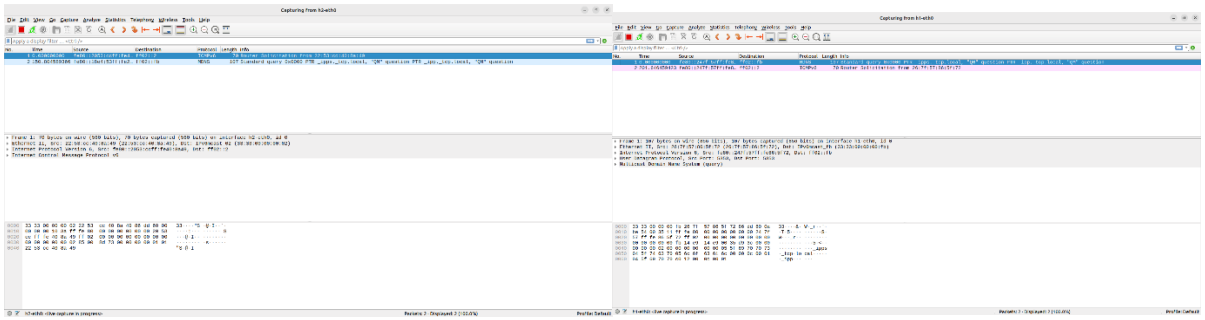
高等電腦網路作業 2

資工碩一 M113040116 蘇柏瑜

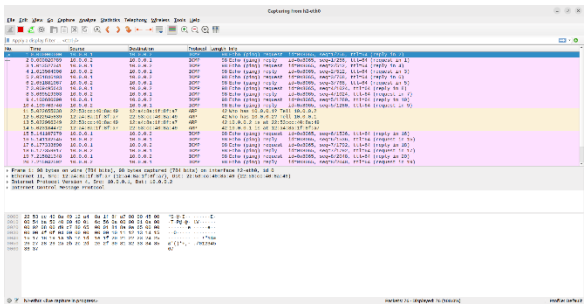
Part1:

Mininet

Host 1, host 2



H1 ping h2



Part2:

1. tshark

a. tshark -f 'icmp' -f "host 8.8.8.8"

```
root@ubuntu:/home/briansu# tshark -i enp0s3 -Y icmp -Y ip.addr==8.8.8.8
Running as user "root" and group "root". This could be dangerous.
Capturing on 'enp0s3'
 ** (tshark:6721) 20:25:30.383114 [Main MESSAGE] -- Capture started.
 ** (tshark:6721) 20:25:30.383221 [Main MESSAGE] -- File: "/tmp/wireshark_enp0s3OJMQB2.pcapng"
^C0 packets captured
```

b. tshark -w packet01.pcap -f 'icmp' -f "host 8.8.8.8"

```
root@ubuntu:/home/briansu# tshark -i enp0s3 -Y icmp -Y ip.addr==8.8.8.8 -w packe
t01.pcap
```

c.

```
^C
root@ubuntu:/home/briansu# 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=116 time=8.50 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=116 time=8.69 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=115 time=8.31 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=116 time=8.64 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=116 time=8.22 ms
64 bytes from 8.8.8.8: icmp_seq=6 ttl=115 time=9.28 ms
64 bytes from 8.8.8.8: icmp_seq=7 ttl=116 time=8.00 ms
64 bytes from 8.8.8.8: icmp_seq=8 ttl=115 time=8.51 ms
64 bytes from 8.8.8.8: icmp_seq=9 ttl=116 time=8.59 ms
64 bytes from 8.8.8.8: icmp_seq=10 ttl=116 time=9.34 ms
64 bytes from 8.8.8.8: icmp_seq=11 ttl=115 time=10.4 ms
64 bytes from 8.8.8.8: icmp_seq=12 ttl=115 time=11.3 ms
64 bytes from 8.8.8.8: icmp_seq=13 ttl=115 time=8.23 ms
```

```
briansu@ubuntu:~$ 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=47 time=22.2 ms
64 bytes from 208.67.220.220: icmp_seq=2 ttl=47 time=38.3 ms
64 bytes from 208.67.220.220: icmp_seq=3 ttl=47 time=22.6 ms
64 bytes from 208.67.220.220: icmp_seq=4 ttl=47 time=27.0 ms
64 bytes from 208.67.220.220: icmp_seq=5 ttl=47 time=22.2 ms
64 bytes from 208.67.220.220: icmp_seq=6 ttl=47 time=35.7 ms
64 bytes from 208.67.220.220: icmp_seq=7 ttl=47 time=24.2 ms
64 bytes from 208.67.220.220: icmp_seq=8 ttl=47 time=20.8 ms
64 bytes from 208.67.220.220: icmp_seq=9 ttl=47 time=23.1 ms
64 bytes from 208.67.220.220: icmp_seq=10 ttl=47 time=21.9 ms
64 bytes from 208.67.220.220: icmp_seq=11 ttl=47 time=27.6 ms
64 bytes from 208.67.220.220: icmp_seq=12 ttl=47 time=26.7 ms
64 bytes from 208.67.220.220: icmp_seq=13 ttl=47 time=26.6 ms
64 bytes from 208.67.220.220: icmp_seq=14 ttl=47 time=21.5 ms
```

