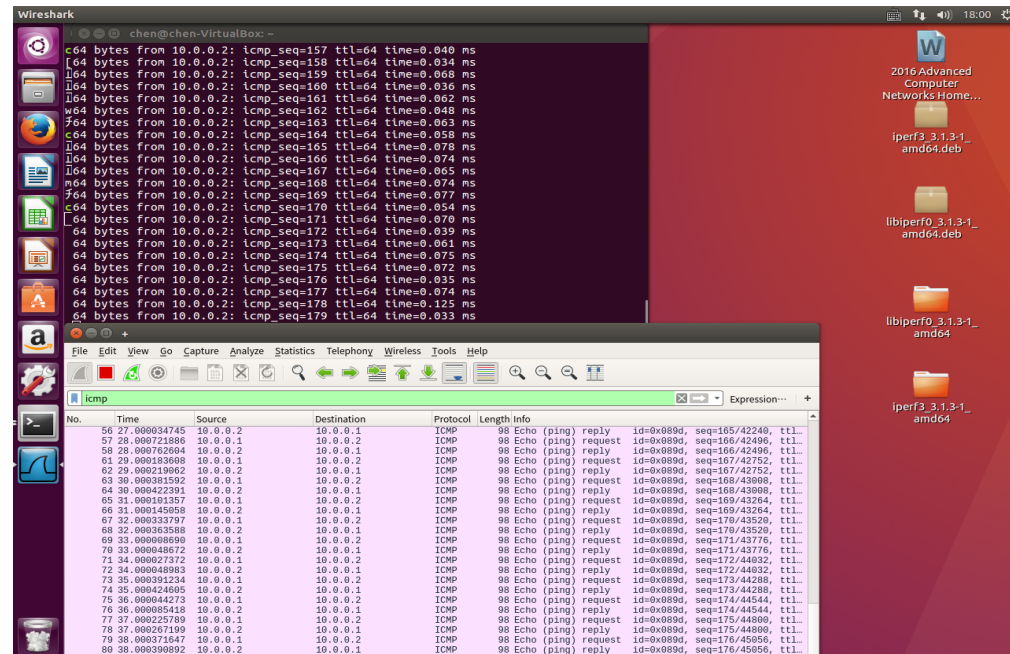


Part1:Mininet

apt-get install mininet

Sudo mn

指令: Mininet : h1 ping h2



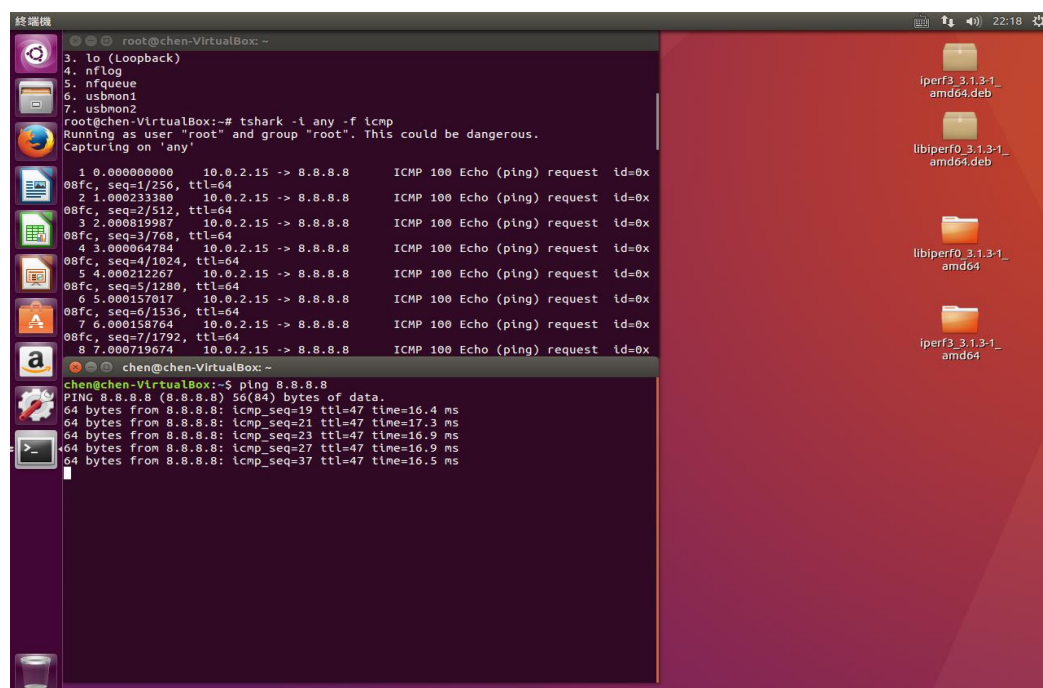
Part2:

1. tshark

請使用 tshark 指令搭配正確的參數達到以下的要求：

- 抓取” icmp” 的封包，並且來源或目的是” 8.8.8.8”

指令: tshark -i any -f icmp



- b. 將擷取的封包儲存為一個檔案，名稱為” packet01”

