NETWORK VIRTUALIZATION

Submitted By

Group::IT
NAND KISHOR KUMAR(20192017)
ANUBHAV MISHRA(20198014)
ABHISHEK BAGE(20198021)
AKASH RAJ(20198081)

Introduction :

Network management currently is undergoing changes towards more flexible ways . This trend is stimulated by Network Virtualization and Software Defined Networks (SDN) that emerged in recent years. These technologies allow networks to be run in a more flexible and cost efficient manner, e.g., by increasing network resource utilization and by decreasing operational costs.

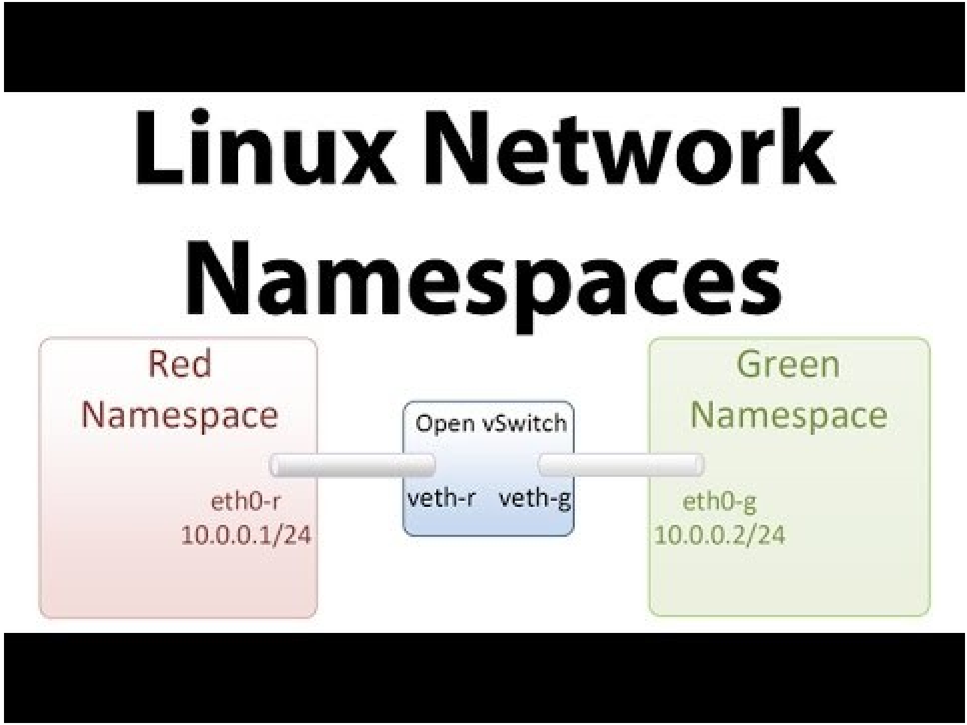
We can divide our Local Area Networks (LANs) into virtual networks and VLANs. The idea of this helps a lot in load balancing.It is used in application development and testing to mimic real world hardware and system software.

Network Virtualization

Network virtualization means abstracting network resources that were originally used in hardware to software..With network virtualization, digital service providers can optimize their server resources (i.e. fewer idle servers), allow them to use standard servers for functions that once required expensive proprietary hardware, and generally improve the speed, flexibility, and reliability of their networks.Network virtualization allows network functions, hardware resources, and software resources to be delivered independent of hardware

Network Namespace :

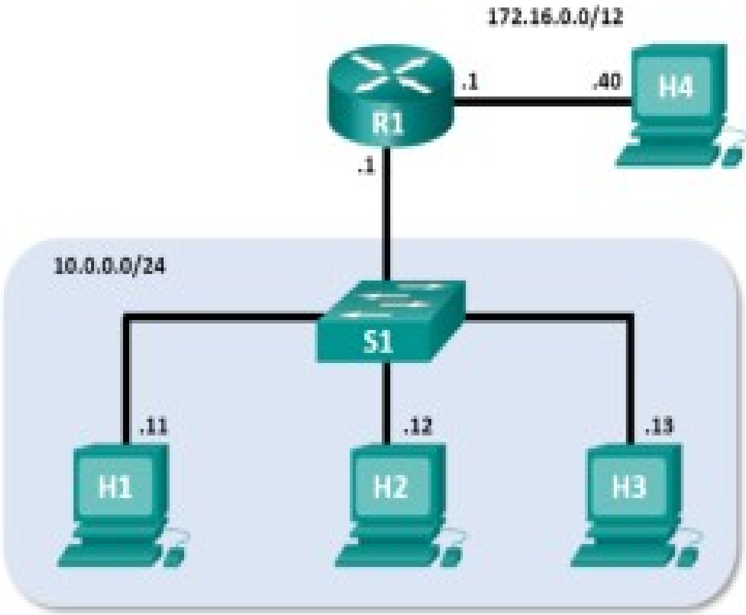
Network namespaces is a Linux kernel feature allowing us to isolate network environments through virtualization , it is a logical copy of the network stack from the host . We can assign Ip address ,network Interfaces,routing table to the virtual systems we create.



Mininet :

Mininet is a open source software emulator for creating large networks on a single machine. It allows us to create different topologies consisting of OpenFlow switches and hosts.We can easily create a realistic virtual network using mininet. Topologies can vary from a single switch to more complicated topologies c . We can easily interact with our network using mininet CLI , customize it , share it or deploy it .

Mininet Topology



Connect Network Namespaces to internet using Open Virtual Switch:

Create Network Namespaces : sudo ip netns add <namespace_name>

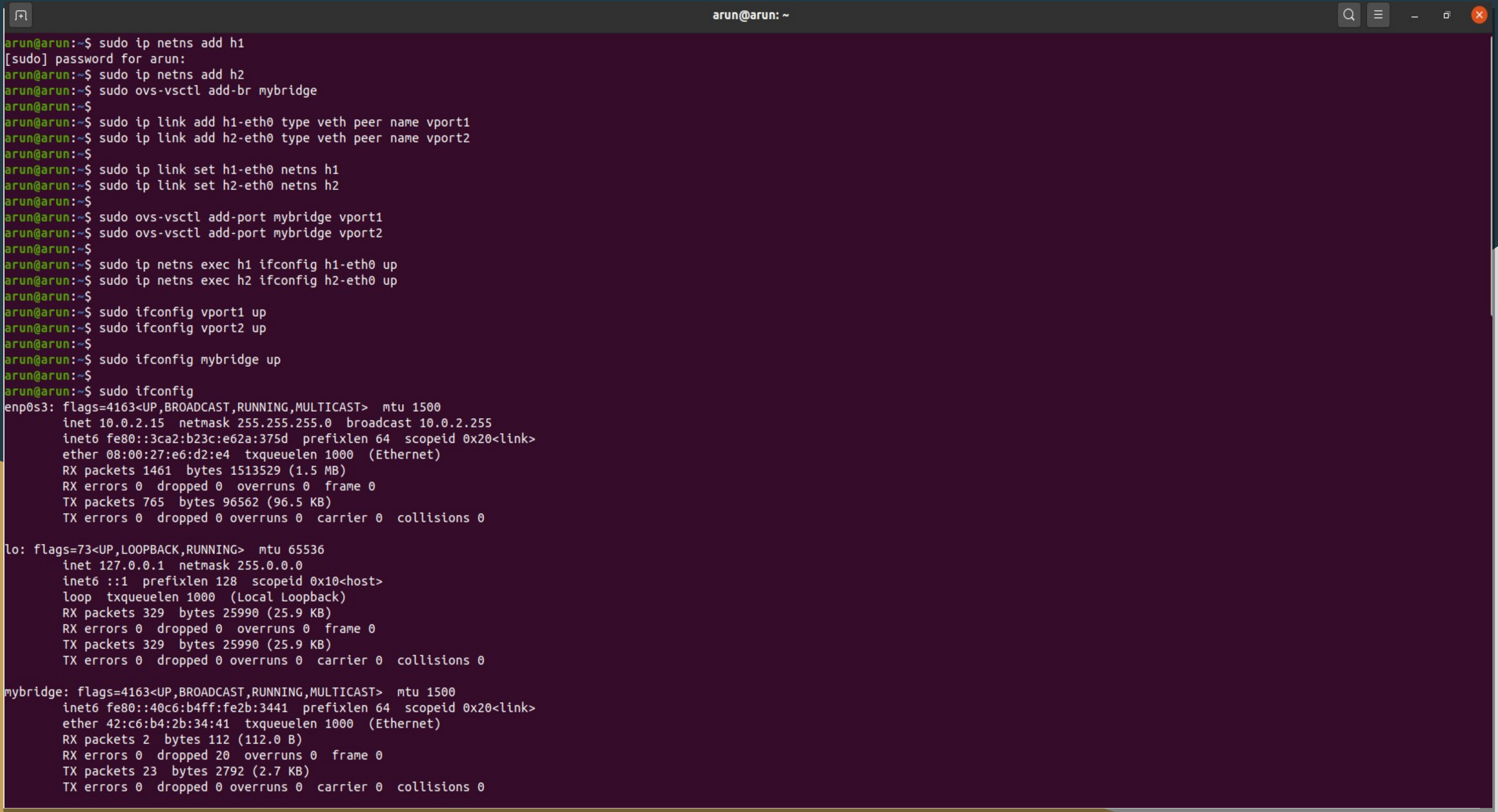
Create Layer-2 Virtual Network with OpenVSwitch : sudo ovs-vsctl add-br <bridge_name>