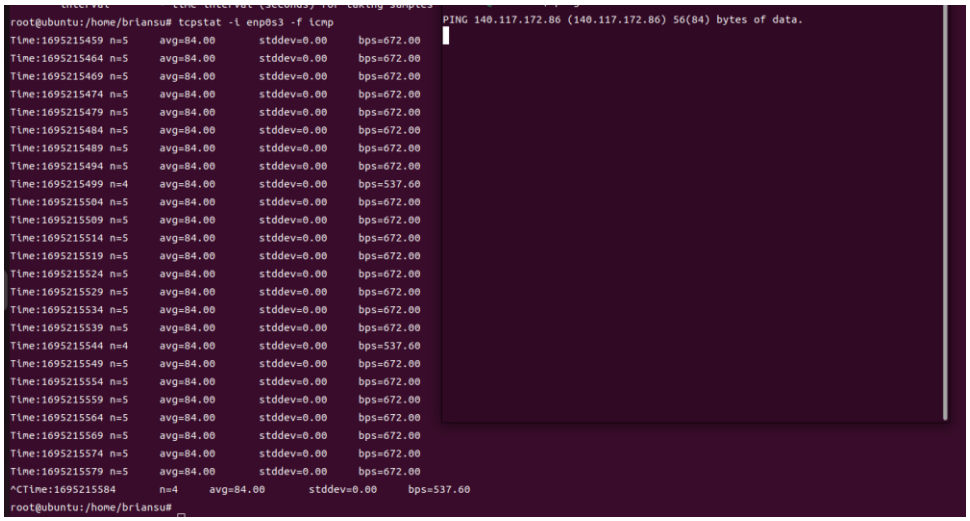
d.

```
root@ubuntu:/home/briansu# tshark -r packet01.pcap
```

e. tshark -w packet01.pcap -f 'icmp' -f "host 8.8.8.8"

2. tcpstat

tcpstat -f 'icmp'