指令: `tshark -w packet01`

```
終端機
root@chen-VirtualBox: ~
11 5.008340183 10.0.2.15 -> 8.8.8.8 ICMP 100 Echo (ping) request id=0x
09c9, seq=6/1536, ttl=64
12 5.025499341 8.8.8.8 -> 10.0.2.15 ICMP 100 Echo (ping) reply id=0x
09c9, seq=6/1536, ttl=47 (request in 11)
13 6.009620110 10.0.2.15 -> 8.8.8.8 ICMP 100 Echo (ping) request id=0x
09c9, seq=7/1792, ttl=64
14 6.025835546 8.8.8.8 -> 10.0.2.15 ICMP 100 Echo (ping) reply id=0x
09c9, seq=7/1792, ttl=47 (request in 13)
15 7.011030662 10.0.2.15 -> 8.8.8.8 ICMP 100 Echo (ping) request id=0x
09c9, seq=8/2048, ttl=64
16 7.027290311 8.8.8.8 -> 10.0.2.15 ICMP 100 Echo (ping) reply id=0x
09c9, seq=8/2048, ttl=47 (request in 15)
17 8.013054709 10.0.2.15 -> 8.8.8.8 ICMP 100 Echo (ping) request id=0x
09c9, seq=9/2304, ttl=64
18 8.029307679 8.8.8.8 -> 10.0.2.15 ICMP 100 Echo (ping) reply id=0x
09c9, seq=9/2304, ttl=47 (request in 17)
^C18 packets captured
root@chen-VirtualBox:~# tshark -w packet01>
-bash: syntax error near unexpected token 'newline'
root@chen-VirtualBox:~# tshark -w packet01
Running as user "root" and group "root". This could be dangerous.
Capturing on 'enp0s3'
26 ^C
root@chen-VirtualBox:~# tshark -r packet01
chen@chen-VirtualBox:~
chen@chen-VirtualBox:~$ 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=1 ttl=47 time=34.1 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=47 time=16.3 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=47 time=16.2 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=47 time=16.0 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=47 time=16.3 ms
64 bytes from 8.8.8.8: icmp_seq=6 ttl=47 time=17.1 ms
64 bytes from 8.8.8.8: icmp_seq=7 ttl=47 time=16.2 ms
64 bytes from 8.8.8.8: icmp_seq=8 ttl=47 time=16.2 ms
64 bytes from 8.8.8.8: icmp_seq=9 ttl=47 time=16.2 ms
^C
--- 8.8.8.8 ping statistics ---
9 packets transmitted, 9 received, 0% packet loss, time 8013ms
rtt min/avg/max/ndev = 16.047/18.350/34.148/5.595 ms
chen@chen-VirtualBox:~$ 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=1 ttl=47 time=15.6 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=47 time=16.3 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=47 time=16.2 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=47 time=16.1 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=47 time=16.3 ms
64 bytes from 8.8.8.8: icmp_seq=6 ttl=47 time=16.3 ms
64 bytes from 8.8.8.8: icmp_seq=7 ttl=47 time=16.0 ms
```

- c. 下達正確的指令後，請 ping 8.8.8.8 及 ping 208.67.220.220
ping 8.8.8.8 及 ping 208.67.220.220

```
終端機
root@chen-VirtualBox: ~
chen@chen-VirtualBox:~$ sudo -i
[sudo] password for chen:
root@chen-VirtualBox:~# tshark -w packet01
Running as user "root" and group "root". This could be dangerous.
Capturing on 'enp0s3'
44 ^C

chen@chen-VirtualBox:~
chen@chen-VirtualBox:~$ 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=1 ttl=47 time=16.64 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=47 time=16.64 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=47 time=17.64 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=47 time=16.64 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=47 time=16.64 ms
64 bytes from 8.8.8.8: icmp_seq=6 ttl=47 time=16.64 ms
64 bytes from 8.8.8.8: icmp_seq=7 ttl=47 time=16.64 ms
64 bytes from 8.8.8.8: icmp_seq=8 ttl=47 time=16.64 ms
64 bytes from 8.8.8.8: icmp_seq=9 ttl=47 time=16.64 ms
64 bytes from 8.8.8.8: icmp_seq=10 ttl=47 time=16.64 ms
64 bytes from 8.8.8.8: icmp_seq=11 ttl=47 time=16.64 ms
64 bytes from 8.8.8.8: icmp_seq=12 ttl=47 time=16.64 ms
64 bytes from 8.8.8.8: icmp_seq=13 ttl=47 time=16.64 ms
^C

chen@chen-VirtualBox:~$ ping 208.67.220.220
PING 208.67.220.220 (208.67.220.220) 56(84) bytes of data.
64 bytes from 208.67.220.220: icmp_seq=1 ttl=54 time=93.6 ms
64 bytes from 208.67.220.220: icmp_seq=2 ttl=54 time=94.0 ms
64 bytes from 208.67.220.220: icmp_seq=3 ttl=54 time=93.8 ms
64 bytes from 208.67.220.220: icmp_seq=4 ttl=54 time=93.8 ms
64 bytes from 208.67.220.220: icmp_seq=5 ttl=54 time=93.7 ms
64 bytes from 208.67.220.220: icmp_seq=6 ttl=54 time=94.1 ms
64 bytes from 208.67.220.220: icmp_seq=7 ttl=54 time=93.6 ms
64 bytes from 208.67.220.220: icmp_seq=8 ttl=54 time=93.9 ms
64 bytes from 208.67.220.220: icmp_seq=9 ttl=54 time=93.9 ms
64 bytes from 208.67.220.220: icmp_seq=10 ttl=54 time=93.6 ms
```