Create Network Interfaces Links and add those to bridge and namespaces :

```
sudo ip link add <interface1_name> type veth peer name <interface2_name>
sudo ip link set <interface_name> netns <namespace_name>
sudo ovs-vsctl add-port <bridge_name> <interface_name>
```

Enable the interfaces and Bridge :

```
sudo ip netns exec <namespace_name>  ifconfig <interface_name> up
sudo ifconfig <bridgeInterface_name>  up
sudo ifconfig <bridge_name> up
```



```
arun@arun:~$ sudo ip netns add h1
[sudo] password for arun:
arun@arun:~$ sudo ip netns add h2
arun@arun:~$ sudo ovs-vsctl add-br mybridge
arun@arun:~$
arun@arun:~$ sudo ip link add h1-eth0 type veth peer name vport1
arun@arun:~$ sudo ip link add h2-eth0 type veth peer name vport2
arun@arun:~$
arun@arun:~$ sudo ip link set h1-eth0 netns h1
arun@arun:~$ sudo ip link set h2-eth0 netns h2
arun@arun:~$
arun@arun:~$ sudo ovs-vsctl add-port mybridge vport1
arun@arun:~$ sudo ovs-vsctl add-port mybridge vport2
arun@arun:~$
arun@arun:~$ sudo ip netns exec h1 ifconfig h1-eth0 up
arun@arun:~$ sudo ip netns exec h2 ifconfig h2-eth0 up
arun@arun:~$
arun@arun:~$ sudo ifconfig vport1 up
arun@arun:~$ sudo ifconfig vport2 up
arun@arun:~$
arun@arun:~$ sudo ifconfig mybridge up
arun@arun:~$
arun@arun:~$ sudo ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::3ca2:b23c:e62a:375d  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:e6:d2:e4  txqueuelen 1000  (Ethernet)
        RX packets 1461  bytes 1513529 (1.5 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 765  bytes 96562 (96.5 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 329  bytes 25990 (25.9 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 329  bytes 25990 (25.9 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

mybridge: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet6 fe80::40c6:b4ff:fe2b:3441  prefixlen 64  scopeid 0x20<link>
        ether 42:c6:b4:2b:34:41  txqueuelen 1000  (Ethernet)
        RX packets 2  bytes 112 (112.0 B)
        RX errors 0  dropped 20  overruns 0  frame 0
        TX packets 23  bytes 2792 (2.7 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Add the Internet Network interface to the bridge :

```
Sudo ovs-vsctl add-port <bridge_name> enp0s3/eth0(Interface_Name)
```

Assign the IP address to bridge and Interfaces using dhclient :

(dhclient : requests IP Address from a DHCP Server)

```
Sudo dhclient <bridge_name>
```

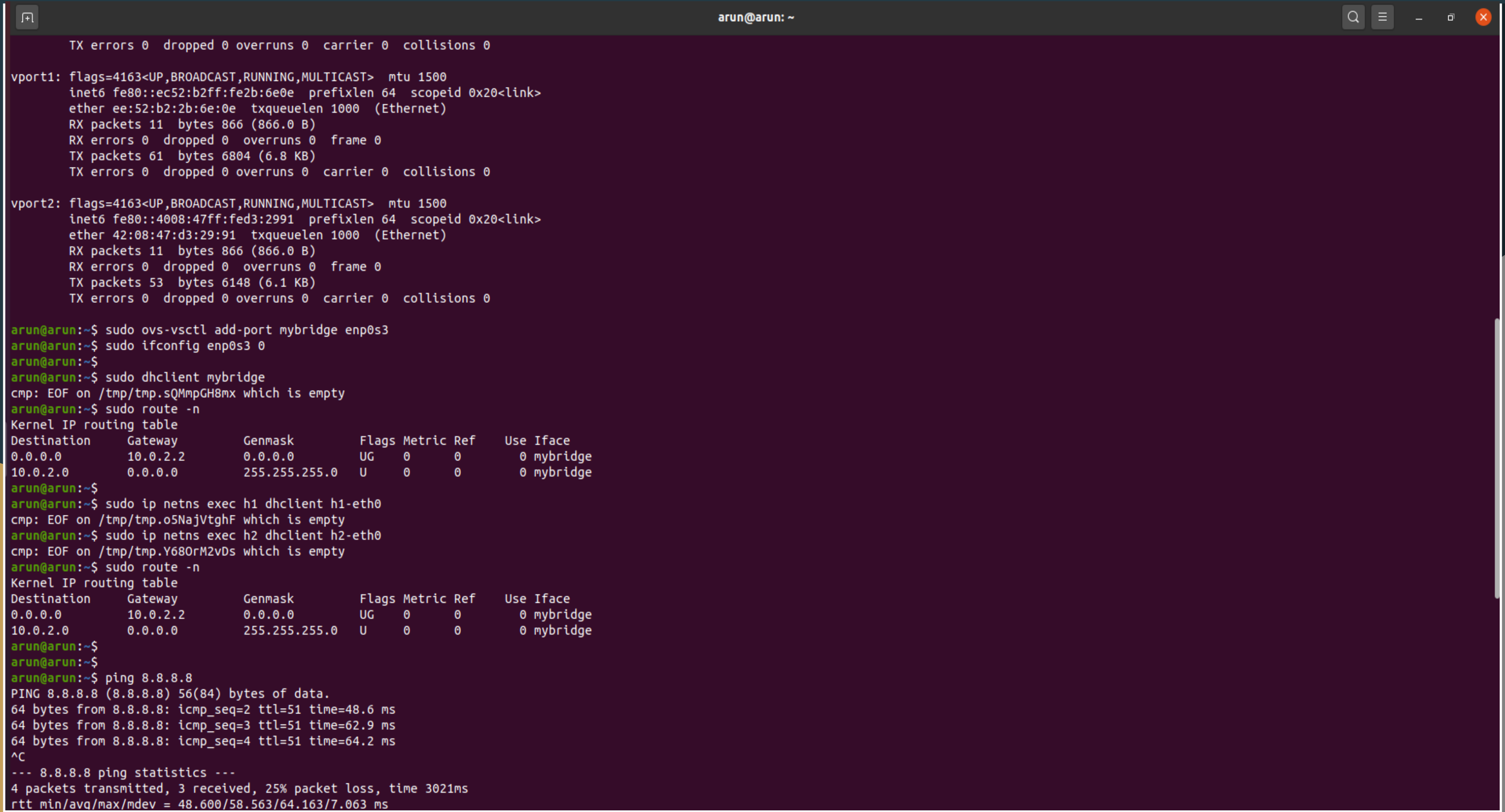
```
Sudo ip netns exec <namespace_name> dhclient <interface_name>
```

See the Kernel IP Routing table : sudo route -n

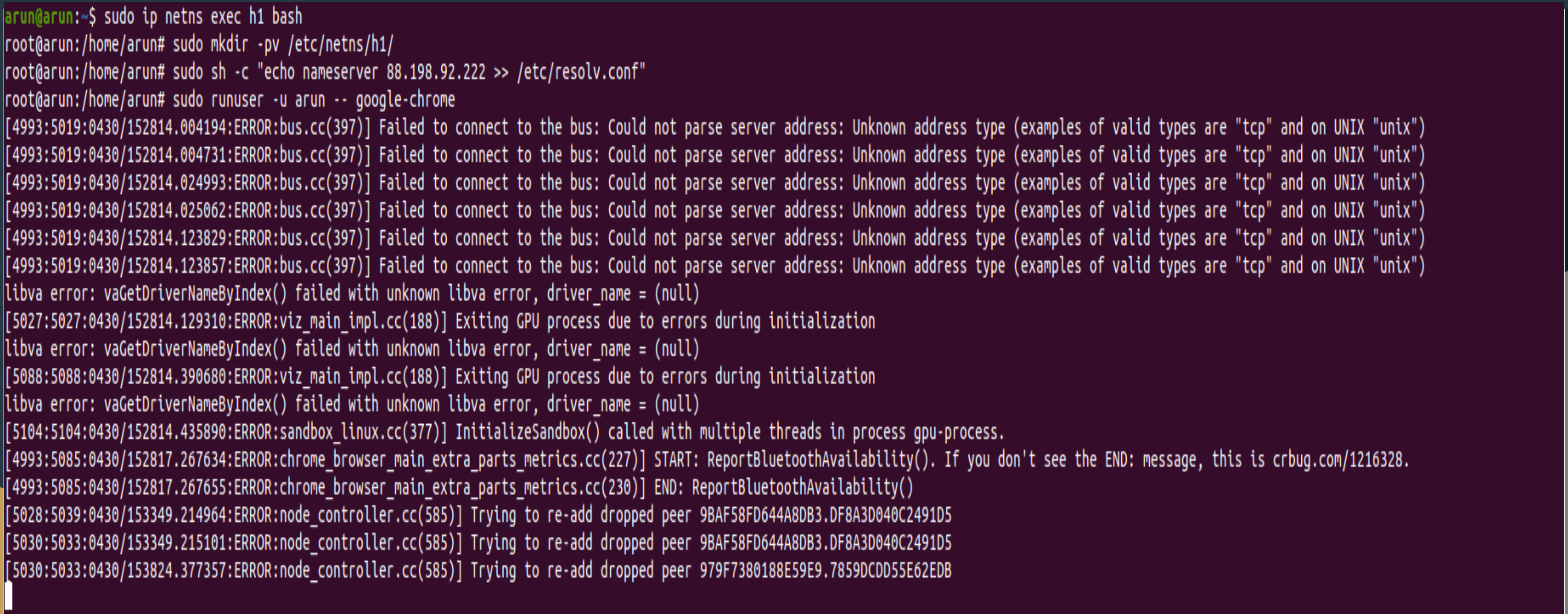
Open bash in Network Namespace : sudo ip netns exec <namespace_name > bash

Run Applications in Network Namespace Bash :

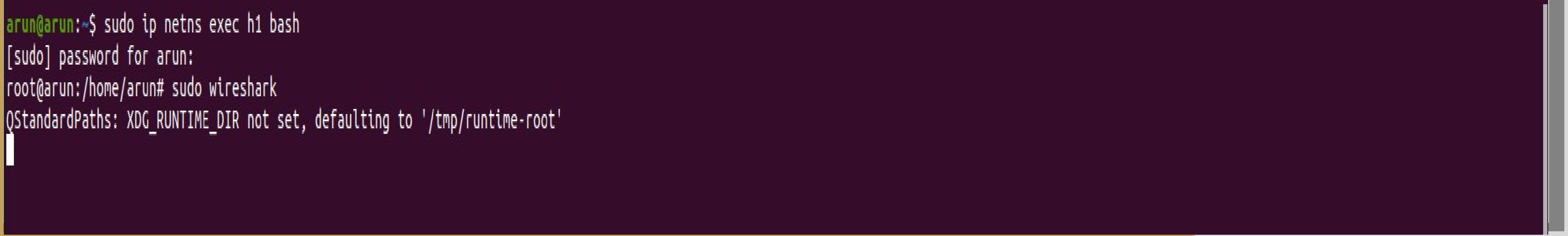
```
Sudo runuser -u <user_name> – <application_name>
```



