root@1745d66150af:/# ifconfig

eth0 Link encap:Ethernet HWaddr 02:42:ac:11:00:02

inet addr:172.17.0.2 Bcast:0.0.0.0 Mask:255.255.0.0

inet6 addr: fe80::42:acff:fe11:2/64 Scope:Link

UP BROADCAST RUNNING MTU:1500 Metric:1

RX packets:6 errors:0 dropped:0 overruns:0 frame:0

TX packets:6 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:1000

RX bytes:468 (468.0 B) TX bytes:468 (468.0 B)

lo Link encap:Local Loopback

inet addr:127.0.0.1 Mask:255.0.0.0

inet6 addr: ::1/128 Scope:Host

UP LOOPBACK RUNNING MTU:16436 Metric:1

RX packets:0 errors:0 dropped:0 overruns:0 frame:0

TX packets:0 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:0

RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)

root@1745d66150af:/#

root@caef75a7ce4d:/# ifconfig

eth0 Link encap:Ethernet HWaddr 02:42:ac:11:00:04

inet addr:172.17.0.4 Bcast:0.0.0.0 Mask:255.255.0.0

inet6 addr: fe80::42:acff:fe11:4/64 Scope:Link

UP BROADCAST RUNNING MTU:1500 Metric:1

RX packets:3 errors:0 dropped:0 overruns:0 frame:0

TX packets:3 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:1000

RX bytes:238 (238.0 B) TX bytes:238 (238.0 B)

lo Link encap:Local Loopback

inet addr:127.0.0.1 Mask:255.0.0.0

inet6 addr: ::1/128 Scope:Host

UP LOOPBACK RUNNING MTU:16436 Metric:1

RX packets:0 errors:0 dropped:0 overruns:0 frame:0

TX packets:0 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:0

RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)

root@caef75a7ce4d:/#

Dear Salman,

Talking about the two experiments you mentioned yesterday:

run qperf tcp\_bw/lat <host\_ib0\_IPADDR> on host-host

configure container’s ib0 and run

qperf tcp\_bw /lat <container’s\_ib0\_IPADDR> on container-container.

for the first one I will do 10 tests and the result should be very close to this one:  
root@zhuangdizhu2 ~]# qperf 172.16.1.2 tcp\_bw

tcp\_bw:

bw = 976 MB/sec

for the second one, it will not work because we cannot configure ib0 on a container.

Containers are different from VMs. when we create a VM on a KVM, there will be an ethernet bridge (virbr0)which sends TCP packets from the VM to the real host’s eth0. This is similar when I start a container (there will be a bridge docker0).

The ib0 on the host receives TCP packet, and then copies it to the infiniband device( TCP packet => ib0 =>infiniband packet=>infiniband device) . The ib device will not know it is TCP packet. In general, ib0 is like a bridge, a band-aid between TCP and RDMA. it is provided by Linux kernel, which is in OS.

For each VM, it has an OS, so each VM can be configured with an ib0, then we use SR-IOV which virtualizes one infiniband card to multiple ones so each VM can access one. This is not true for containers which are built without their own OS. Actually they are built on the same OS (the real host’s OS), so if you run “uname -a” in a container you will find it’s the same kernel version as that of the host. So anyway, there will be only one ib0.

ib0 needs to be shared but we cannot do it now because there is no such thing like ”infiniband bridge”. We cannot attach inifinband device directly to the ethernet bridge(like docker0 for containers and virbr0 for VMs).

There is a way to run a qperf tcp\_bw test over ib device on containers but it has nothing to do with configuring ib0 on a container:

docker run --net host -i -t --name <container\_name> -p 19765:19765 <image>:<tag> /bin/bash

When we start a container using the above command it tells a container to use the host’s network namespace instead of using the container’s. But by doing so the network namespace will be destroyed and there will be no difference between running qperf on containers and running two qperf programs on hosts without containers.

The reason for using containers is more about “security” or ”isolation” than “virtualization”. Containers by default use their own namespaces(User namespace, memory namespace, CPU namespace, network namespace, etc). They cannot see each other even though the host can see them.

Besides, there is one way by which two containers on the can connect to each other[1]. And I will try to figure it out today.

Ok, that’s what I have learned from Michael, and I hope it will be useful for you. I write this e-mail as a study journal for my self as well ^\_^.

# Reference:

# 1. eIPoIB Manual Configuration. URL: <https://community.mellanox.com/docs/DOC-1316>