- d. 結束後使用 `tshark` 指令查看擷取的檔案
指令: `tshark -r packet01`

```
chen@chen-VirtualBox:~$ sudo -i
[sudo] password for chen:
root@chen-VirtualBox:~# tshark -w packet01
Running as user "root" and group "root". This could be dangerous.
Capturing on 'enp0s3'
174 ^C
root@chen-VirtualBox:~# tshark -r packet01
Running as user "root" and group "root". This could be dangerous.
tshark: The file "packet01" doesn't exist.
root@chen-VirtualBox:~# tshark -r packet01
Running as user "root" and group "root". This could be dangerous.
1 0.000000000 10.0.2.15 -> 8.8.8.8 ICMP 98 Echo (ping) request id=0x0
ab0, seq=1/256, ttl=64
2 0.016143100 8.8.8.8 -> 10.0.2.15 ICMP 98 Echo (ping) reply id=0x0
ab0, seq=1/256, ttl=47 (request in 1)
3 1.001328568 10.0.2.15 -> 8.8.8.8 ICMP 98 Echo (ping) request id=0x0
ab0, seq=2/512, ttl=64
4 1.018062164 8.8.8.8 -> 10.0.2.15 ICMP 98 Echo (ping) reply id=0x0
ab0, seq=2/512, ttl=47 (request in 3)
5 2.003665628 10.0.2.15 -> 8.8.8.8 ICMP 98 Echo (ping) request id=0x0
ab0, seq=3/768, ttl=64
6 2.020436009 8.8.8.8 -> 10.0.2.15 ICMP 98 Echo (ping) reply id=0x0
ab0, seq=3/768, ttl=47 (request in 5)

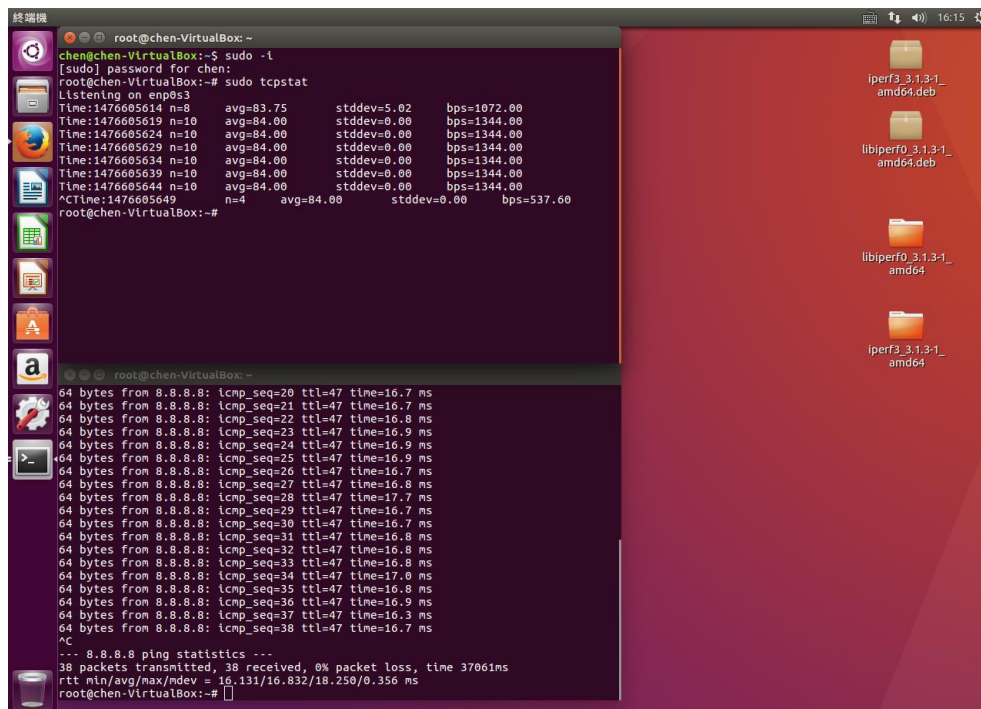
4 bytes from 8.8.8.8: icmp_seq=15 ttl=47 time=16.7 ms
4 bytes from 8.8.8.8: icmp_seq=16 ttl=47 time=16.8 ms
4 bytes from 8.8.8.8: icmp_seq=17 ttl=47 time=15.9 ms
4 bytes from 8.8.8.8: icmp_seq=18 ttl=47 time=16.4 ms
4 bytes from 8.8.8.8: icmp_seq=19 ttl=47 time=15.9 ms
4 bytes from 8.8.8.8: icmp_seq=20 ttl=47 time=16.6 ms
4 bytes from 8.8.8.8: icmp_seq=21 ttl=47 time=16.8 ms
4 bytes from 8.8.8.8: icmp_seq=22 ttl=47 time=16.8 ms
4 bytes from 8.8.8.8: icmp_seq=23 ttl=47 time=17.1 ms
4 bytes from 8.8.8.8: icmp_seq=24 ttl=47 time=16.7 ms
4 bytes from 8.8.8.8: icmp_seq=25 ttl=47 time=16.8 ms
4 bytes from 8.8.8.8: icmp_seq=26 ttl=47 time=16.4 ms
4 bytes from 8.8.8.8: icmp_seq=27 ttl=47 time=15.9 ms
4 bytes from 8.8.8.8: icmp_seq=28 ttl=47 time=16.0 ms
4 bytes from 8.8.8.8: icmp_seq=29 ttl=47 time=16.7 ms
4 bytes from 8.8.8.8: icmp_seq=30 ttl=47 time=16.3 ms
4 bytes from 8.8.8.8: icmp_seq=31 ttl=47 time=16.0 ms
4 bytes from 8.8.8.8: icmp_seq=32 ttl=47 time=16.7 ms
4 bytes from 8.8.8.8: icmp_seq=33 ttl=47 time=16.7 ms
C
-- 8.8.8.8 ping statistics --
33 packets transmitted, 33 received, 0% packet loss, time 32048ms
rtt min/avg/max/mdev = 15.913/16.626/17.165/0.363 ms
hen@chen-VirtualBox:~$
```