```
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

vport1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet6 fe80::ec52:b2ff:fe2b:6e0e  prefixlen 64  scopeid 0x20<link>
        ether ee:52:b2:2b:6e:0e  txqueuelen 1000  (Ethernet)
        RX packets 11  bytes 866 (866.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 61  bytes 6804 (6.8 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

vport2: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet6 fe80::4008:47ff:fed3:2991  prefixlen 64  scopeid 0x20<link>
        ether 42:08:47:d3:29:91  txqueuelen 1000  (Ethernet)
        RX packets 11  bytes 866 (866.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 53  bytes 6148 (6.1 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

arun@arun:~$ sudo ovs-vsctl add-port mybridge enp0s3
arun@arun:~$ sudo ifconfig enp0s3 0
arun@arun:~$
arun@arun:~$ sudo dhclient mybridge
cmp: EOF on /tmp/tmp.sQMmpGH8mx which is empty
arun@arun:~$ sudo route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         10.0.2.2        0.0.0.0         UG    0      0        0 mybridge
10.0.2.0        0.0.0.0         255.255.255.0   U     0      0        0 mybridge
arun@arun:~$
arun@arun:~$ sudo ip netns exec h1 dhclient h1-eth0
cmp: EOF on /tmp/tmp.o5NajVtghF which is empty
arun@arun:~$ sudo ip netns exec h2 dhclient h2-eth0
cmp: EOF on /tmp/tmp.Y68OrM2vDs which is empty
arun@arun:~$ sudo route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         10.0.2.2        0.0.0.0         UG    0      0        0 mybridge
10.0.2.0        0.0.0.0         255.255.255.0   U     0      0        0 mybridge
arun@arun:~$
arun@arun:~$
arun@arun:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=2 ttl=51 time=48.6 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=51 time=62.9 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=51 time=64.2 ms
^C
--- 8.8.8.8 ping statistics ---
4 packets transmitted, 3 received, 25% packet loss, time 3021ms
rtt min/avg/max/mdev = 48.600/58.563/64.163/7.063 ms
```

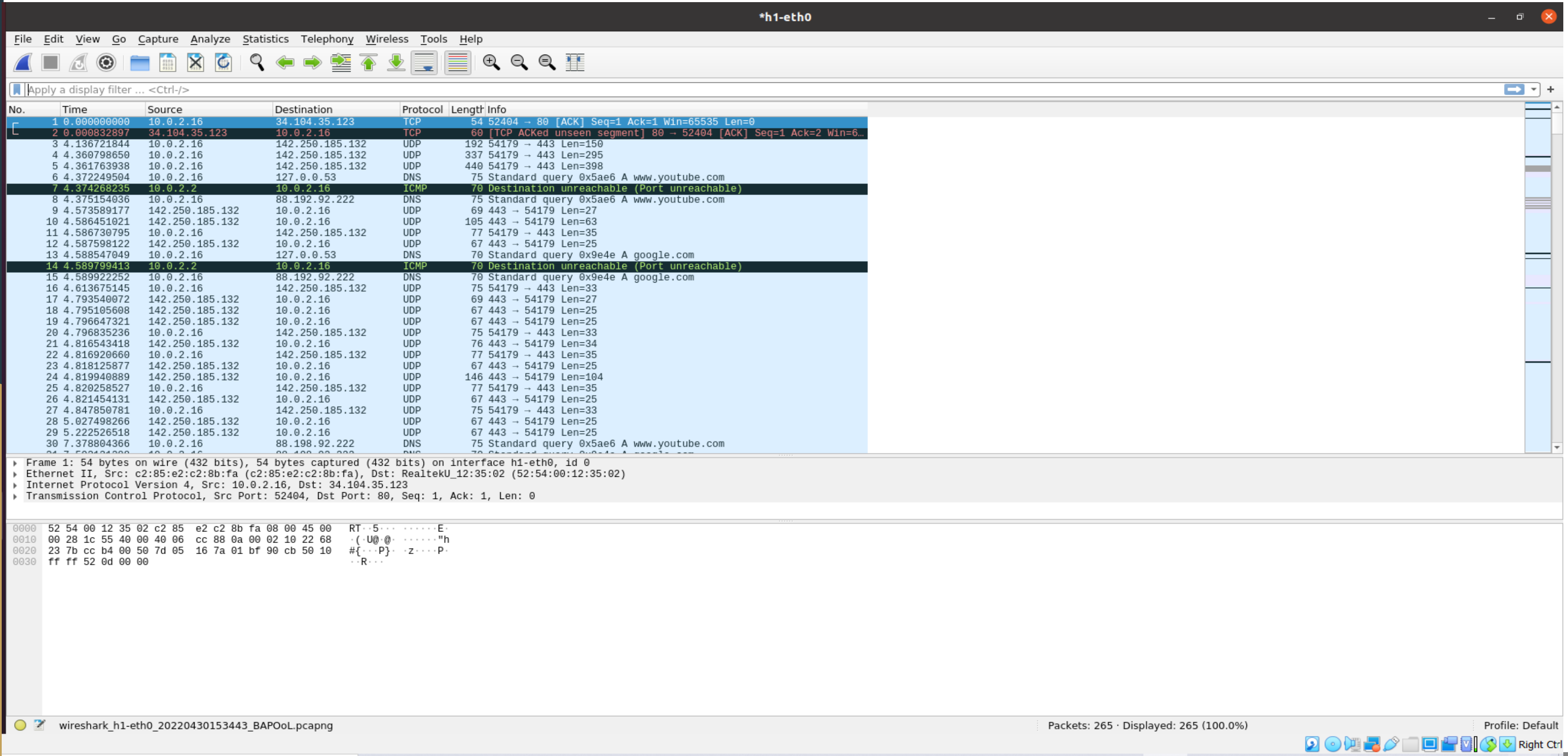


