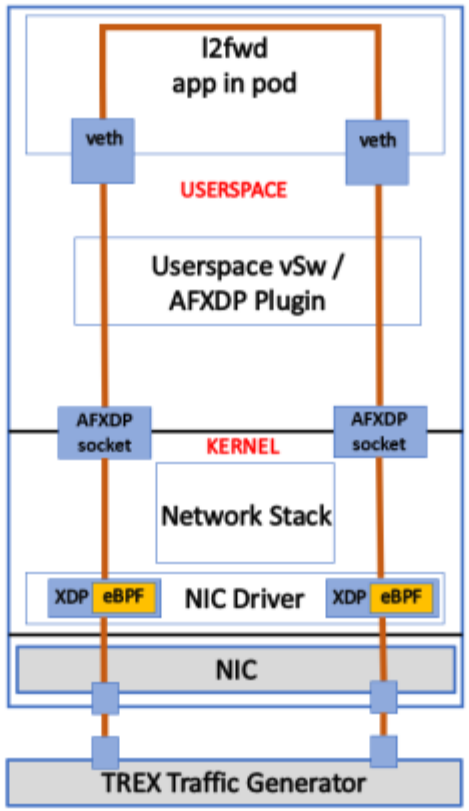


Userspace Acceleration
(OVS-DPDK, VPP vSwitch)

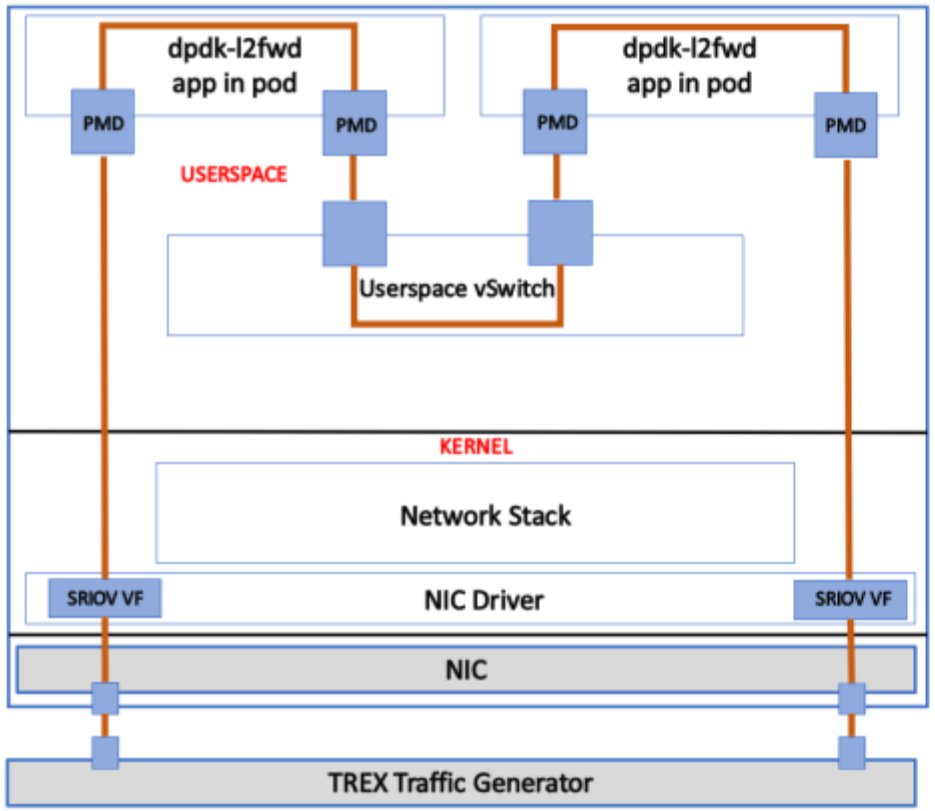


SRIOV Acceleration

- Different networking models based on packet acceleration techniques



eBPF Acceleration
(using AFXDP)



Combined Model Acceleration
(SR-IOV + OVS-DPDK/VPP)