- e. 以上請寫出正確的指令並截圖證明

2. tcpstat

請使用 `tcpstat` 指令搭配正確的參數達到以下的要求：

- 抓取"icmp"的封包
`Sudo tcpstat`
- ping 任意位址
`ping 8.8.8.8`
- 完成後中斷 `tcpstat`，將顯示的結果截圖，並寫出正確的指令



3. tcpdump & tcpstat & gnuplot

本題要讓同學練習使用以上三個工具，將網路流量監測的結果繪製成圖表，繳交作業時請一併附上繪製出的圖檔，請依照下列步驟操作。

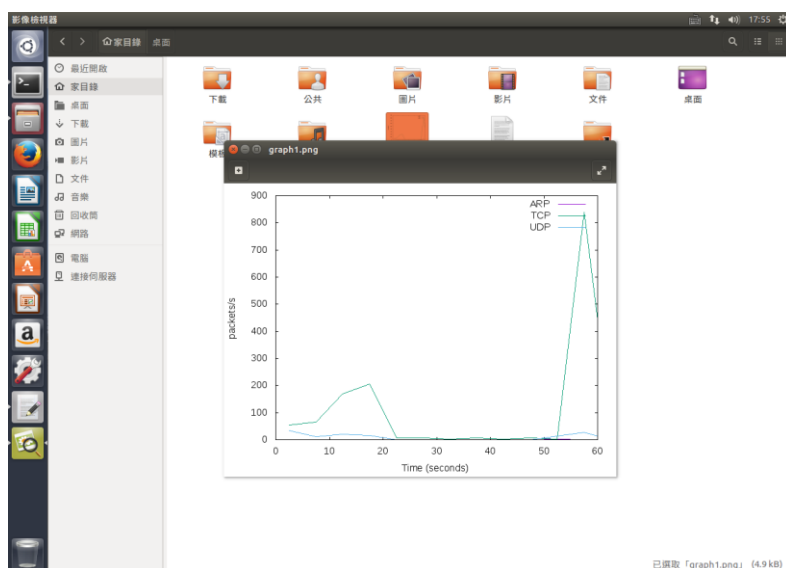
- 使用 `tcpdump` 擷取網路封包，每台電腦的網卡代號可能不同

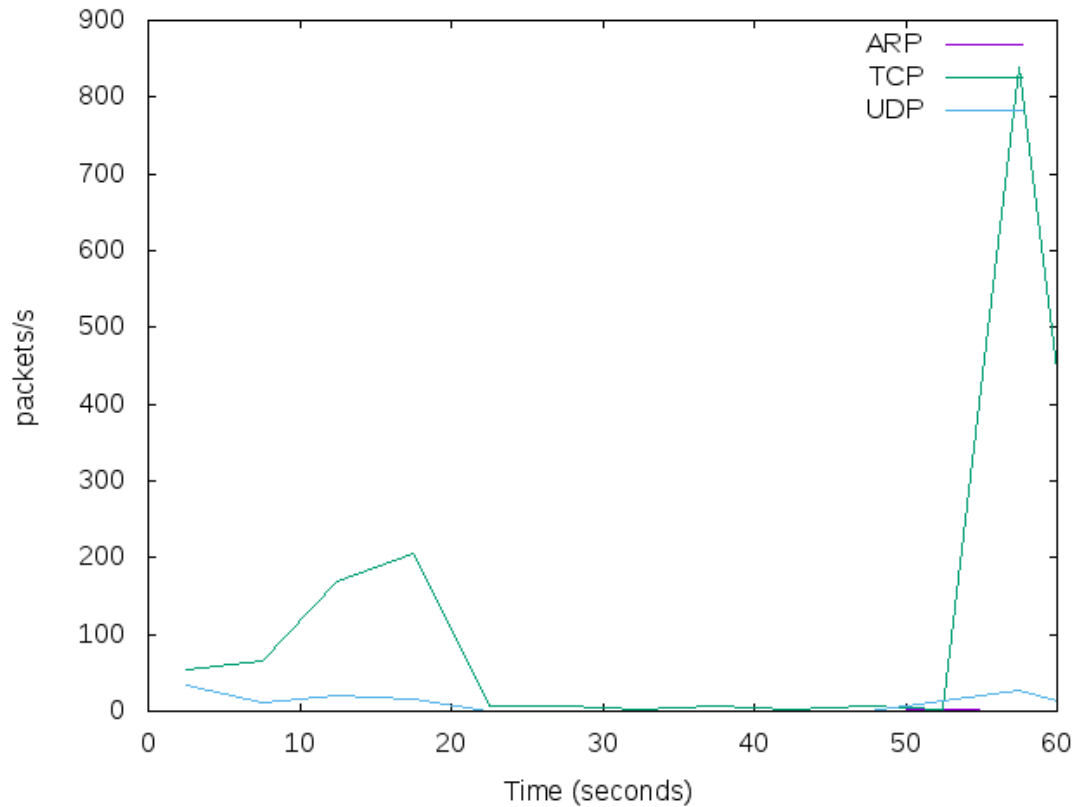
`tcpdump -i eth1 -w rawdata.dmp`

- 開啟瀏覽器瀏覽網頁約一分鐘
- 中斷 `tcpdump`
- 使用 `tcpstat` 將擷取的檔案做格式化

`tcpstat -r rawdata.dmp -o "%r %A %T %U %I %b\n" > tcpstat.log`

