

SDN in Docker - Project Report

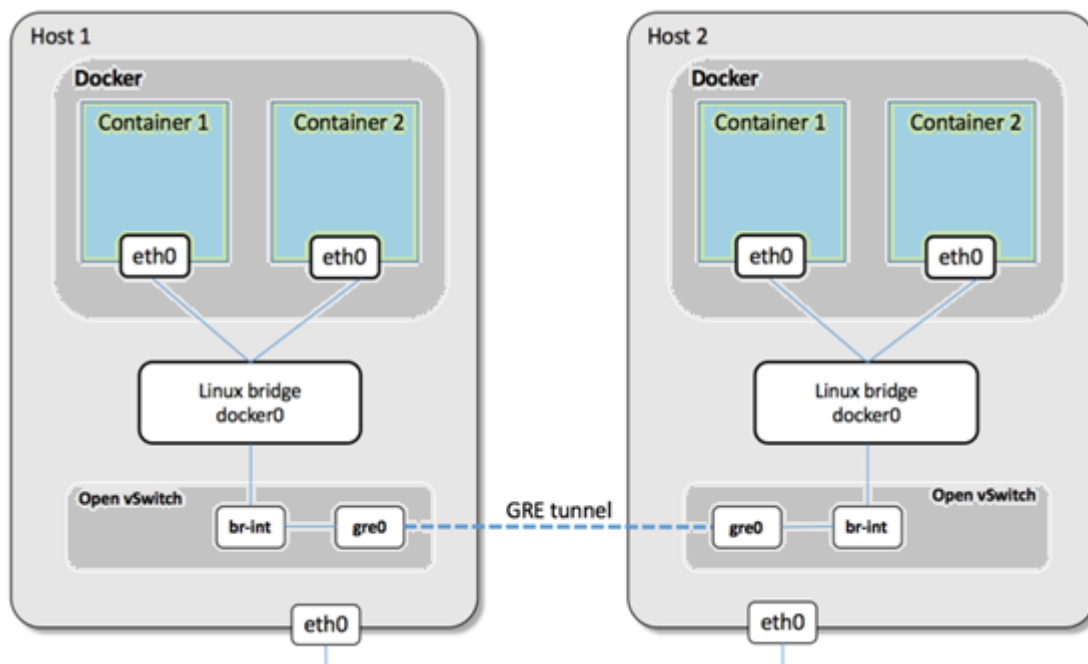
Team Members:

Vivek Nagalapura Ravindra

Amith Kumar Matapady

Problem Statement:

To set-up a connection between two host each with docker container(s) using Open vSwitch and install OpenDaylight and its features.



Introduction:

Docker:

Docker is a set of platform as a service (PaaS) products that uses OS-level virtualization to deliver software in packages called containers. Containers are isolated from one another and bundle their own software, libraries and configuration files; they can communicate with each other through well-defined channels.

Open vSwitch:

Open vSwitch is an open-source implementation of a distributed virtual multilayer switch. br-int is the default name for the core bridge used on compute and network nodes.

Generic Routing Encapsulation:

Generic Routing Encapsulation (GRE) is a communication protocol used to establish a direct, point-to-point connection between network nodes.

OpenDaylight:

OpenDaylight (ODL) is a modular open platform for customizing and automating networks of any size and scale. In this project, we are going to launch OpenDaylight in Docker. We are using the docker version of OpenDaylight Carbon built on 26 May 2017.

Virtual Ethernet Device:

The virtual Ethernet devices (or veth devices) is a local ethernet tunnel. They can act as tunnels between network namespaces to create a bridge to a physical network device in another namespace, but can also be used as standalone network devices.

Linux Bridge:

Linux bridge is a layer 2 virtual device that on its own cannot receive or transmit anything unless you bind one or more real devices to it.

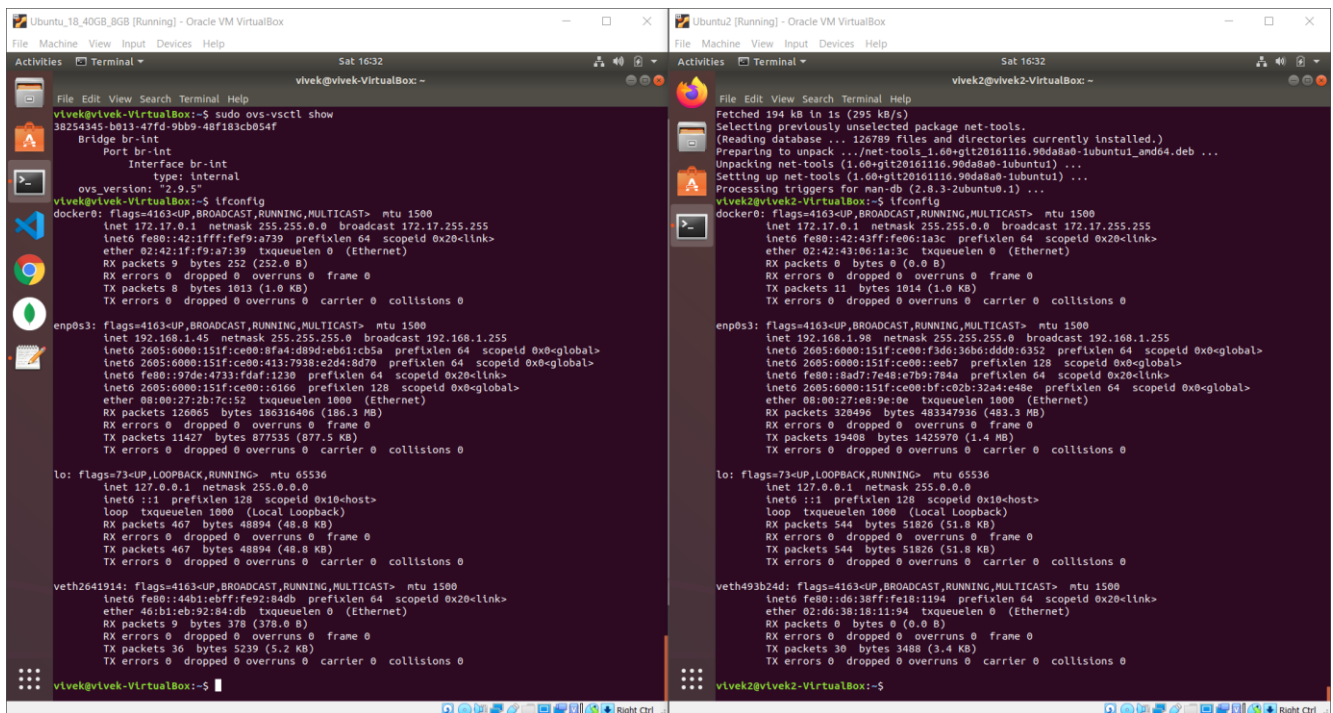
Wireshark:

Wireshark is a free and open-source packet analyzer. It is used for network troubleshooting, analysis, software and communications protocol development, and education.

