Virtio-net failover support

Jens Freimann
Senior Software Engineer
Red Hat
Agenda

● Why
● Problem
● Existing solutions
● Approach for DPDK / Open problems
● Roadmap
When there is a fast NIC available to the VM I want to use it

But I also want the flexibility of a PV device
I want fast *and* flexible

I want to be able to migrate my VMs
Need to migrate -> virtio-net?
But ... it will be slower!

So...
Let’s combine them?
Migration

1. Unplug VF
2. Switch over to virtio-net device
3. Migrate
4. On target: re-plug VFIO device if available
Migration

Start migration

DPDK app

Virtio-net

VF

Hypervisor

NIC

DPDK app

Virtio-net

VF

Hypervisor

NIC
1. Unplug VF

- QEMU request PCIe unplug VF
- Wait for completion, roll-back if no answer
Migration

1. Unplug VF

2. Switch over to virtio-net device
   ● In receive and xmit function always decide if VF available or not
Migration

1. Unplug VF
2. Switch over to virtio-net device
3. Migrate
   • New migration state wait-unplug
   • New events for libvirt to consume
Migration

1. Unplug VF
2. Switch over to virtio-net device
3. Migrate
4. On target: re-plug VFIO device
Migration: possible complications

● Roll-back
● Retain VF resources in QEMU until successfully unplugged from guest
● Guest doesn’t respond to PCI unplug request?
● Need to re-plug VF to source VM
There must be a solution for that, right?
Exists: Bonding

- Bond a VF and a PV device
- Active-backup
- Exists in DPDK and Linux
- Proven to work

Problem: requires manual configuration on guest side
We want to avoid that!
Exists: vdev_netvsc and failsafe
Exists: support in Linux/KVM stack

VIRTIO_NET_F_STANDBY

net_failover kernel module

Failover, primary, standby device

MATCH devices with same MAC
Exists: SR-IOV support in netvsc (Stephen Hemminger)

Transparently manage the VF device from PV driver

2-device model (in linux)
Ideas for virtio net_failover in DPDK

- Look at netvsc
- Combine with VIRTIO_NET_F_STANDBY support
virtio-net driver in DPDK

QEMU started with:
- device vfio-pci,...,net_failover_standby_id="standby0"
- device virtio-net-pci,...,failover=on

Configure (rte_eth-dev_configure) and start rte_eth_dev for VF device from virtio-net code
virtio-net driver in DPDK

Look for device with same MAC, save as port id of vf
Use this port id to set device owner (rte_eth_dev_owner_set)
In NetVSC: driver receives vf association message.
-> How to do this in virtio-net?

RX/TX: if VF attached use it to receive/send

Register LSC event callbacks
virtio-net driver in DPDK

At migration: QEMU triggers unplug of PCI VF device

Virtio-net driver needs to receive notification of pci unplug OR...
virtio-net driver in DPDK

... send a message before we start migration
- before unplug
- how? Via control virtqueue? We would have to make it bi-directional first
- via the device event notification framework in DPDK? Can we use this in PMDs? Is it meant for apps only? What if both app and pmd register for events?
Roadmap

First version of patches within 4-6 weeks

Target: include in DPDK 20.02 release
Thank you!

Questions?

jfreimann@redhat.com

linkedin.com/company/red-hat

youtube.com/user/RedHatVideos

facebook.com/redhatinc

twitter.com/RedHat