- 使用 `vim` 寫一個 `script`，名稱為 `script1`，`script` 內容如下圖





4. mininet & iperf & gnuplot

本題要讓同學練習在 mininet 下使用 iperf，將分為以下 4 個部份：

<請使用 iperf3 指令完成作業>

4-1. 請說明 iperf 的用途，以及在什麼情況下你會需要使用它？

Ans: iperf 是一個透過 tcp 測試系統網路能力的工具，由於可以專門測試 tcp 層不管 application layer 的干擾，因此特別適合用於針對系統底層網路能力來測試，例如網路卡、驅動程式、作業系統等等。

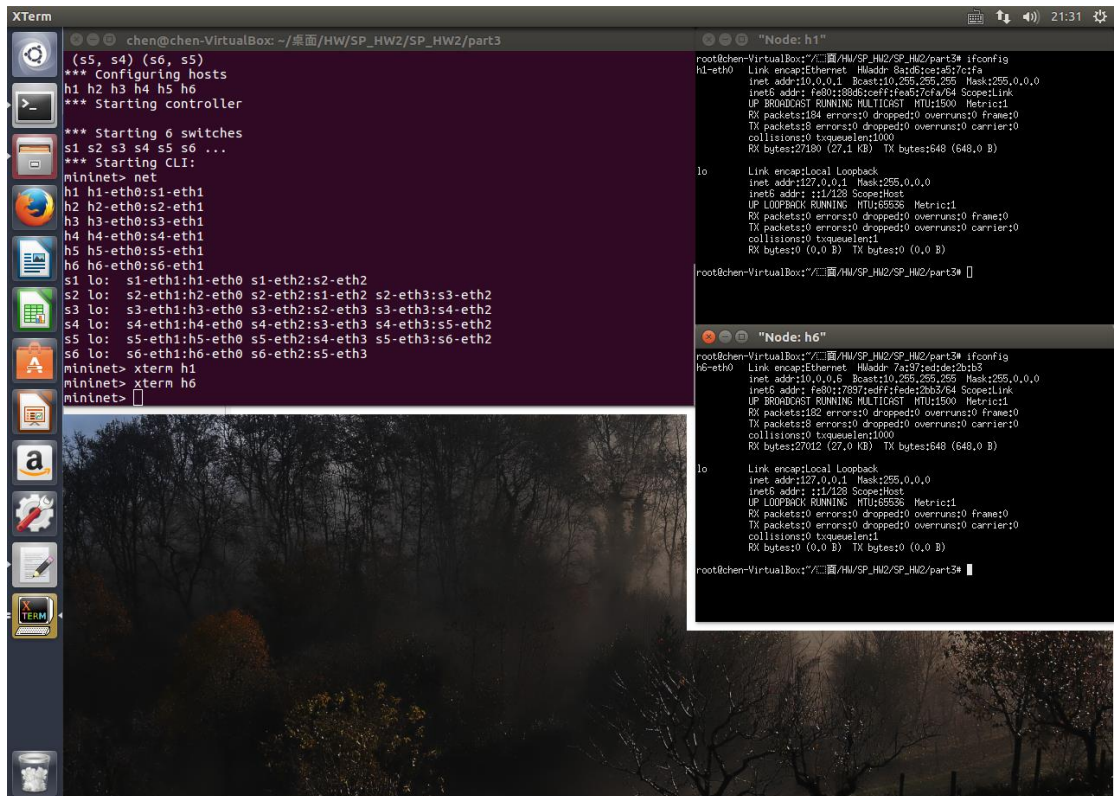
4-2. 請在 mininet 下設計 topology，使用 iperf3 指令開啟 Server 和 Client，使 Server 及 Client 之間有 0~4 個節點，並測量使用 TCP 及 UDP 傳輸時不同數量節點的頻寬變化，將結果存成檔案。