Implementation:

Installations:

We have instances of Ubuntu running on two Virtual Machines. These act as our host machines.



The image shows two terminal windows side-by-side, both titled 'Ubuntu_18_40GB_8GB [Running] - Oracle VM VirtualBox'. The left window is the 'vivek@vivek-VirtualBox' terminal, and the right window is the 'vivek2@vivek2-VirtualBox' terminal. Both terminals show the output of the 'ifconfig' command for various network interfaces. The left terminal shows interfaces 'veth2641914' and 'veth2641914'. The right terminal shows interfaces 'veth493b24d' and 'veth493b24d'. The output for each interface includes details like IP address, netmask, broadcast, and statistics for RX and TX packets, bytes, errors, and collisions.

```
vivek@vivek-VirtualBox:~$ ifconfig
veth2641914: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255
    ether 02:42:1f:9a:73:99 txqueuelen 0 (Ethernet)
    RX packets 9 bytes 252 (252.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

veth2641914: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.45 netmask 255.255.255.0 broadcast 192.168.1.255
    ether 08:00:27:2b:7c:52 txqueuelen 1000 (Ethernet)
    RX packets 120005 bytes 186316406 (186.3 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 11427 bytes 877535 (877.5 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

veth2641914: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 127.0.0.1 netmask 255.0.0.0 broadcast 127.0.0.1
    ether ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 407 bytes 48894 (48.8 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 467 bytes 48894 (48.8 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

veth2641914: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255
    ether 02:42:1f:9a:73:99 txqueuelen 0 (Ethernet)
    RX packets 9 bytes 252 (252.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

vivek2@vivek2-VirtualBox:~$ ifconfig
veth493b24d: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255
    ether 02:42:1f:9a:73:99 txqueuelen 0 (Ethernet)
    RX packets 9 bytes 252 (252.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

veth493b24d: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.98 netmask 255.255.255.0 broadcast 192.168.1.255
    ether 08:00:27:2b:7c:52 txqueuelen 1000 (Ethernet)
    RX packets 320490 bytes 483347930 (483.3 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 19408 bytes 1425970 (1.4 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

veth493b24d: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 127.0.0.1 netmask 255.0.0.0 broadcast 127.0.0.1
    ether ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 544 bytes 51826 (51.8 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 544 bytes 51826 (51.8 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

veth493b24d: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255
    ether 02:42:1f:9a:73:99 txqueuelen 0 (Ethernet)
    RX packets 9 bytes 252 (252.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

The below software were installed for this project. The Open vSwitch and Docker need to be installed on both the host machines. Rest of them need to be installed on at least one of them (all of these need to be available on a single system).

1. Open VSwitch

```
vivek@vivek-VirtualBox: ~  
File Edit View Search Terminal Help  
vivek@vivek-VirtualBox:~$ sudo apt-get install -y openvswitch-switch openvswitch-common  
[sudo] password for vivek:  
Sorry, try again.  
[sudo] password for vivek:  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
The following packages were automatically installed and are no longer required:  
  efibootmgr libfwupd  
Use 'sudo apt autoremove' to remove them.  
Suggested packages:  
  ethtool openvswitch-doc  
The following NEW packages will be installed:  
  openvswitch-common openvswitch-switch  
0 upgraded, 2 newly installed, 0 to remove and 6 not upgraded.  
Need to get 2,322 kB of archives.  
After this operation, 10.3 MB of additional disk space will be used.  
Get:1 http://us.archive.ubuntu.com/ubuntu bionic-updates/main amd64 openvswitch-common amd64 2.9.5-0ubuntu0.18.04.1 [815 kB]  
Get:2 http://us.archive.ubuntu.com/ubuntu bionic-updates/main amd64 openvswitch-switch amd64 2.9.5-0ubuntu0.18.04.1 [1,507 kB]  
Fetched 2,322 kB in 1s (2,646 kB/s)  
Selecting previously unselected package openvswitch-common.  
(Reading database ... 164931 files and directories currently installed.)  
Preparing to unpack .../openvswitch-common_2.9.5-0ubuntu0.18.04.1_amd64.deb ...  
Unpacking openvswitch-common (2.9.5-0ubuntu0.18.04.1) ...  
Selecting previously unselected package openvswitch-switch.  
Preparing to unpack .../openvswitch-switch_2.9.5-0ubuntu0.18.04.1_amd64.deb ...  
Unpacking openvswitch-switch (2.9.5-0ubuntu0.18.04.1) ...  
Setting up openvswitch-common (2.9.5-0ubuntu0.18.04.1) ...  
Setting up openvswitch-switch (2.9.5-0ubuntu0.18.04.1) ...  
update-alternatives: using /usr/lib/openvswitch-switch/ovs-vswitchd to provide /usr/sbin/ovs-vswitchd (ovs-vswitchd) in auto mode  
Created symlink /etc/systemd/system/multi-user.target.wants/openvswitch-switch.service → /lib/systemd/system/openvswitch-switch.service.  
ovs-vswitchd.service is a disabled or a static unit, not starting it.  
ovsdb-server.service is a disabled or a static unit, not starting it.  
Processing triggers for ureadahead (0.100.0-21) ...  
Processing triggers for systemd (237-3ubuntu10.39) ...  
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...  
vivek@vivek-VirtualBox:~$
```

2. Docker

```
sudo: docker: command not found  
vivek2@vivek2-VirtualBox:~$ sudo curl -sSL https://get.docker.com/ | sh  
# Executing docker install script, commit: 442e66405c304fa92af8aadaa1d9b31bf4b0ad94  
+ sudo -E sh -c apt-get update -qq >/dev/null  
+ sudo -E sh -c DEBIAN_FRONTEND=noninteractive apt-get install -y -qq apt-transport-https ca-certificates curl >/dev/null  
+ sudo -E sh -c curl -fsSL "https://download.docker.com/linux/ubuntu/gpg" | apt-key add -qq - >/dev/null  
Warning: apt-key output should not be parsed (stdout is not a terminal)  
+ sudo -E sh -c echo "deb [arch=amd64] https://download.docker.com/linux/ubuntu bionic stable" > /etc/apt/sources.list.d/docker.list  
+ sudo -E sh -c apt-get update -qq >/dev/null
```

```
Activities Terminal Sat 16:11
vivek2@vivek2-VirtualBox: ~
File Edit View Search Terminal Help
Go version: go1.12.17
Git commit: afacb8b7f0
Built: Wed Mar 11 01:24:19 2020
OS/Arch: linux/amd64
Experimental: false
containerd:
Version: 1.2.13
GitCommit: 7ad184331fa3e55e52b890ea95e65ba581ae3429
runc:
Version: 1.0.0-rc10
GitCommit: dc9208a3303feef5b3839f4323d9beb36df0a9dd
docker-init:
Version: 0.18.0
GitCommit: fec3683
If you would like to use Docker as a non-root user, you should now consider
adding your user to the "docker" group with something like:

    sudo usermod -aG docker vivek2

Remember that you will have to log out and back in for this to take effect!

