Ethernet Interface

Table of contents

List of Figures

Counters

Counters are used to provide information about how well an operating system, an application, a service, or a driver is performing. The counter data help determine system bottlenecks and fine-tune the system and application performance. The operating system, network, and devices provide counter data that an application can consume to provide users with a graphical view of how well the system is performing.
The counter index is a Queue Pair (QP) attribute given in the QP context. Multiple QPs may be associated with the same counter set. If multiple QPs share the same counter, the counter value will represent the cumulative total.

RoCE Counters

- RoCE counters are available only through sysfs located under:

    - # /sys/class/infiniband/<device>/ports/*/hw_counters/

    - # /sys/class/infiniband/<device>/hw_counters/

    - # /sys/class/infiniband/<device>/ports/*/counters/

For mlx5 port and RoCE counters, refer to the Understanding mlx5 Linux Counters Community post.

SR-IOV Counters

Physical Function can also read Virtual Functions' port counters through sysfs located under # /sys/class/net/<interface_name>/device/sriov/<index>/stats/

ethtool Counters

The ethtool counters are counted in different places, according to which they are divided into groups. Each counters group may also have different counter types.
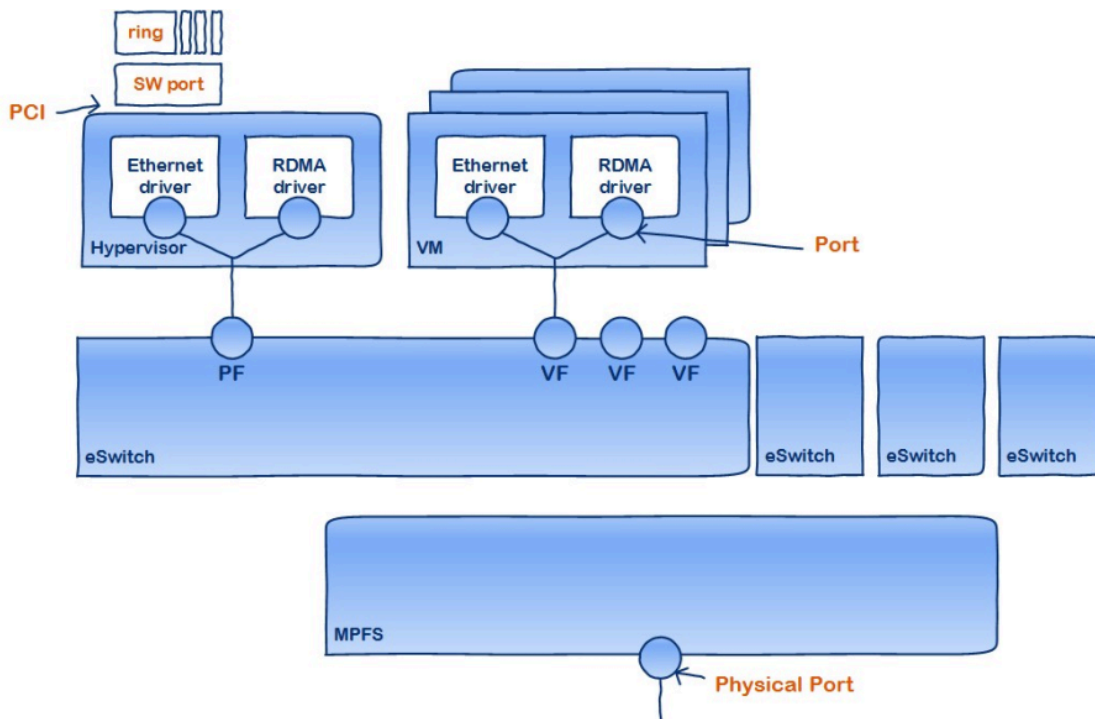
For the full list of supported ethtool counters, refer to the <u>Understanding mlx5 ethtool Counters</u> community post.

Persistent Naming

To avoid network interface renaming after boot or driver restart, set the desired constant interface name in the "/etc/udev/rules.d/70-persistent-net.rules" file.

- Example for Ethernet interfaces:

```
PCI device 15b3:1019 (mlx5_core)
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",
ATTR{address}=="00:02:c9:fa:c3:50", ATTR{dev_id}=="0x0", ATTR{type}=="1",
KERNEL=="eth*", NAME="eth1"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",
ATTR{address}=="00:02:c9:fa:c3:51", ATTR{dev_id}=="0x0", ATTR{type}=="1",
KERNEL=="eth*", NAME="eth2"
```

```
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",
ATTR{address}=="00:02:c9:e9:56:a1", ATTR{dev_id}=="0x0", ATTR{type}=="1",
KERNEL=="eth*", NAME="eth3"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",
ATTR{address}=="00:02:c9:e9:56:a2", ATTR{dev_id}=="0x0", ATTR{type}=="1",
KERNEL=="eth*", NAME="eth4"
```

- Example for IPoIB interfaces:

```
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{dev_id}=="0x0",
ATTR{type}=="32", NAME="ib0"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{dev_id}=="0x1",
ATTR{type}=="32", NAME="ib1"
```

Interrupt Request (IRQ) Naming

Once IRQs are allocated by the driver, they are named `mlx5_comp<x>@pci:<pci_addr>`. The IRQ name is constant and is not affected by the interface state.
The mlx5_core driver allocates all IRQs during loading time to support the maximum possible number of channels. Once the driver is up, no further IRQs are freed or allocated. Changing the number of working channels does not re-allocate or free the IRQs.