此題有以下幾點事項注意：

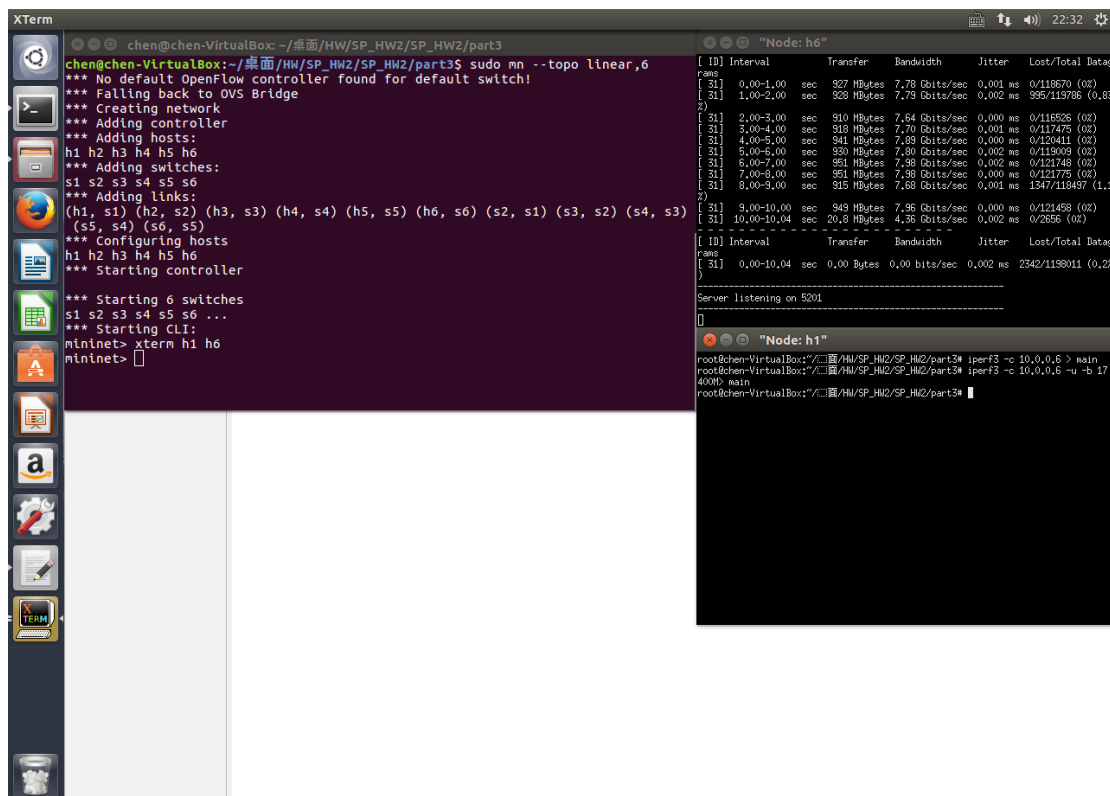
(1) 需在 mininet nodes 上完成，mininet nodes 如下圖所示。


```
终端机
chen@chen-VirtualBox: ~/桌面/HW/SP_HW2/SP_HW2/part3
[sudo] password for chen:
chen@chen-VirtualBox:~/桌面/HW/SP_HW2/SP_HW2/part3$ sudo mn --topo linear,6
*** No default OpenFlow controller found for default switch!
*** Falling back to OVS Bridge
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2 h3 h4 h5 h6
*** Adding switches:
s1 s2 s3 s4 s5 s6
*** Adding links:
(h1, s1) (h2, s2) (h3, s3) (h4, s4) (h5, s5) (h6, s6) (s2, s1) (s3, s2) (s4, s3)
(s5, s4) (s6, s5)
*** Configuring hosts
h1 h2 h3 h4 h5 h6
*** Starting controller
*** Starting 6 switches
s1 s2 s3 s4 s5 s6 ...
*** Starting CLI:
mininet> net
h1 h1-eth0:s1-eth1
h2 h2-eth0:s2-eth1
h3 h3-eth0:s3-eth1
```

```
终端机
chen@chen-VirtualBox: ~/桌面/HW/SP_HW2/SP_HW2/part3
*** Adding links:
(h1, s1) (h2, s2) (h3, s3) (h4, s4) (h5, s5) (h6, s6) (s2, s1) (s3, s2) (s4, s3)
(s5, s4) (s6, s5)
*** Configuring hosts
h1 h2 h3 h4 h5 h6
*** Starting controller
*** Starting 6 switches
s1 s2 s3 s4 s5 s6 ...
*** Starting CLI:
mininet> net
h1 h1-eth0:s1-eth1
h2 h2-eth0:s2-eth1
h3 h3-eth0:s3-eth1
h4 h4-eth0:s4-eth1
h5 h5-eth0:s5-eth1
h6 h6-eth0:s6-eth1
s1 lo: s1-eth1:h1-eth0 s1-eth2:s2-eth2
s2 lo: s2-eth1:h2-eth0 s2-eth2:s1-eth2 s2-eth3:s3-eth2
s3 lo: s3-eth1:h3-eth0 s3-eth2:s2-eth3 s3-eth3:s4-eth2
s4 lo: s4-eth1:h4-eth0 s4-eth2:s3-eth3 s4-eth3:s5-eth2
s5 lo: s5-eth1:h5-eth0 s5-eth2:s4-eth3 s5-eth3:s6-eth2
s6 lo: s6-eth1:h6-eth0 s6-eth2:s5-eth3
mininet>
```



- (2) 0~4 個節點表示 Server 和 Client 間隔幾個 Host。
 - (3) 請在 Client 端指令後加入 “> [檔案名稱]” 將輸出導入檔案中。
 - (4) 請注意 UDP 頻寬有預設的最大值，需調整頻寬最大值才能看出差異。
- UDP