WARNING: Adding a user to the "docker" group will grant the ability to run
containers which can be used to obtain root privileges on the
docker host.
Refer to https://docs.docker.com/engine/security/security/#docker-daem
on-attack-surface
for more information.
vivek2@vivek2-VirtualBox:~$
```

3. Java Development Kit(JDK) 11

```
vivek2@vivek2-VirtualBox:~$ sudo apt-get install default-jdk
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  efibootmgr libfwup1 ubuntu-fan
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  ca-certificates-java default-jdk-headless default-jre default-jre-headless fonts-dejavu-extra
  java-common libatk-wrapper-java libatk-wrapper-java-jni libgif7 libice-dev
  libpthread-stubs0-dev libsm-dev libx11-dev libx11-doc libxau-dev libxcb1-dev libxdmcp-dev
  libxt-dev openjdk-11-jdk openjdk-11-jdk-headless openjdk-11-jre openjdk-11-jre-headless
  x11proto-core-dev x11proto-dev xorg-sgml-doctools xtrans-dev
Suggested packages:
  libice-doc libsm-doc libxcb-doc libxt-doc openjdk-11-demo openjdk-11-source visualvm
  fonts-ipafont-gothic fonts-ipafont-mincho fonts-wqy-microhei | fonts-wqy-zenhei
The following NEW packages will be installed:
  ca-certificates-java default-jdk default-jdk-headless default-jre default-jre-headless
  fonts-dejavu-extra java-common libatk-wrapper-java libatk-wrapper-java-jni libgif7 libice-dev
  libpthread-stubs0-dev libsm-dev libx11-dev libx11-doc libxau-dev libxcb1-dev libxdmcp-dev
  libxt-dev openjdk-11-jdk openjdk-11-jdk-headless openjdk-11-jre openjdk-11-jre-headless
  x11proto-core-dev x11proto-dev xorg-sgml-doctools xtrans-dev
0 upgraded, 27 newly installed, 0 to remove and 6 not upgraded.
Need to get 238 MB of archives.
After this operation, 400 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
```

4. OpenDaylight

```
Activities Terminal Sat 18:28
vivek@vivek-VirtualBox -
See 'docker run --help'.
vivek@vivek-VirtualBox:~$ sudo docker rm fe0be65f31e2fc0e32c7ca3192dafab7e0f070182265962dd6e39412934819e
fe0be65f31e2fc0e32c7ca3192dafab7e0f070182265962dd6e39412934819e
vivek@vivek-VirtualBox:~$ sudo docker run -d -p 8081:8081 --name=opendaylight yurekten/odl-carbon
14a49f749db9316575460f5c7ec47bc1e9ca3d547cd778825c7a310c7bbdcde
vivek@vivek-VirtualBox:~$ ssh -p 8101 karaf@localhost
Warning: REMOTE HOST IDENTIFICATION HAS CHANGED!
It is possible that someone is doing something nasty!
Someone could be eavesdropping on you right now (man-in-the-middle attack)!
It is also possible that a host key has just been changed.
The fingerprint for the RSA key sent by the remote host is
sha256:290dgdflQWGL2323RGPvOkt08gu0qBvF32Q0wYrV4.
Please contact your system administrator.
Add correct host key in /home/vivek/.ssh/known_hosts to get rid of this message.
Offending RSA key in /home/vivek/.ssh/known_hosts:5
remove with:
ssh-keygen -f "/home/vivek/.ssh/known_hosts" -R "[localhost]:8101"
RSA host key for [localhost]:8101 has changed and you have requested strict checking.
Host key verification failed.
vivek@vivek-VirtualBox:~$ ssh-keygen -f "/home/vivek/.ssh/known_hosts" -R "[localhost]:8101"
# Host [localhost]:8101 found: line 5
/home/vivek/.ssh/known_hosts updated.
Original contents retained as /home/vivek/.ssh/known_hosts.old
vivek@vivek-VirtualBox:~$ ssh -p 8101 karaf@localhost
The authenticity of host '[localhost]:8101 ([127.0.0.1]:8101)' can't be established.
RSA key fingerprint is sha256:290dgdflQWGL2323RGPvOkt08gu0qBvF32Q0wYrV4.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[localhost]:8101' (RSA) to the list of known hosts.
Password authentication
Password:

OpenDaylight

Hit '<tab>' for a list of available commands
and '[cmd] --help' for help on a specific command.
Hit 'ctrl-d' or type 'system:shutdown' or 'logout' to shutdown OpenDaylight.