```
arun@arun:~$ sudo ip netns exec h1 bash
root@arun:/home/arun# sudo mkdir -pv /etc/netns/h1/
root@arun:/home/arun# sudo sh -c "echo nameserver 88.198.92.222 >> /etc/resolv.conf"
root@arun:/home/arun# sudo runuser -u arun -- google-chrome
[4993:5019:0430/152814.004194:ERROR:bus.cc(397)] Failed to connect to the bus: Could not parse server address: Unknown address type (examples of valid types are "tcp" and on UNIX "unix")
[4993:5019:0430/152814.004731:ERROR:bus.cc(397)] Failed to connect to the bus: Could not parse server address: Unknown address type (examples of valid types are "tcp" and on UNIX "unix")
[4993:5019:0430/152814.024993:ERROR:bus.cc(397)] Failed to connect to the bus: Could not parse server address: Unknown address type (examples of valid types are "tcp" and on UNIX "unix")
[4993:5019:0430/152814.025062:ERROR:bus.cc(397)] Failed to connect to the bus: Could not parse server address: Unknown address type (examples of valid types are "tcp" and on UNIX "unix")
[4993:5019:0430/152814.123829:ERROR:bus.cc(397)] Failed to connect to the bus: Could not parse server address: Unknown address type (examples of valid types are "tcp" and on UNIX "unix")
[4993:5019:0430/152814.123857:ERROR:bus.cc(397)] Failed to connect to the bus: Could not parse server address: Unknown address type (examples of valid types are "tcp" and on UNIX "unix")
libva error: vaGetDriverNameByIndex() failed with unknown libva error, driver_name = (null)
[5027:5027:0430/152814.129310:ERROR:viz_main_impl.cc(188)] Exiting GPU process due to errors during initialization
libva error: vaGetDriverNameByIndex() failed with unknown libva error, driver_name = (null)
[5088:5088:0430/152814.390680:ERROR:viz_main_impl.cc(188)] Exiting GPU process due to errors during initialization
libva error: vaGetDriverNameByIndex() failed with unknown libva error, driver_name = (null)
[5104:5104:0430/152814.435890:ERROR:sandbox_linux.cc(377)] InitializeSandbox() called with multiple threads in process gpu-process.
[4993:5085:0430/152817.267634:ERROR:chrome_browser_main_extra_parts_metrics.cc(227)] START: ReportBluetoothAvailability(). If you don't see the END: message, this is crbug.com/1216328.
[4993:5085:0430/152817.267655:ERROR:chrome_browser_main_extra_parts_metrics.cc(230)] END: ReportBluetoothAvailability()
[5028:5039:0430/153349.214964:ERROR:node_controller.cc(585)] Trying to re-add dropped peer 9BAF58FD644A8DB3.DF8A3D040C2491D5
[5030:5033:0430/153349.215101:ERROR:node_controller.cc(585)] Trying to re-add dropped peer 9BAF58FD644A8DB3.DF8A3D040C2491D5
[5030:5033:0430/153824.377357:ERROR:node_controller.cc(585)] Trying to re-add dropped peer 979F7380188E59E9.7859DCDD55E62EDB
```



```
arun@arun:~$ sudo ip netns exec h1 bash
[sudo] password for arun:
root@arun:/home/arun# sudo wireshark
QStandardPaths: XDG_RUNTIME_DIR not set, defaulting to '/tmp/runtime-root'
```

Tracing packets in wireshark in Network Namespace:



```
No.   Time          Source            Destination       Protocol Length Info
    1 0.000000000   10.0.2.16         34.104.35.123     TCP       54 52404 → 80 [ACK] Seq=1 Ack=1 Win=65535 Len=0
    2 0.000832897   34.104.35.123     10.0.2.16         TCP       60 [TCP ACKed unseen segment] 80 → 52404 [ACK] Seq=1 Ack=2 Win=6…
    3 4.136721844   10.0.2.16         142.250.185.132   UDP      192 54179 → 443 Len=150
    4 4.360798650   10.0.2.16         142.250.185.132   UDP      337 54179 → 443 Len=295
    5 4.361763938   10.0.2.16         142.250.185.132   UDP      440 54179 → 443 Len=398
    6 4.372249504   10.0.2.16         127.0.0.53        DNS       75 Standard query 0x5ae6 A www.youtube.com
    7 4.374268235   10.0.2.2          10.0.2.16         ICMP      70 Destination unreachable (Port unreachable)
    8 4.375154036   10.0.2.16         88.192.92.222     DNS       75 Standard query 0x5ae6 A www.youtube.com
    9 4.573589177   142.250.185.132   10.0.2.16         UDP       69 443 → 54179 Len=27
   10 4.586451021   142.250.185.132   10.0.2.16         UDP      105 443 → 54179 Len=63
   11 4.586730795   10.0.2.16         142.250.185.132   UDP       77 54179 → 443 Len=35
   12 4.587598122   142.250.185.132   10.0.2.16         UDP       67 443 → 54179 Len=25
   13 4.588547049   10.0.2.16         127.0.0.53        DNS       70 Standard query 0x9e4e A google.com
   14 4.589799413   10.0.2.2          10.0.2.16         ICMP      70 Destination unreachable (Port unreachable)
   15 4.589922252   10.0.2.16         88.192.92.222     DNS       70 Standard query 0x9e4e A google.com
   16 4.613675145   10.0.2.16         142.250.185.132   UDP       75 54179 → 443 Len=33
   17 4.793540072   142.250.185.132   10.0.2.16         UDP       69 443 → 54179 Len=27
   18 4.795105608   142.250.185.132   10.0.2.16         UDP       67 443 → 54179 Len=25
   19 4.796647321   142.250.185.132   10.0.2.16         UDP       67 443 → 54179 Len=25
   20 4.796835236   10.0.2.16         142.250.185.132   UDP       75 54179 → 443 Len=33
   21 4.816543418   142.250.185.132   10.0.2.16         UDP       76 443 → 54179 Len=34
   22 4.816920660   10.0.2.16         142.250.185.132   UDP       77 54179 → 443 Len=35
   23 4.818125877   142.250.185.132   10.0.2.16         UDP       67 443 → 54179 Len=25
   24 4.819940889   142.250.185.132   10.0.2.16         UDP      146 443 → 54179 Len=104
   25 4.820258527   10.0.2.16         142.250.185.132   UDP       77 54179 → 443 Len=35
   26 4.821454131   142.250.185.132   10.0.2.16         UDP       67 443 → 54179 Len=25
   27 4.847850781   10.0.2.16         142.250.185.132   UDP       75 54179 → 443 Len=33
   28 5.027498266   142.250.185.132   10.0.2.16         UDP       67 443 → 54179 Len=25
   29 5.222526518   142.250.185.132   10.0.2.16         UDP       67 443 → 54179 Len=25
   30 7.378804366   10.0.2.16         88.198.92.222     DNS       75 Standard query 0x5ae6 A www.youtube.com
```