TCP

The screenshot shows a terminal window with the following content:

```

chen@chen-VirtualBox: ~/桌面/HW/SP_HW2/SP_HW2/part3
*** No default OpenFlow controller found for default switch!
*** Falling back to OVS Bridge
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2 h3 h4 h5 h6
*** Adding switches:
s1 s2 s3 s4 s5 s6
*** Adding links:
(h1, s1) (h2, s2) (h3, s3) (h4, s4) (h5, s5) (h6, s6) (s2, s1) (s3, s2) (s4, s3)
(s5, s4) (s6, s5)
*** Configuring hosts
h1 h2 h3 h4 h5 h6
*** Starting controller
*** Starting 6 switches
s1 s2 s3 s4 s5 s6 ...
*** Starting CLI:
mininet> xterm h1 h6
mininet>

```

On the right, a window titled "Node: h6" shows the following output:

```

Server listening on 5201
Accepted connection from 10.0.0.1, port 40342
[ S1] local 10.0.0.8 port 5201 connected to 10.0.0.1 port 40344
[ ID] Interval      Transfer     Bandwidth
[ S1] 0.00-1.00 sec  2.54 GBytes  21.8 Gbits/sec
[ S1] 1.00-2.00 sec  2.57 GBytes  22.1 Gbits/sec
[ S1] 2.00-3.00 sec  2.23 GBytes  19.3 Gbits/sec
[ S1] 3.00-4.00 sec  2.54 GBytes  21.8 Gbits/sec
[ S1] 4.00-5.00 sec  2.49 GBytes  21.4 Gbits/sec
[ S1] 5.00-6.00 sec  2.54 GBytes  21.8 Gbits/sec
[ S1] 6.00-7.00 sec  2.57 GBytes  22.1 Gbits/sec
[ S1] 7.00-8.00 sec  2.57 GBytes  22.1 Gbits/sec
[ S1] 8.00-9.00 sec  2.47 GBytes  21.2 Gbits/sec
[ S1] 9.00-10.00 sec 2.51 GBytes  21.5 Gbits/sec
[ S1] 10.00-10.04 sec 36.6 MBytes  8.76 Gbits/sec
-----
[ ID] Interval      Transfer     Bandwidth
[ S1] 0.00-10.04 sec 0.00 Bytes   0.00 bits/sec
[ S1] 0.00-10.04 sec 25.1 GBytes  21.5 Gbits/sec
Server listening on 5201

```

At the bottom, a window titled "Node: h1" shows the command being executed:

```

root@chen-VirtualBox: ~/桌面/HW/SP_HW2/SP_HW2/part3# iperf3 -c 10.0.0.6 > main
root@chen-VirtualBox: ~/桌面/HW/SP_HW2/SP_HW2/part3#

```

4-3. 請參考 Part2 第三小題自行修改 script，將第二步 tcp 及 udp 的結果使用參考指令處理後，利用 gnuplot 繪製兩張結果圖。

參考指令：cat [檔案名稱] | grep sec | head -n 20 | tr - " " | awk '{print \$4,\$8}' > [檔案名稱]

輸出圖會類似此範例，UDP 請同學自行繪製。

Ans:QQ 盡力了...還是不會...

4-4.請說明 TCP 及 UDP 產生結果差異的原因。

Ans:

TCP 這個協定最主要的特色在於傳輸資料時，需要驗證資料，確保正確性。所以花的時間稍多一點。

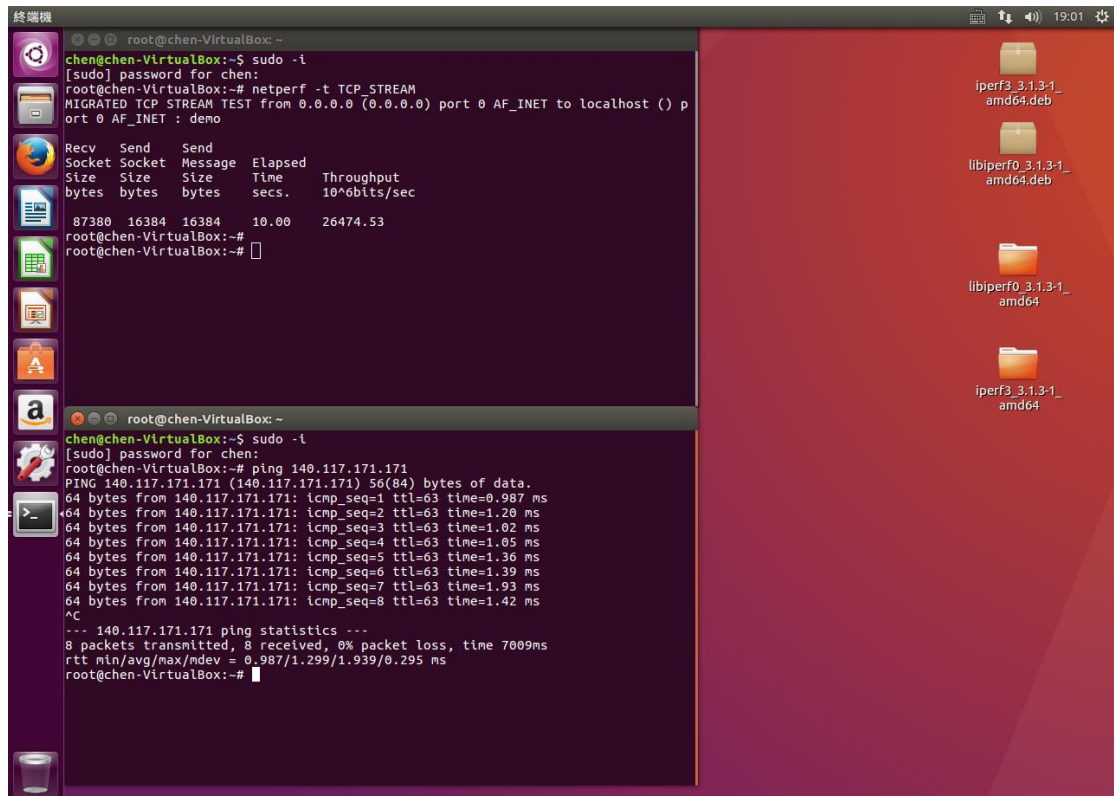
而 UDP 這個協定最主要的特色在於傳輸資料時，不需要驗證資料，不保證正確性，所以比較省時間。而一般來說，像是多媒體串流(streaming)就是使用這種協定。