opendaylight-user@root>feature:list | grep odl-dluxapps
odl-dluxapps-applications | 0.5.0-Carbon | odl-dlux-0.5.0-Carbon | OpenDaylight DluxApps all applications
odl-dluxapps-nodes | 0.5.0-Carbon | odl-dlux-0.5.0-Carbon | Enable nodes in OpenDaylight dlux
odl-dluxapps-topology | 0.5.0-Carbon | odl-dlux-0.5.0-Carbon | Enable nodes in OpenDaylight dlux
odl-dluxapps-yangui | 0.5.0-Carbon | odl-dlux-0.5.0-Carbon | Enable Yang UI in OpenDaylight dlux
odl-dluxapps-yangman | 0.5.0-Carbon | odl-dlux-0.5.0-Carbon | Enable Yangman in OpenDaylight dlux
odl-dluxapps-yangvisualizer | 0.5.0-Carbon | odl-dlux-0.5.0-Carbon | Enable Yang visualizer in OpenDaylight dlux
odl-dluxapps-yangutils | 0.5.0-Carbon | odl-dlux-0.5.0-Carbon | Loads yangutils library in OpenDaylight dlux
opendaylight-user@root>feature:install odl-restconf odl-ndsa1-apidocs odl-dlux-core odl-dluxapps-nodes odl-dluxapps-topology odl-dluxapps-yangui odl-dluxapps-yangvisualizer odl-dluxapps-yangman odl-l2swit
ch-switch
ch-switch
```

5. Wireshark

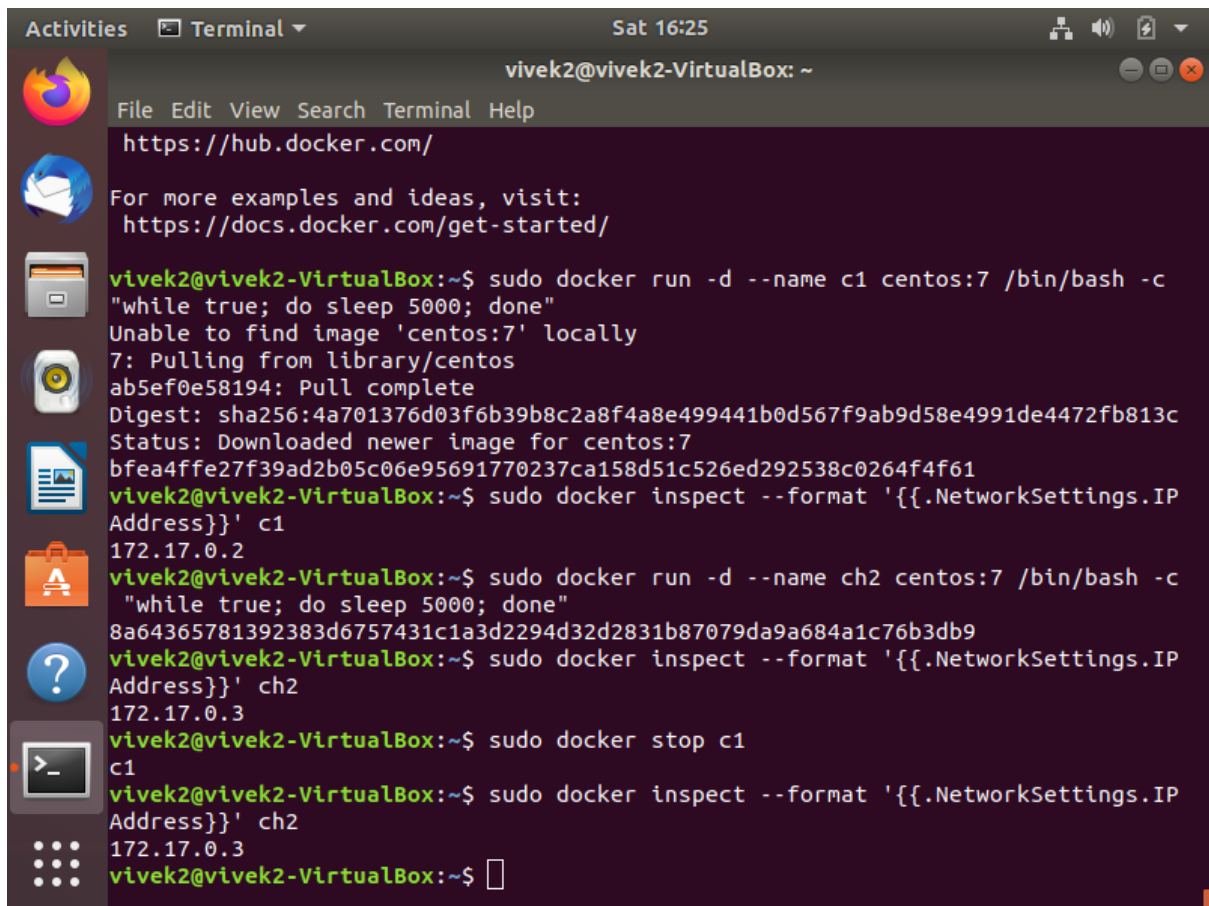
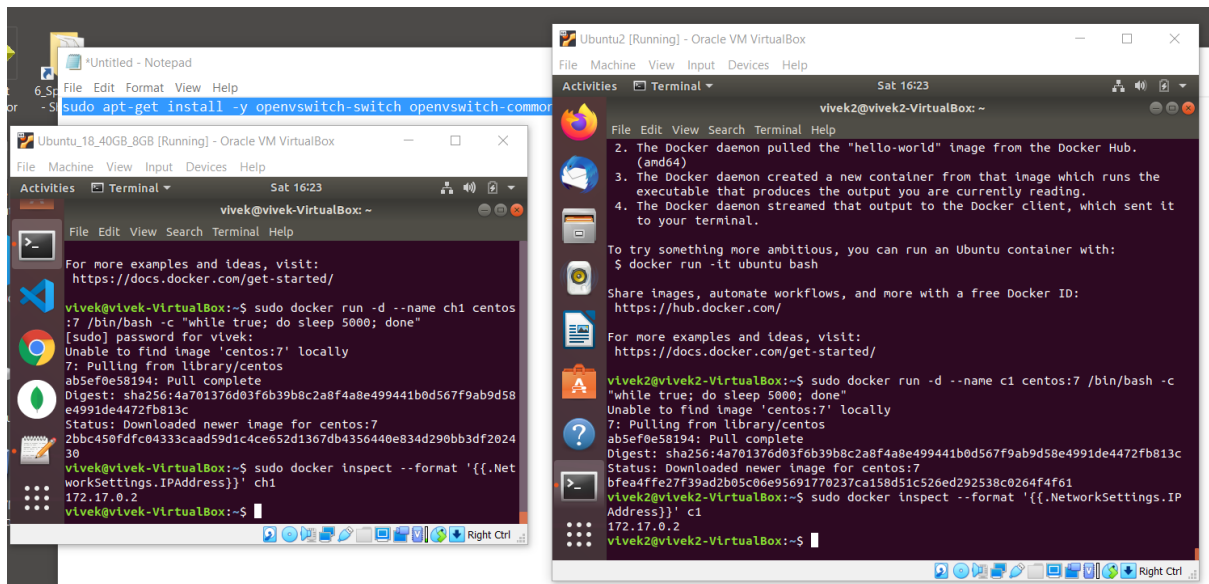
```
Connection to localhost closed.
vivek@vivek-VirtualBox:~$ sudo docker pull manell/wireshark
[sudo] password for vivek:
Using default tag: latest
latest: Pulling from manell/wireshark
d3938036b19c: Pull complete
a9b30c108bda: Pull complete
67de21feec18: Pull complete
817da545be2b: Pull complete
d967c497ce23: Pull complete
a8c836392158: Pull complete
266183ba57c5: Pull complete
Digest: sha256:5ca01277b780c0403de99143c513e3022bac10ab7d8fc7f6b66bda9d7f461aaa
Status: Downloaded newer image for manell/wireshark:latest
docker.io/manell/wireshark:latest
vivek@vivek-VirtualBox:~$ sudo docker run -ti --net=host --privileged -v $HOME:/root:ro -e XAUTHO
RITY=/root/.Xauthority -e DISPLAY=$DISPLAY manell/wireshark
..
No protocol specified
..
QXcbConnection: Could not connect to display :0
vivek@vivek-VirtualBox:~$
```

Connect docker using OVS bridge:

We connect the docker to one of the virtual ethernet device pair and the other device to the Open vSwitch bridge br-int (the core bridge).

Steps:

1. Run the docker container on both the host machines and get the IP Addresses. We found that that both the containers have the same IP Address. So we ran another container and killed the first one on one of the host machines.



2. Then we tried to ping the other container (in the other host) from this container and found that the destination was unreachable.