```
▸ Frame 1: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface h1-eth0, id 0
▸ Ethernet II, Src: c2:85:e2:c2:8b:fa (c2:85:e2:c2:8b:fa), Dst: RealtekU_12:35:02 (52:54:00:12:35:02)
▸ Internet Protocol Version 4, Src: 10.0.2.16, Dst: 34.104.35.123
▸ Transmission Control Protocol, Src Port: 52404, Dst Port: 80, Seq: 1, Ack: 1, Len: 0
```

```
0000  52 54 00 12 35 02 c2 85  e2 c2 8b fa 08 00 45 00   RT··5··· ······E·
0010  00 28 1c 55 40 00 40 06  cc 88 0a 00 02 10 22 68   ·(·U@·@· ······"h
0020  23 7b cc b4 00 50 7d 05  16 7a 01 bf 90 cb 50 10   #{···P}· ·z····P·
0030  ff ff 52 0d 00 00                                  ··R···
```

wireshark_h1-eth0_20220430153443_BAPOoL.pcapng

Packets: 265 · Displayed: 265 (100.0%)

Profile: Default

Connect Network Namespaces to internet using Mininet :

Create topology of consisting 1 switch and 2 hosts :

Sudo mn –topo=single,2

Enable/Up the switch : sudo ifconfig <switch_name> up

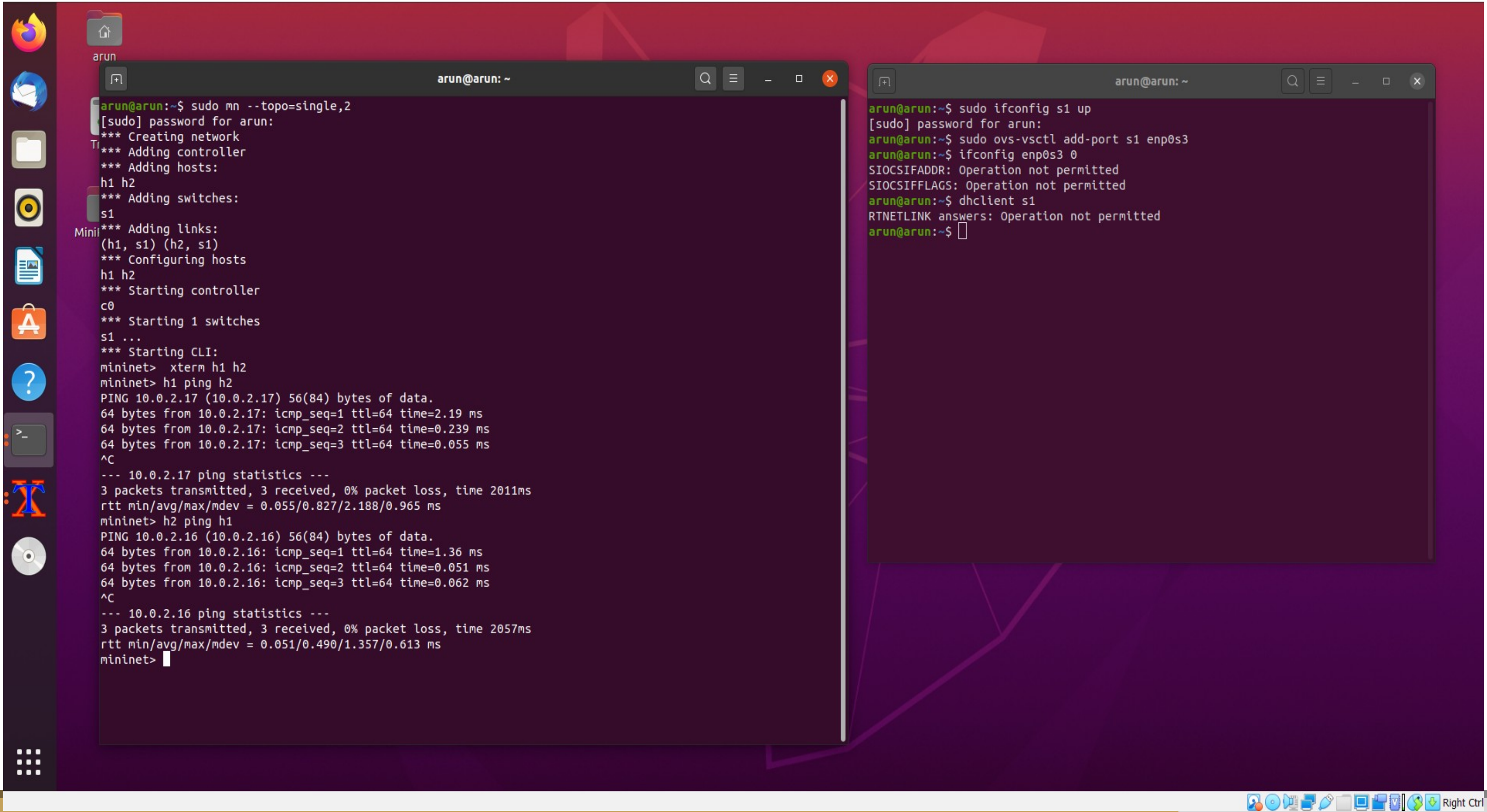
Assign IP Address to Switch : dhclient <switch_name>

Open the terminal for for host in mininet : xterm <host_name>

Assign IP Address to hosts : host_name> dhclient <interface_name>

Run Wireshark on Host : sudo wireshark

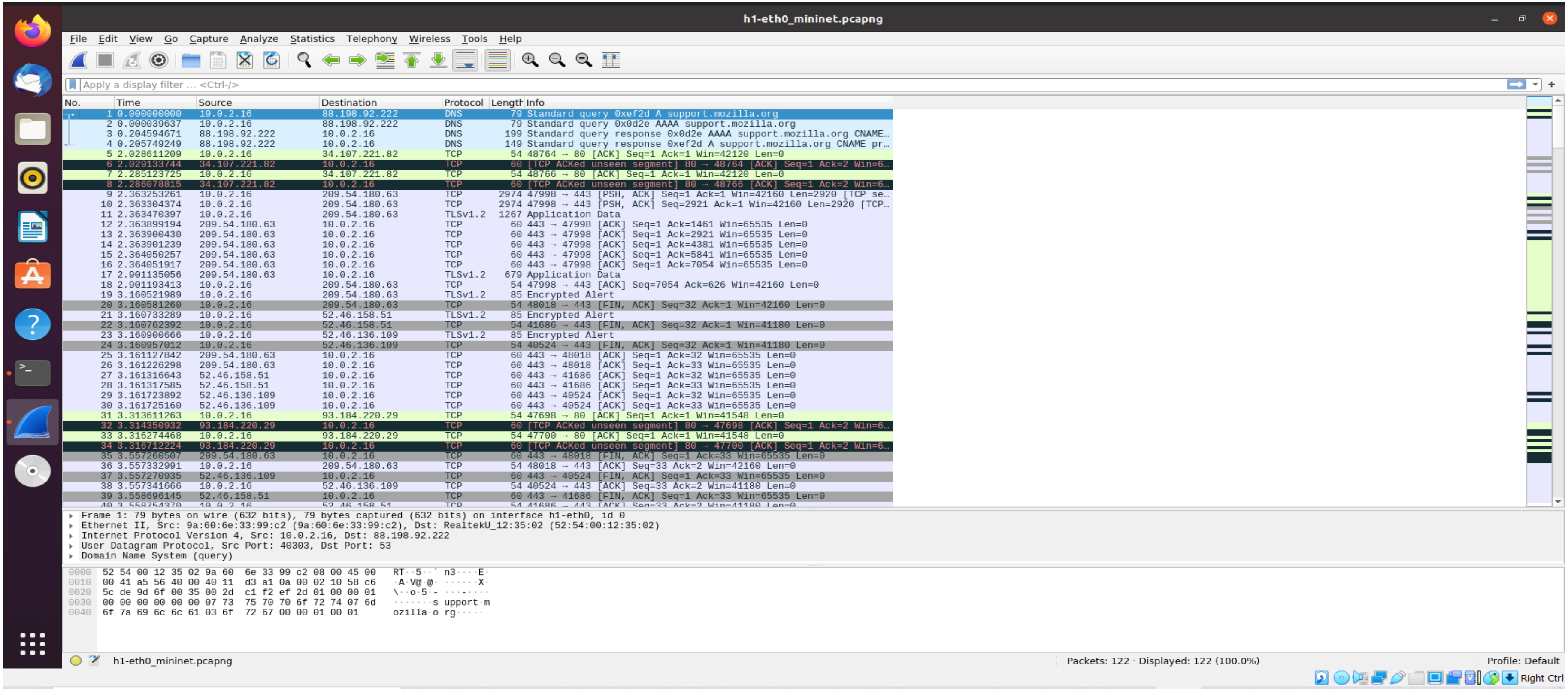
Run Applications in host : runuser -u <user_name> – <application_name>



