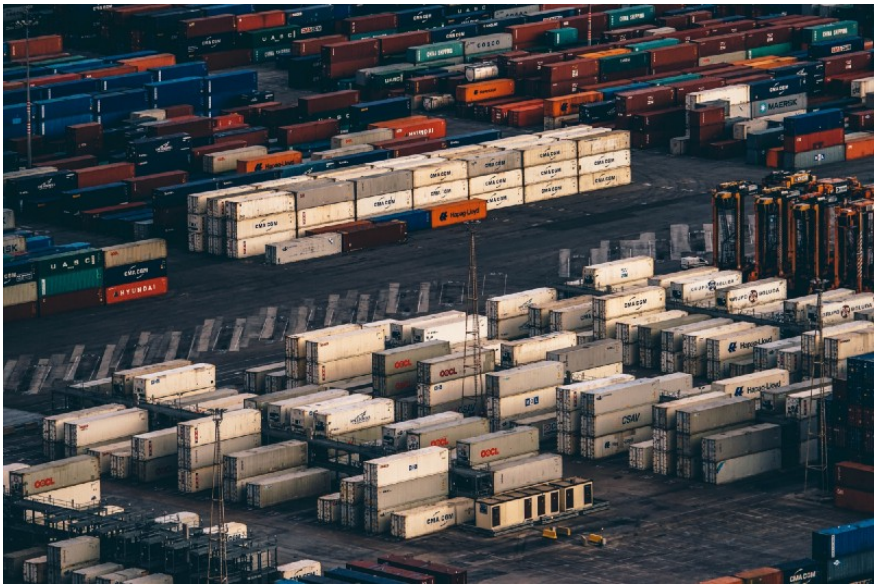
Michael Nikitochkin Follow
Mar 28, 2017 · 2 min read



Docker network performance

I think everyone has thought about what the difference between **BRIDGE** and **HOST** modes to run containers, except run applications with the same port. To test network performance we need 2 instances:

- 192.168.89.3—server instance where we would run docker containers
- 192.168.89.4—client instance

I chose http://software.es.net/iperf/ to measure the network bandwidth. Very simple and have enough features to check basic metrics. On the server instance, we require the docker. I tested against Docker 1.12.6 version.

Test Original Network Throughput

First, we need to get the original stats without docker containers. Running on the server instance:

```
[root@192.168.89.3 ~]# iperf3 -s -p 5202
```

and on the client machine:

```
[root@192.168.89.4 ~]# iperf3 -c 192.168.89.3 -p 5202
```

Both server and client would return useful information. For now, we need only the result values:

```
Connecting to host 192.168.89.3, port 5202
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval       Transfer   Bandwidth     Retr
[  4] 0.00-10.00 sec 884 MBytes 742 Mbits/sec 75  sender
[  4] 0.00-10.00 sec 882 MBytes 740 Mbits/sec     receiver
```

I use `c4.large` instances and AWS limit the network to 500 Mbit/sec, here we have a bit more: **740Mbit/sec**.

Test Docker container with Network mode

To run `iperf3` in the docker is quiet simple. There are a lot of images available in the hub.docker.com.

```
[root@192.168.89.3 ~]# docker run --net=host  -it --rm --
name=iperf3-server networkstatic/iperf3 -s -p 5203
```

From client side there are no changes, only I changed the default ports depends on how we run the server `iperf3:`

```
[root@192.168.89.4 ~]# iperf3 -c 192.168.89.3 -p 5203      #
Run client
Connecting to host 192.168.89.3, port 5202
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval       Transfer   Bandwidth    Retr
[  4] 0.00-10.00 sec 884 MBytes 741 Mbits/sec 63  sender
[  4] 0.00-10.00 sec 881 MBytes 739 Mbits/sec     receiver
```

The results are pretty the same: **740 Mbit/sec**.

Test Docker container with Bridge mode

This time I used next port. So you can run in same time all servers and do tests.

```
[root@192.168.89.3 ~]# docker run  -it --rm -p 5204:5204 --
name=iperf3-server networkstatic/iperf3 -s -p 5204
```

From client side there are no changes:

```
[root@192.168.89.4 ~]# iperf3 -c 192.168.89.3 -p 5204
Connecting to host 192.168.89.3, port 5202
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval       Transfer   Bandwidth     Retr
[  4] 0.00-10.00 sec 692 MBytes 580 Mbits/sec 405     sender
[  4] 0.00-10.00 sec 691 MBytes 580 Mbits/sec
receiver
```

OK. We have only **580 Mbit/sec**. It is 80% of the maximum allowed. When we did test few years ago, for older versions of docker we had 50%.

Summary

There is no risk of reaching or exceeding the maximum throughput of your network by running an application inside the docker. Result table:

```
| Max Possible | Host mode    | Bridge mode  |
| 740 Mbit/sec | 740 Mbit/sec | 580 Mbit/sec |
| 100 %        | 100 %        | 80 %         |
```

That’s all folks!