```
vivek@vivek-VirtualBox:~$ sudo docker run -d --name ch1 centos:7 /bin/bash -c "while true; do sleep 5000; done"
[sudo] password for vivek:
Unable to find image 'centos:7' locally
7: Pulling from library/centos
ab5ef0e58194: Pull complete
Digest: sha256:4a701376d03f6b39b8c2a8f4a8e499441b0d567f9ab9d58e4991de4472fb813c
Status: Downloaded newer image for centos:7
2bbc450fd0c04333caad59d1c4ce652d1367db4356440e834d290bb3df202430
vivek@vivek-VirtualBox:~$ sudo docker inspect --format '{{.NetworkSettings.IPAddress}}' ch1
172.17.0.2
vivek@vivek-VirtualBox:~$ sudo docker exec -it ch1 bash
[root@2bbc450fd0c0 /]# ping 172.17.0.3
PING 172.17.0.3 (172.17.0.3) 56(84) bytes of data.
From 172.17.0.2 icmp_seq=1 Destination Host Unreachable
From 172.17.0.2 icmp_seq=2 Destination Host Unreachable
From 172.17.0.2 icmp_seq=3 Destination Host Unreachable
From 172.17.0.2 icmp_seq=4 Destination Host Unreachable
From 172.17.0.2 icmp_seq=5 Destination Host Unreachable
From 172.17.0.2 icmp_seq=6 Destination Host Unreachable
^C
--- 172.17.0.3 ping statistics ---
8 packets transmitted, 0 received, 100% packet loss, time 7207ms
pipe 4
[root@2bbc450fd0c0 /]# exit
exit
vivek@vivek-VirtualBox:~$
```

3. We started setting up the OVS bridge to establish the connection between the 2 containers on different hosts.

```
vivek@vivek-VirtualBox:~$ sudo ip link add veth0 type veth peer name veth1
vivek@vivek-VirtualBox:~$ sudo ovs-vsctl add-port br-int veth1
vivek@vivek-VirtualBox:~$ sudo apt install bridge-utils
sudo: app: command not found
vivek@vivek-VirtualBox:~$ sudo apt install bridge-utils
Reading package lists... Done
Building dependency tree
Reading state information... Done
bridge-utils is already the newest version (1.5-15ubuntu1).
bridge-utils set to manually installed.
The following packages were automatically installed and are no longer required:
  efibootmgr libfwupd1 ubuntu-fan
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 6 not upgraded.
vivek@vivek-VirtualBox:~$ sudo brctl addif docker0 veth0
vivek@vivek-VirtualBox:~$ sudo ip link set veth1 up
vivek@vivek-VirtualBox:~$ sudo ip link set veth0 up
vivek@vivek-VirtualBox:~$
```

```

vivek@vivek-VirtualBox:~$ ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT group default ql
en 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP mode DEFAULT group
default qlen 1000
    link/ether 08:00:27:2b:7c:52 brd ff:ff:ff:ff:ff:ff
3: docker0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP mode DEFAULT group
default
    link/ether 02:42:1f:f9:a7:39 brd ff:ff:ff:ff:ff:ff
7: veth2641914@if6: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master docker0 state
UP mode DEFAULT group default
    link/ether 46:b1:eb:92:84:db brd ff:ff:ff:ff:ff:ff link-netnsid 0
8: ovs-system: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN mode DEFAULT group default ql
en 1000
    link/ether 42:59:26:b7:36:6d brd ff:ff:ff:ff:ff:ff
9: br-int: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN mode DEFAULT group default qlen 1
000
    link/ether 62:58:87:4e:9a:4f brd ff:ff:ff:ff:ff:ff
10: veth1@veth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master ovs-system state
UP mode DEFAULT group default qlen 1000
    link/ether a2:ac:8c:7c:a8:e0 brd ff:ff:ff:ff:ff:ff
11: veth0@veth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master docker0 state UP
mode DEFAULT group default qlen 1000
    link/ether 22:e0:a6:39:2d:de brd ff:ff:ff:ff:ff:ff
vivek@vivek-VirtualBox:~$ sudo ovs-vsctl add-port br-int gre0 -- set interface gre0 type=gre opti
ons:remote_ip=192.168.1.98
vivek@vivek-VirtualBox:~$ sudo ovs-vsctl show
38254345-b013-47fd-9bb9-48f183cb054f
    Bridge br-int
        Port "veth1"
        Interface "veth1"
        Port "gre0"
        Interface "gre0"
            type: gre
            options: {remote_ip="192.168.1.98"}
        Port br-int
        Interface br-int
            type: internal
    ovs_version: "2.9.5"
vivek@vivek-VirtualBox:~$ sudo brctl show
bridge name      bridge id      STP enabled    interfaces
docker0          8000.02421ff9a739    no             veth0
veth2641914
vivek@vivek-VirtualBox:~$

```

4. After setting up the bridge, we were able to ping from one container to the other.