```
arun@arun:~$ sudo mn --topo=single,2
[sudo] password for arun:
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2
*** Adding switches:
s1
*** Adding links:
(h1, s1) (h2, s1)
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Starting CLI:
mininet>  xterm h1 h2
mininet> h1 ping h2
PING 10.0.2.17 (10.0.2.17) 56(84) bytes of data.
64 bytes from 10.0.2.17: icmp_seq=1 ttl=64 time=2.19 ms
64 bytes from 10.0.2.17: icmp_seq=2 ttl=64 time=0.239 ms
64 bytes from 10.0.2.17: icmp_seq=3 ttl=64 time=0.055 ms
^C
--- 10.0.2.17 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2011ms
rtt min/avg/max/mdev = 0.055/0.827/2.188/0.965 ms
mininet> h2 ping h1
PING 10.0.2.16 (10.0.2.16) 56(84) bytes of data.
64 bytes from 10.0.2.16: icmp_seq=1 ttl=64 time=1.36 ms
64 bytes from 10.0.2.16: icmp_seq=2 ttl=64 time=0.051 ms
64 bytes from 10.0.2.16: icmp_seq=3 ttl=64 time=0.062 ms
^C
--- 10.0.2.16 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2057ms
rtt min/avg/max/mdev = 0.051/0.490/1.357/0.613 ms
mininet> 
```

```
arun@arun:~$ sudo ifconfig s1 up
[sudo] password for arun:
arun@arun:~$ sudo ovs-vsctl add-port s1 enp0s3
arun@arun:~$ ifconfig enp0s3 0
SIOCSIFADDR: Operation not permitted
SIOCSIFFLAGS: Operation not permitted
arun@arun:~$ dhclient s1
RTNETLINK answers: Operation not permitted
arun@arun:~$ 
```

Tracing Packets in Wireshark on host in Mininet :



```
No.  Time          Source          Destination     Protocol Length Info
   1 0.000000000   10.0.2.16       88.198.92.222   DNS        79 Standard query 0xef2d A support.mozilla.org
   2 0.000039637   10.0.2.16       88.198.92.222   DNS        79 Standard query 0x0d2e AAAA support.mozilla.org
   3 0.204594671   88.198.92.222   10.0.2.16       DNS       199 Standard query response 0x0d2e AAAA support.mozilla.org CNAME…
   4 0.205749249   88.198.92.222   10.0.2.16       DNS       149 Standard query response 0xef2d A support.mozilla.org CNAME pr…
   5 2.028611209   10.0.2.16       34.107.221.82   TCP        54 48764 → 80 [ACK] Seq=1 Ack=1 Win=42120 Len=0
   6 2.029133744   34.107.221.82   10.0.2.16       TCP        60 [TCP ACKed unseen segment] 80 → 48764 [ACK] Seq=1 Ack=2 Win=6…
   7 2.285123725   10.0.2.16       34.107.221.82   TCP        54 48766 → 80 [ACK] Seq=1 Ack=1 Win=42120 Len=0
   8 2.286078815   34.107.221.82   10.0.2.16       TCP        60 [TCP ACKed unseen segment] 80 → 48766 [ACK] Seq=1 Ack=2 Win=6…
   9 2.363253261   10.0.2.16       209.54.180.63   TCP      2974 47998 → 443 [PSH, ACK] Seq=1 Ack=1 Win=42160 Len=2920 [TCP se…
  10 2.363304374   10.0.2.16       209.54.180.63   TCP      2974 47998 → 443 [PSH, ACK] Seq=2921 Ack=1 Win=42160 Len=2920 [TCP…
  11 2.363470397   10.0.2.16       209.54.180.63   TLSv1.2  1267 Application Data
  12 2.363899194   209.54.180.63   10.0.2.16       TCP        60 443 → 47998 [ACK] Seq=1 Ack=1461 Win=65535 Len=0
  13 2.363900430   209.54.180.63   10.0.2.16       TCP        60 443 → 47998 [ACK] Seq=1 Ack=2921 Win=65535 Len=0
  14 2.363901239   209.54.180.63   10.0.2.16       TCP        60 443 → 47998 [ACK] Seq=1 Ack=4381 Win=65535 Len=0
  15 2.364050257   209.54.180.63   10.0.2.16       TCP        60 443 → 47998 [ACK] Seq=1 Ack=5841 Win=65535 Len=0
  16 2.364051917   209.54.180.63   10.0.2.16       TCP        60 443 → 47998 [ACK] Seq=1 Ack=7054 Win=65535 Len=0
  17 2.901135056   209.54.180.63   10.0.2.16       TLSv1.2   679 Application Data
  18 2.901193413   10.0.2.16       209.54.180.63   TCP        54 47998 → 443 [ACK] Seq=7054 Ack=626 Win=42160 Len=0
  19 3.160521989   10.0.2.16       209.54.180.63   TLSv1.2    85 Encrypted Alert
  20 3.160581260   10.0.2.16       209.54.180.63   TCP        54 48018 → 443 [FIN, ACK] Seq=32 Ack=1 Win=42160 Len=0
  21 3.160733289   10.0.2.16       52.46.158.51    TLSv1.2    85 Encrypted Alert
  22 3.160762392   10.0.2.16       52.46.158.51    TCP        54 41686 → 443 [FIN, ACK] Seq=32 Ack=1 Win=41180 Len=0
  23 3.160900666   10.0.2.16       52.46.136.109   TLSv1.2    85 Encrypted Alert
  24 3.160957012   10.0.2.16       52.46.136.109   TCP        54 40524 → 443 [FIN, ACK] Seq=32 Ack=1 Win=41180 Len=0
  25 3.161127842   209.54.180.63   10.0.2.16       TCP        60 443 → 48018 [ACK] Seq=1 Ack=32 Win=65535 Len=0
  26 3.161226298   209.54.180.63   10.0.2.16       TCP        60 443 → 48018 [ACK] Seq=1 Ack=33 Win=65535 Len=0
  27 3.161316643   52.46.158.51    10.0.2.16       TCP        60 443 → 41686 [ACK] Seq=1 Ack=32 Win=65535 Len=0
  28 3.161317585   52.46.158.51    10.0.2.16       TCP        60 443 → 41686 [ACK] Seq=1 Ack=33 Win=65535 Len=0
  29 3.161723892   52.46.136.109   10.0.2.16       TCP        60 443 → 40524 [ACK] Seq=1 Ack=32 Win=65535 Len=0
  30 3.161725160   52.46.136.109   10.0.2.16       TCP        60 443 → 40524 [ACK] Seq=1 Ack=33 Win=65535 Len=0
  31 3.313611263   10.0.2.16       93.184.220.29   TCP        54 47698 → 80 [ACK] Seq=1 Ack=1 Win=41548 Len=0
  32 3.314350932   93.184.220.29   10.0.2.16       TCP        60 [TCP ACKed unseen segment] 80 → 47698 [ACK] Seq=1 Ack=2 Win=6…
  33 3.316274468   10.0.2.16       93.184.220.29   TCP        54 47700 → 80 [ACK] Seq=1 Ack=1 Win=41548 Len=0
  34 3.316712224   93.184.220.29   10.0.2.16       TCP        60 [TCP ACKed unseen segment] 80 → 47700 [ACK] Seq=1 Ack=2 Win=6…
  35 3.557260507   209.54.180.63   10.0.2.16       TCP        60 443 → 48018 [FIN, ACK] Seq=1 Ack=33 Win=65535 Len=0
  36 3.557332991   10.0.2.16       209.54.180.63   TCP        54 48018 → 443 [ACK] Seq=33 Ack=2 Win=42160 Len=0
  37 3.557270935   52.46.136.109   10.0.2.16       TCP        60 443 → 40524 [FIN, ACK] Seq=1 Ack=33 Win=65535 Len=0
  38 3.557341666   10.0.2.16       52.46.136.109   TCP        54 40524 → 443 [ACK] Seq=33 Ack=2 Win=41180 Len=0
  39 3.558696145   52.46.158.51    10.0.2.16       TCP        60 443 → 41686 [FIN, ACK] Seq=1 Ack=33 Win=65535 Len=0
  40 3.558754370   10.0.2.16       52.46.158.51    TCP        54 41686 → 443 [ACK] Seq=33 Ack=2 Win=41180 Len=0
```