5. netperf

請利用 netperf 完成以下要求，Server ip : 140.117.171.171

- a. 測量你的電腦與 Server 間的 TCP 網路效能

指令:netperf -t TCP_STREAM



The screenshot shows a terminal window with the following commands and output:

```
chen@chen-VirtualBox:~$ sudo -i
[sudo] password for chen:
root@chen-VirtualBox:~# netperf -t TCP_STREAM
MIGRATED TCP STREAM TEST from 0.0.0.0 (0.0.0.0) port 0 AF_INET to localhost () port 0 AF_INET : demo
```

Recv Size	Send Size	Send Message Size	Elapsed Time	Throughput
bytes	bytes	bytes	secs.	10^6bits/sec
87380	16384	16384	10.00	26474.53

```
root@chen-VirtualBox:~#
root@chen-VirtualBox:~#
```

Below the netperf output, there is a ping command and its statistics:

```
chen@chen-VirtualBox:~$ sudo -i
[sudo] password for chen:
root@chen-VirtualBox:~# ping 140.117.171.171
PING 140.117.171.171 (140.117.171.171) 56(84) bytes of data.
64 bytes from 140.117.171.171: icmp_seq=1 ttl=63 time=0.987 ms
64 bytes from 140.117.171.171: icmp_seq=2 ttl=63 time=1.20 ms
64 bytes from 140.117.171.171: icmp_seq=3 ttl=63 time=1.02 ms
64 bytes from 140.117.171.171: icmp_seq=4 ttl=63 time=1.05 ms
64 bytes from 140.117.171.171: icmp_seq=5 ttl=63 time=1.36 ms
64 bytes from 140.117.171.171: icmp_seq=6 ttl=63 time=1.39 ms
64 bytes from 140.117.171.171: icmp_seq=7 ttl=63 time=1.93 ms
64 bytes from 140.117.171.171: icmp_seq=8 ttl=63 time=1.42 ms
^C
--- 140.117.171.171 ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7009ms
rtt min/avg/max/mdev = 0.987/1.299/1.939/0.295 ms
root@chen-VirtualBox:~#
```

- b. 測量你的電腦與 Server 間的 UDP 網路效能

指令:netperf -t UDP_STREAM

```
終端機
root@chen-VirtualBox: ~
chen@chen-VirtualBox:~$ sudo -i
[sudo] password for chen:
root@chen-VirtualBox:~# netperf -t UDP_STREAM
MIGRATED UDP STREAM TEST from 0.0.0.0 (0.0.0.0) port 0 AF_INET to localhost () p
ort 0 AF_INET : demo

Socket  Message  Elapsed      Messages
Size   Size    Time         Okay Errors   Throughput
bytes  bytes   secs              #      #      10^6bits/sec
212992  65507   10.00       907236    0      47541.89
212992  10.00   902790      47308.91

root@chen-VirtualBox:~#
root@chen-VirtualBox:~#

root@chen-VirtualBox: ~
chen@chen-VirtualBox:~$ sudo -i
[sudo] password for chen:
Sorry, try again.
[sudo] password for chen:
Sorry, try again.
[sudo] password for chen:
root@chen-VirtualBox:~# ping 140.117.171.171
PING 140.117.171.171 (140.117.171.171) 56(84) bytes of data:
64 bytes from 140.117.171.171: icmp_seq=1 ttl=63 time=0.847 ms
64 bytes from 140.117.171.171: icmp_seq=2 ttl=63 time=6.17 ms
64 bytes from 140.117.171.171: icmp_seq=3 ttl=63 time=0.725 ms
64 bytes from 140.117.171.171: icmp_seq=4 ttl=63 time=0.751 ms
64 bytes from 140.117.171.171: icmp_seq=5 ttl=63 time=0.698 ms
64 bytes from 140.117.171.171: icmp_seq=6 ttl=63 time=0.743 ms
64 bytes from 140.117.171.171: icmp_seq=7 ttl=63 time=1.44 ms
64 bytes from 140.117.171.171: icmp_seq=8 ttl=63 time=1.42 ms
^C
--- 140.117.171.171 ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7001ms
rtt min/avg/max/mdev = 0.698/1.599/6.170/1.752 ms
root@chen-VirtualBox:~#
```

c. 請寫出正確的指令並截圖證明。