```

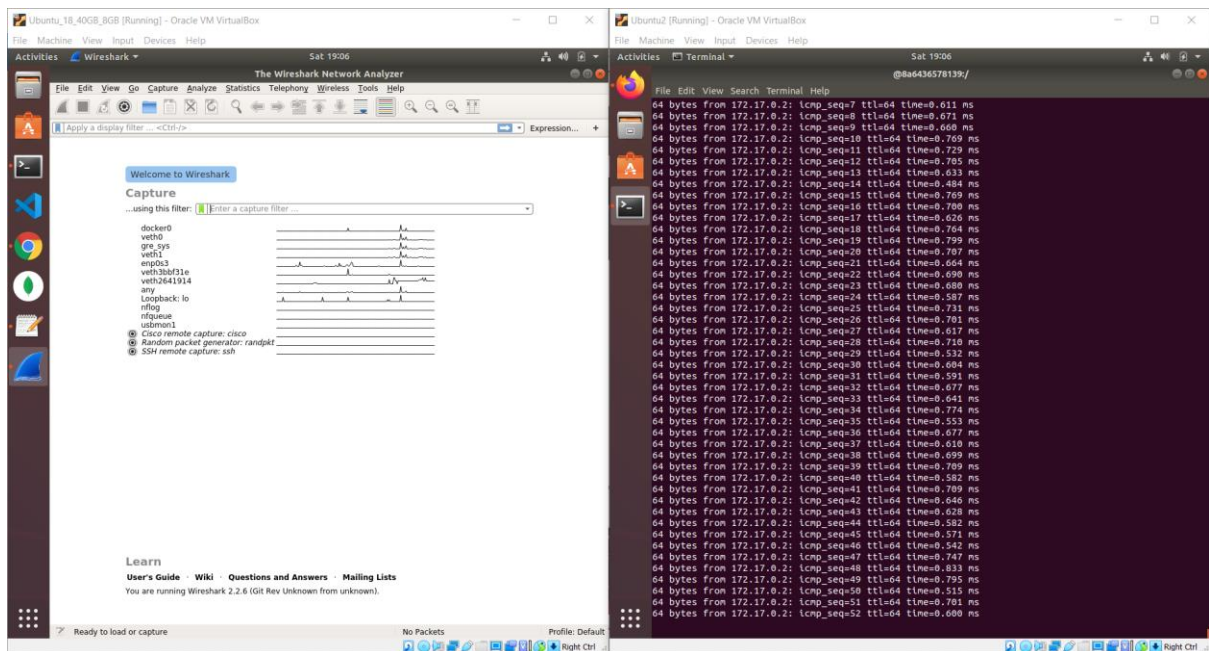
vivek@vivek-VirtualBox:~$ ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT group default ql
en 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP mode DEFAULT group
default qlen 1000
    link/ether 08:00:27:2b:7c:52 brd ff:ff:ff:ff:ff:ff
3: docker0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP mode DEFAULT group
default
    link/ether 02:42:1f:f9:a7:39 brd ff:ff:ff:ff:ff:ff
7: veth2641914@if6: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master docker0 state
UP mode DEFAULT group default
    link/ether 46:b1:eb:92:84:db brd ff:ff:ff:ff:ff:ff link-netnsid 0
8: ovs-system: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN mode DEFAULT group default ql
en 1000
    link/ether 42:59:26:b7:36:6d brd ff:ff:ff:ff:ff:ff
9: br-int: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN mode DEFAULT group default qlen 1
000
    link/ether 62:58:87:4e:9a:4f brd ff:ff:ff:ff:ff:ff
10: veth1@veth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master ovs-system state
UP mode DEFAULT group default qlen 1000
    link/ether a2:ac:8c:7c:a8:e0 brd ff:ff:ff:ff:ff:ff
11: veth0@veth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master docker0 state UP
mode DEFAULT group default qlen 1000
    link/ether 22:e0:a6:39:2d:de brd ff:ff:ff:ff:ff:ff
vivek@vivek-VirtualBox:~$ sudo ovs-vsctl add-port br-int gre0 -- set interface gre0 type=gre opti
ons:remote_ip=192.168.1.98
vivek@vivek-VirtualBox:~$ sudo ovs-vsctl show
38254345-b013-47fd-9bb9-48f183cb054f
    Bridge br-int
        Port "veth1"
        Interface "veth1"
        Port "gre0"
        Interface "gre0"
            type: gre
            options: {remote_ip="192.168.1.98"}
        Port br-int
        Interface br-int
            type: internal
    ovs_version: "2.9.5"
vivek@vivek-VirtualBox:~$ sudo brctl show
bridge name      bridge id      STP enabled    interfaces
docker0          8000.02421ff9a739    no             veth0
veth2641914
vivek@vivek-VirtualBox:~$ sudo docker exec -it ch1 bash
[root@2bbc450dfdc0 /]# ping 172.17.0.3
PING 172.17.0.3 (172.17.0.3) 56(84) bytes of data:
64 bytes from 172.17.0.3: icmp_seq=1 ttl=64 time=1.17 ms
64 bytes from 172.17.0.3: icmp_seq=2 ttl=64 time=0.693 ms
64 bytes from 172.17.0.3: icmp_seq=3 ttl=64 time=0.555 ms
64 bytes from 172.17.0.3: icmp_seq=4 ttl=64 time=0.687 ms
64 bytes from 172.17.0.3: icmp_seq=5 ttl=64 time=0.752 ms
64 bytes from 172.17.0.3: icmp_seq=6 ttl=64 time=0.818 ms
64 bytes from 172.17.0.3: icmp_seq=7 ttl=64 time=0.679 ms
^C
-- 172.17.0.3 ping statistics --
8 packets transmitted, 8 received, 0% packet loss, time 7124ms
rtt min/avg/max/mdev = 0.555/0.754/1.178/0.175 ms
[root@2bbc450dfdc0 /]#

vivek@vivek-VirtualBox:~$ sudo docker exec -it ch2 bash
[root@8a643657b139 /]# ping 172.17.0.2
PING 172.17.0.2 (172.17.0.2) 56(84) bytes of data:
64 bytes from 172.17.0.2: icmp_seq=1 ttl=64 time=1.78 ms
64 bytes from 172.17.0.2: icmp_seq=2 ttl=64 time=0.639 ms
64 bytes from 172.17.0.2: icmp_seq=3 ttl=64 time=0.798 ms
64 bytes from 172.17.0.2: icmp_seq=4 ttl=64 time=0.666 ms
64 bytes from 172.17.0.2: icmp_seq=5 ttl=64 time=0.804 ms
64 bytes from 172.17.0.2: icmp_seq=6 ttl=64 time=0.812 ms
64 bytes from 172.17.0.2: icmp_seq=7 ttl=64 time=0.762 ms
64 bytes from 172.17.0.2: icmp_seq=8 ttl=64 time=0.771 ms
64 bytes from 172.17.0.2: icmp_seq=9 ttl=64 time=0.672 ms
64 bytes from 172.17.0.2: icmp_seq=10 ttl=64 time=0.652 ms
^C
-- 172.17.0.2 ping statistics --
10 packets transmitted, 10 received, 0% packet loss, time 9363ms
rtt min/avg/max/mdev = 0.639/0.835/1.783/0.324 ms
[root@8a643657b139 /]#