```
▸ Frame 1: 79 bytes on wire (632 bits), 79 bytes captured (632 bits) on interface h1-eth0, id 0
▸ Ethernet II, Src: 9a:60:6e:33:99:c2 (9a:60:6e:33:99:c2), Dst: RealtekU_12:35:02 (52:54:00:12:35:02)
▸ Internet Protocol Version 4, Src: 10.0.2.16, Dst: 88.198.92.222
▸ User Datagram Protocol, Src Port: 40303, Dst Port: 53
▸ Domain Name System (query)
```

```
0000  52 54 00 12 35 02 9a 60  6e 33 99 c2 08 00 45 00   RT··5··` n3····E·
0010  00 41 a5 56 40 00 40 11  d3 a1 0a 00 02 10 58 c6   ·A·V@·@· ······X·
0020  5c de 9d 6f 00 35 00 2d  c1 f2 ef 2d 01 00 00 01   \··o·5·- ···-····
0030  00 00 00 00 00 00 07 73  75 70 70 6f 72 74 07 6d   ·······s upport·m
0040  6f 7a 69 6c 6c 61 03 6f  72 67 00 00 01 00 01      ozilla·o rg·····
```

h1-eth0_mininet.pcapng — Packets: 122 · Displayed: 122 (100.0%) — Profile: Default

Create a network namespace and run a webserver on namespace :

Create A Network Namespace : sudo ip netns add <namespace_name>

Execute commands in Network Namespace :

sudo ip netns exec <namespace_name> your-command

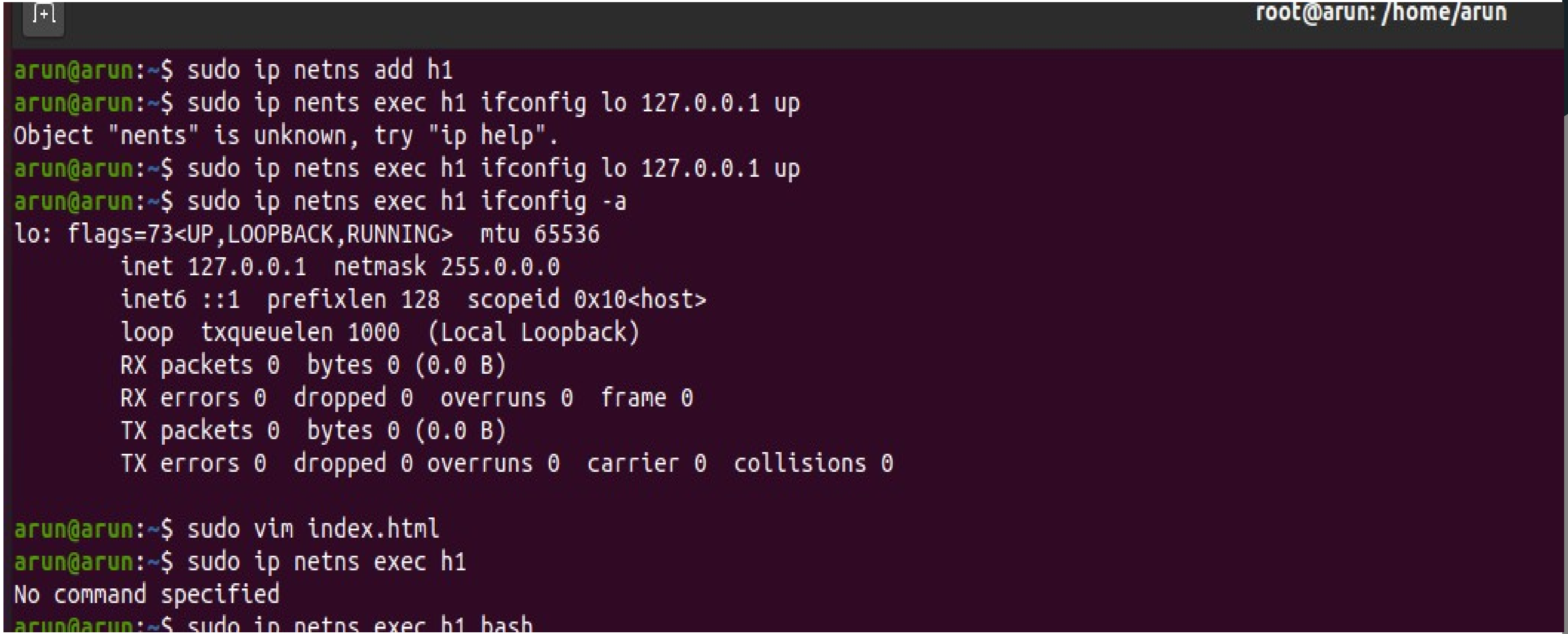
Set the IP Address to the Default Interface “lo” in the Namespace :

sudo ip netns exec <namespace_name> ifconfig lo <IP_address> up

See the Network Interfaces on Network Namespace :

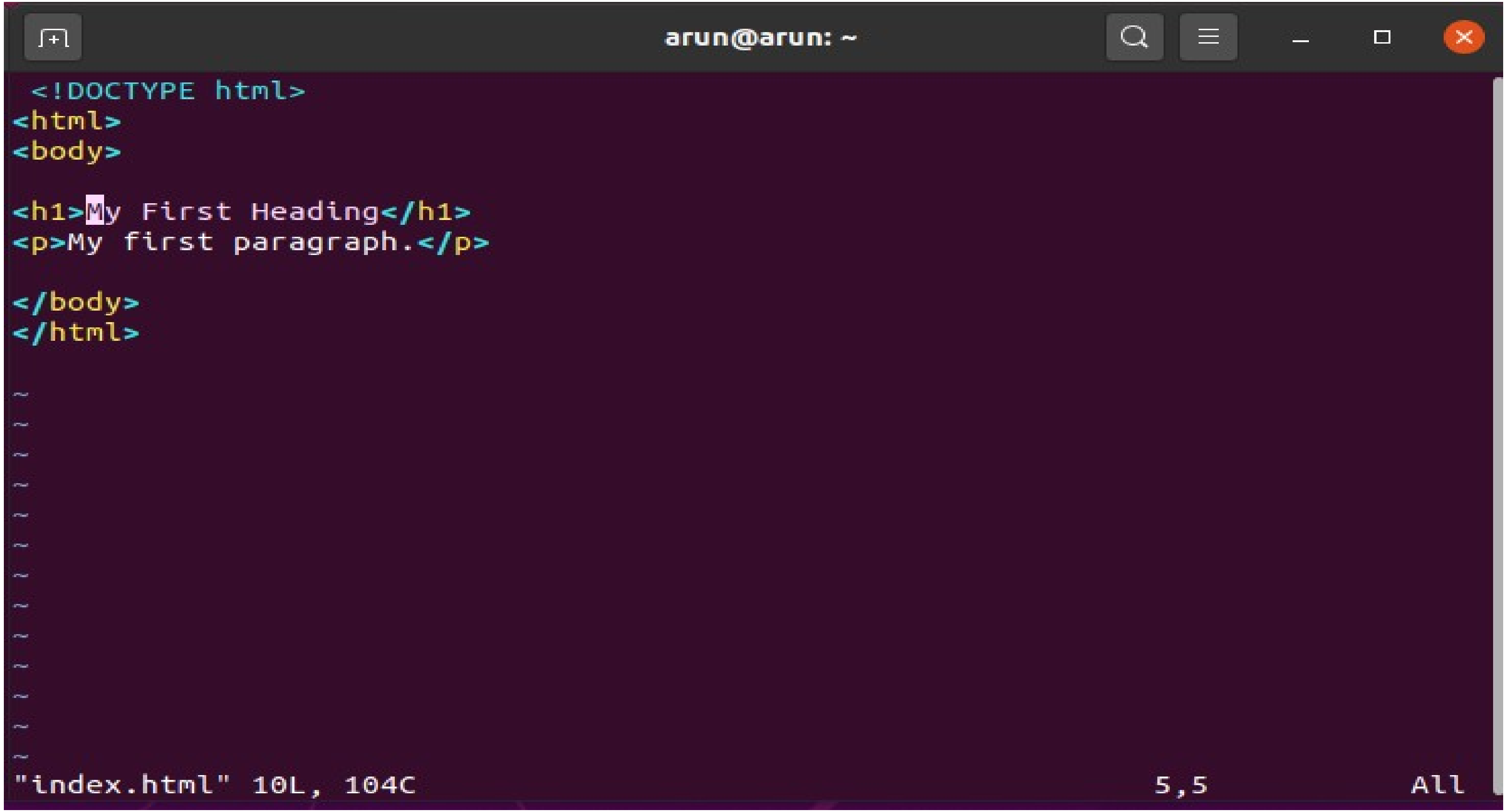
Sudo ip netns exec <namespace_name> ifconfig -a

Implementation :



```
arun@arun:~$ sudo ip netns add h1
arun@arun:~$ sudo ip nents exec h1 ifconfig lo 127.0.0.1 up
Object "nents" is unknown, try "ip help".
arun@arun:~$ sudo ip netns exec h1 ifconfig lo 127.0.0.1 up
arun@arun:~$ sudo ip netns exec h1 ifconfig -a
lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 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

arun@arun:~$ sudo vim index.html
arun@arun:~$ sudo ip netns exec h1
No command specified
arun@arun:~$ sudo ip netns exec h1 bash
```

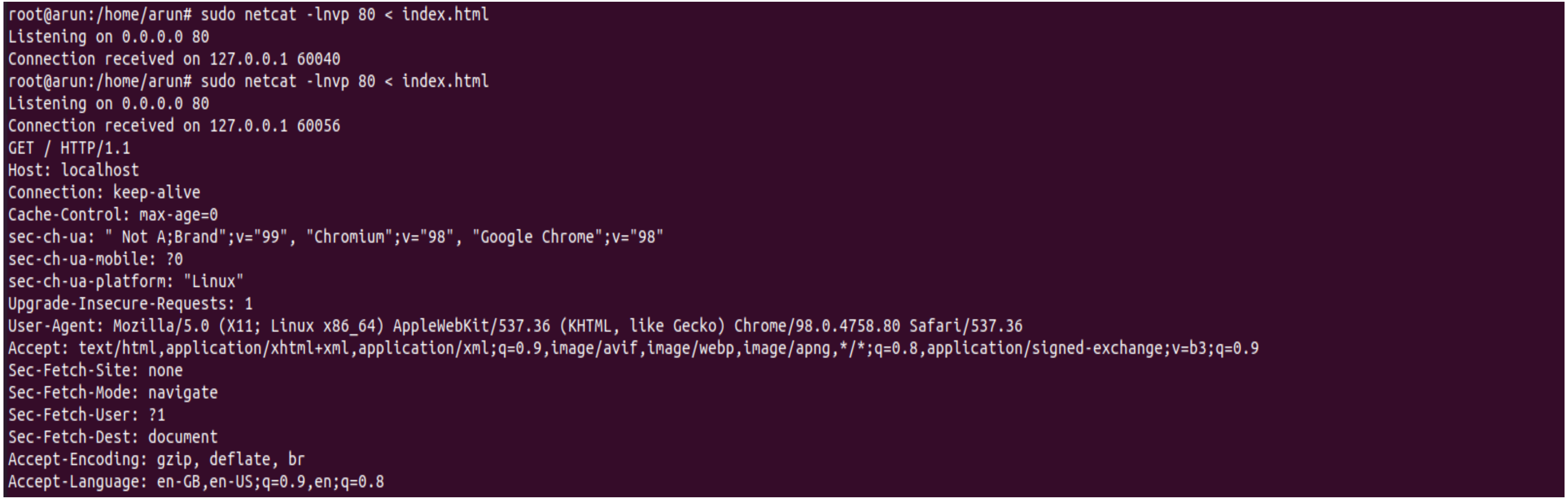
Create a web page (index.html) :



```
 <!DOCTYPE html>
<html>
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>


~
~
~
~
~
~
~
~
~
~
~
~
~
"index.html" 10L, 104C                                  5,5           All
```

Run a webserver on port 80 , in namespace using natcat command :

Sudo netcat -lnvp 80 < index.html

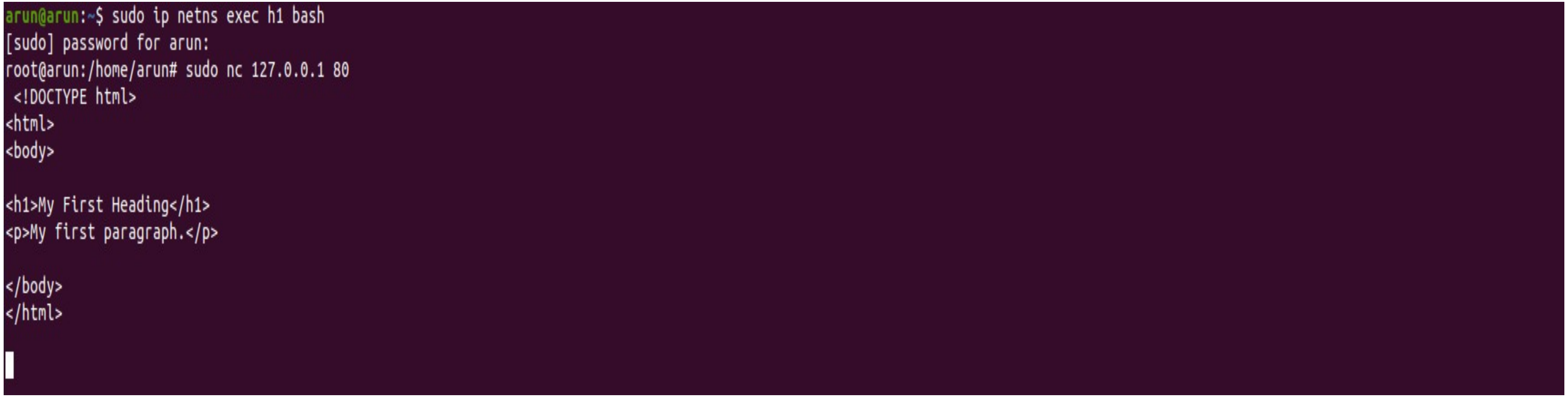


```
root@arun:/home/arun# sudo netcat -lnvp 80 < index.html
Listening on 0.0.0.0 80
Connection received on 127.0.0.1 60040
root@arun:/home/arun# sudo netcat -lnvp 80 < index.html
Listening on 0.0.0.0 80
Connection received on 127.0.0.1 60056
GET / HTTP/1.1
Host: localhost
Connection: keep-alive
Cache-Control: max-age=0
sec-ch-ua: " Not A;Brand";v="99", "Chromium";v="98", "Google Chrome";v="98"
sec-ch-ua-mobile: ?0
sec-ch-ua-platform: "Linux"
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/98.0.4758.80 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Sec-Fetch-Site: none
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Accept-Encoding: gzip, deflate, br
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
```

Access webserver from namespace :

Open namespace : sudo ip netns exec <namespace_name> bash

Access webserver in namespace : sudo netncat <IP_Address> <Port_No(80)>



```
arun@arun:~$ sudo ip netns exec h1 bash
[sudo] password for arun:
root@arun:/home/arun# sudo nc 127.0.0.1 80
 <!DOCTYPE html>
<html>
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>
```

Conclusion :

Network virtualization describes the combination of one or more platforms to form a virtual network.These virtual networks can act as an independent entity , enabling us to emulate links between services and applications. Network virtualization is possible through links and nodes. We can create our own virtual networks based on our requirements .

References :

- http://sdnflex.org/?page_id=2
- https://ivypanda.com/essays/concept-of-the-network-virtualization/#:~:text=.%2C%202013).-,Conclusion,links%20between%20services%20and%20applications.
- https://www.cloud-native-everything.com/network-namespaces-in-5-min/
- https://fosshelp.blogspot.com/2014/07/how-to-create-network-namespace-and-run.html
- https://gist.github.com/shreyakupadhyay/84dc75607ec1078aca3129c8958f3683