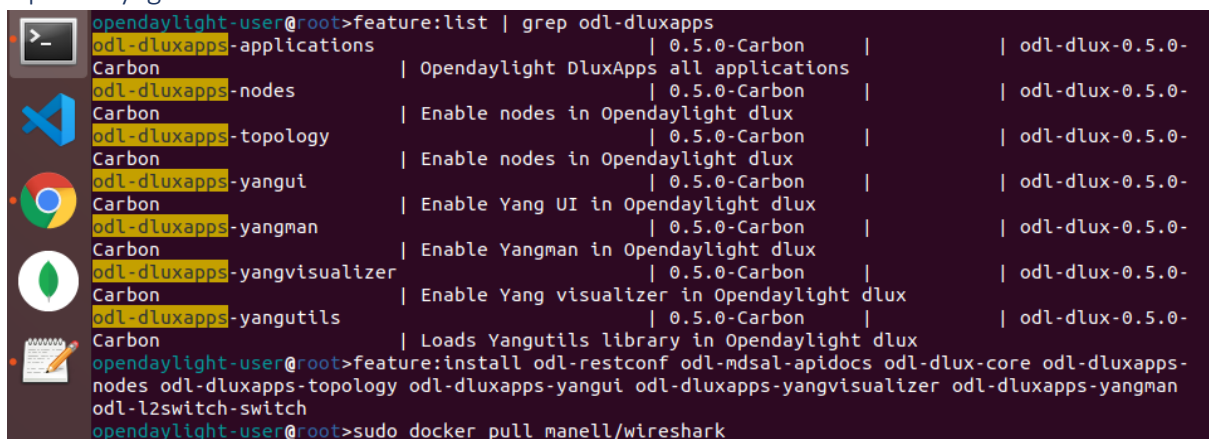
```


Wireshark:

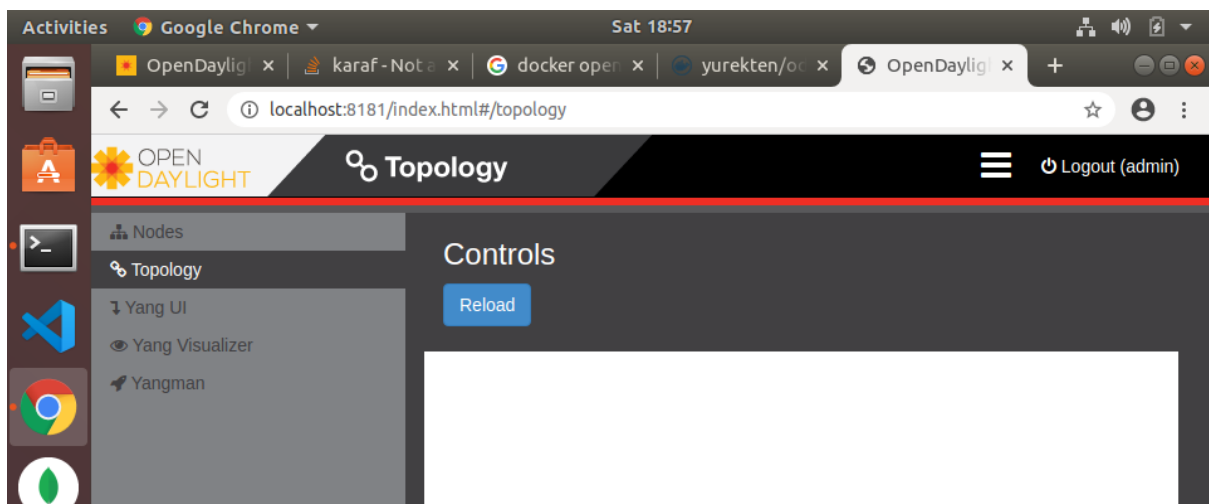
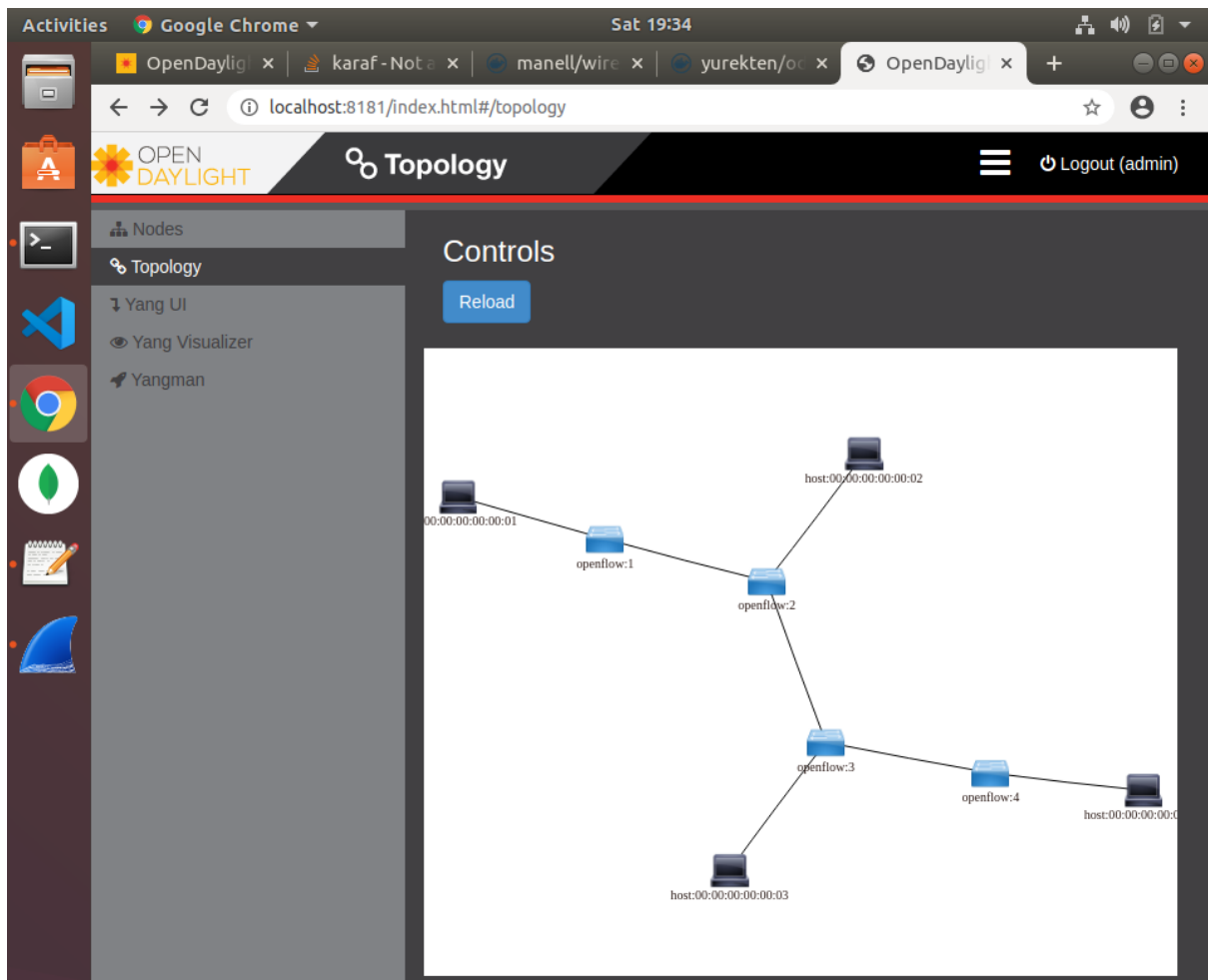
We pinged from one container to another and checked the traffic on Wireshark.



Open Daylight Feature Install:



We installed the OpenDaylight features and tested the mininet topology in this container.

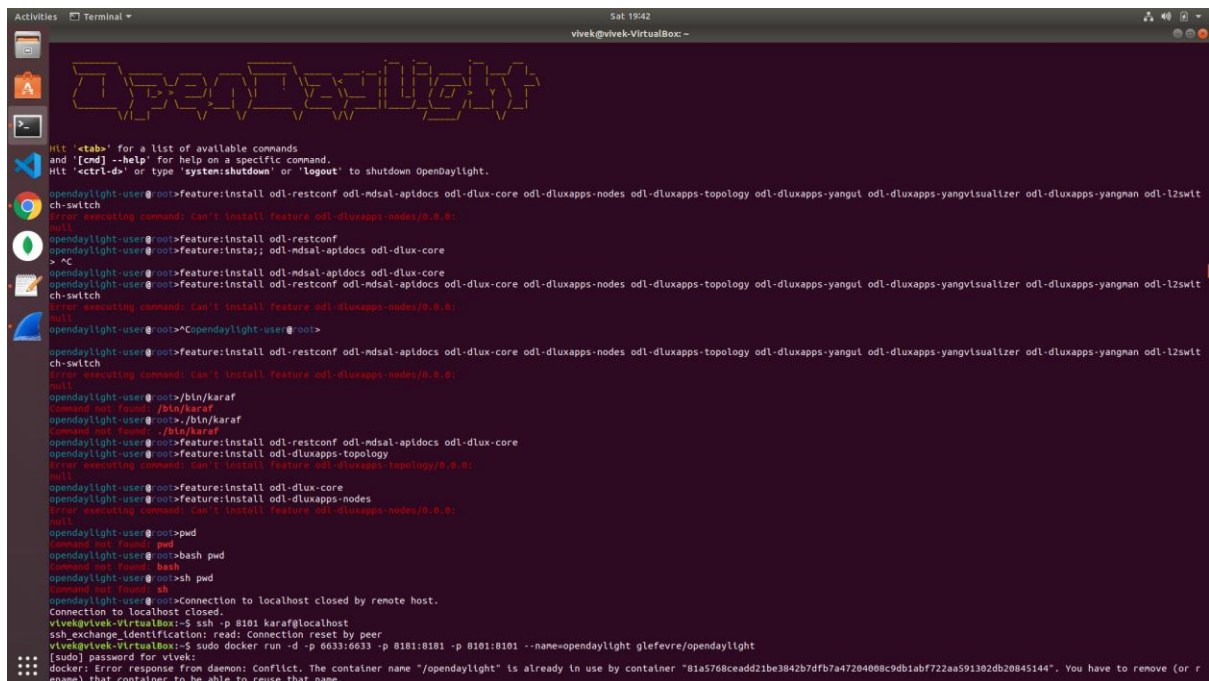


Challenges Faced

- Main hurdle was learning Docker and containerization of applications.
- Creating 2 heavy weight VMs on the same laptop machine was a challenge since we have limited processing power.
- Containers on different hosts had the same IP Addresses, so we had to respawn another to get a different IP on one of them.
- Issues with JAVA versions: Initially we had JDK 8 installed, but an upgrade changed it to JDK 11 but JAVA_HOME variable was still pointing to JDK 8 path which had become invalid.

Figuring out that the JDK had upgraded took us some time since the Open Daylight was failing to launch.

- Initially we had installed OpenDaylight Boron which did not have the dluxapps features. After figuring out the list of features that were available in Boron, we switched to Carbon. During this, we faced issues with docker images not stopping, we had to stop the images forcefully before we were able to switch to Carbon. Also when we re-started the container, SSH into the container didn't work as the host key had changed. So we had to generate and save the new SSH key.



```
Hit '<tab>' for a list of available commands
and '[cmd]' --help' for help on a specific command.
Hit '<ctrl-c>' or type 'system:shutdown' or 'logout' to shutdown OpenDaylight.

opendaylight-user@root>feature:install odl-restconf odl-ndsal-apidocs odl-dlux-core odl-dluxapps-nodes odl-dluxapps-topology odl-dluxapps-yangut odl-dluxapps-yangvisualizer odl-dluxapps-yangman odl-l2swit
ch-switch
Error executing command: Can't install feature odl-dluxapps-nodes/0.0.0:
null

opendaylight-user@root>feature:install odl-restconf
opendaylight-user@root>feature:install odl-ndsal-apidocs odl-dlux-core
> ^C
opendaylight-user@root>feature:install odl-ndsal-apidocs odl-dlux-core
opendaylight-user@root>feature:install odl-restconf odl-ndsal-apidocs odl-dlux-core odl-dluxapps-nodes odl-dluxapps-topology odl-dluxapps-yangut odl-dluxapps-yangvisualizer odl-dluxapps-yangman odl-l2swit
ch-switch
Error executing command: Can't install feature odl-dluxapps-nodes/0.0.0:
null

opendaylight-user@root>^C
opendaylight-user@root>feature:install odl-restconf odl-ndsal-apidocs odl-dlux-core odl-dluxapps-nodes odl-dluxapps-topology odl-dluxapps-yangut odl-dluxapps-yangvisualizer odl-dluxapps-yangman odl-l2swit
ch-switch
Error executing command: Can't install feature odl-dluxapps-nodes/0.0.0:
null

opendaylight-user@root>./bin/karaf
Command not found: ./bin/karaf
opendaylight-user@root>./bin/karaf
Command not found: ./bin/karaf
opendaylight-user@root>feature:install odl-restconf odl-ndsal-apidocs odl-dlux-core
opendaylight-user@root>feature:install odl-dluxapps-topology
Error executing command: Can't install feature odl-dluxapps-topology/0.0.0:
null

opendaylight-user@root>feature:install odl-dlux-core
opendaylight-user@root>feature:install odl-dluxapps-nodes
Error executing command: Can't install feature odl-dluxapps-nodes/0.0.0:
null

opendaylight-user@root>pwd
Command not found: pwd
opendaylight-user@root>bash pwd
Command not found: bash
opendaylight-user@root>sh pwd
Command not found: sh
opendaylight-user@root>Connection to localhost closed by remote host.
Connection to localhost closed.
vivek@vivek-VirtualBox:~$ ssh -p 8101 karaf@localhost
ssh_exchange_identification: read: Connection reset by peer
vivek@vivek-VirtualBox:~$ sudo docker run -d -p 6633:6633 -p 8101:8101 -p 8101:8101 --name=opendaylight glefevre/opendaylight
[sudo] password for vivek:
docker: Error response from daemon: Conflict. The container name "/opendaylight" is already in use by container "81a5768ceadd21be3842b7dfb7a4728408c9db1abf722aa591302db20845144". You have to remove (or r
ename) that container to be able to reuse that name.
```

Conclusion:

Docker is a easy and efficient way to build and deploy any application. We were successful in setting up a communication channel between 2 docker containers on different host machines using Open vSwitch. We were able to use a containerized OpenDaylight-Carbon to display the topology of a Mininet running on a separate VM.

References:

- [1] [https://en.wikipedia.org/wiki/Docker_\(software\)](https://en.wikipedia.org/wiki/Docker_(software))
- [2] https://en.wikipedia.org/wiki/Open_vSwitch
- [3] <https://www.imperva.com/blog/what-is-gre-tunnel/>
- [4] <https://hub.docker.com/r/yurekten/odl-carbon/>
- [5] <https://goyalankit.com/blog/linux-bridge>
- [6] <http://man7.org/linux/man-pages/man4/veth.4.html>
- [7] <https://en.wikipedia.org/wiki/Wireshark>
- [8] <https://app.box.com/s/t4yob7umn1m88b4p4vmbeu1scele31e/folder/63985323299>