3. tcpdump & tcpstat & gnuplot

如附檔 graph1.png

4. mininet & iperf & gnuplot

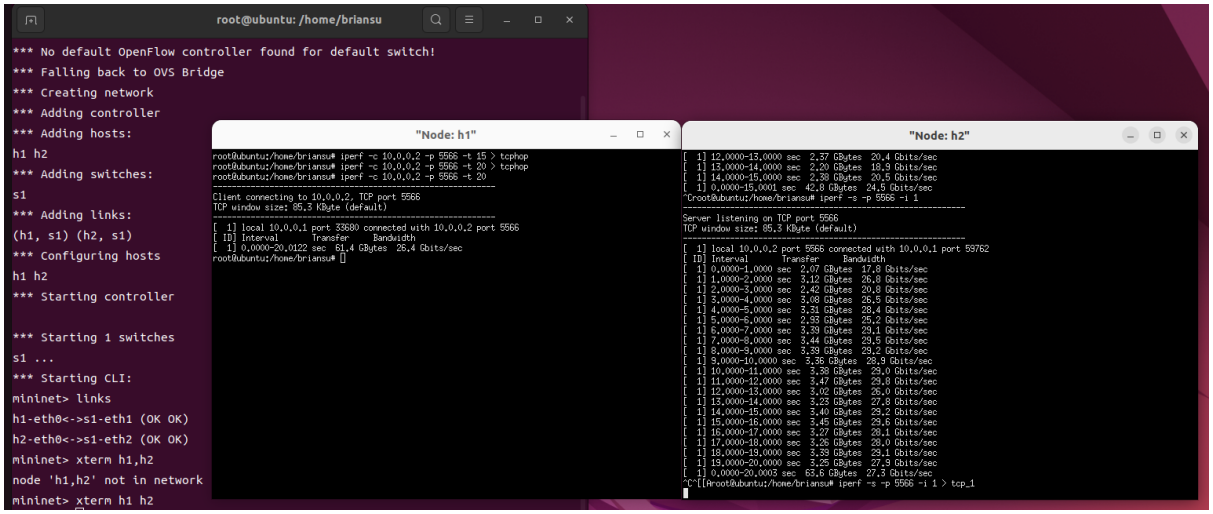
4-1

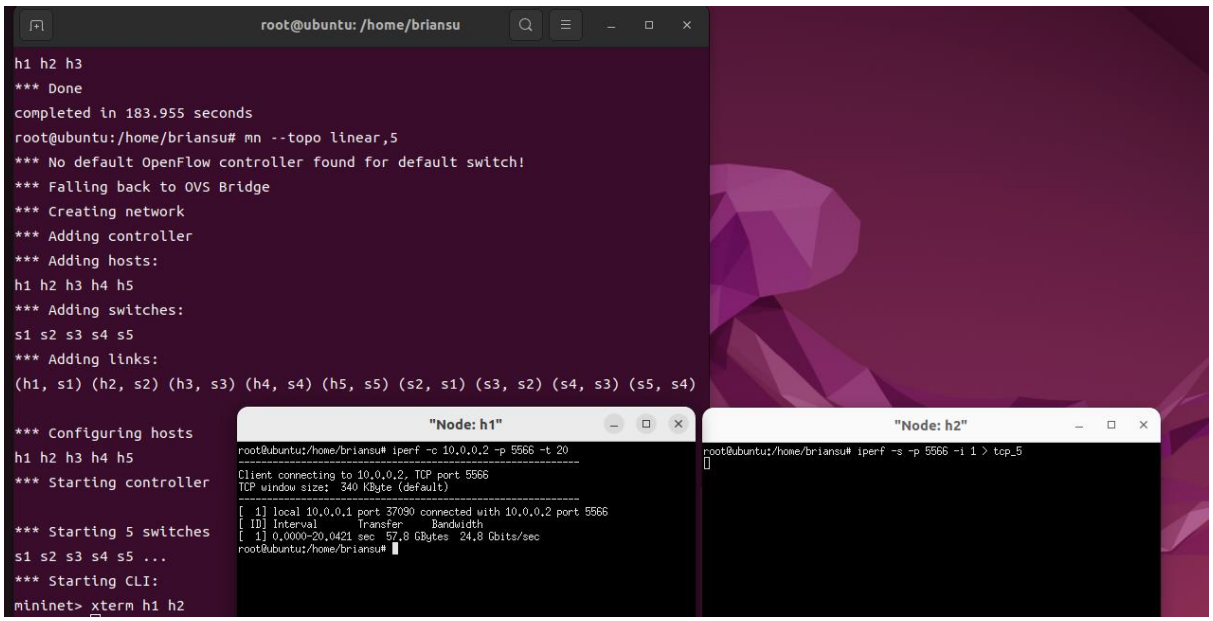
1. 測試網絡頻寬和性能： 兩個主機之間的網路連接的實際頻寬
2. 網路性能調試： 遇到性能問題時，幫助確定位置
3. 測試網路負載：模擬實際網絡流量
4. 優化網路設置： 測試不同網絡設置的效能，例如調整路由器、防火牆或交換機的參數
5. 監控網絡性能： 定期運行並記錄結果，以監控網絡性能的變化

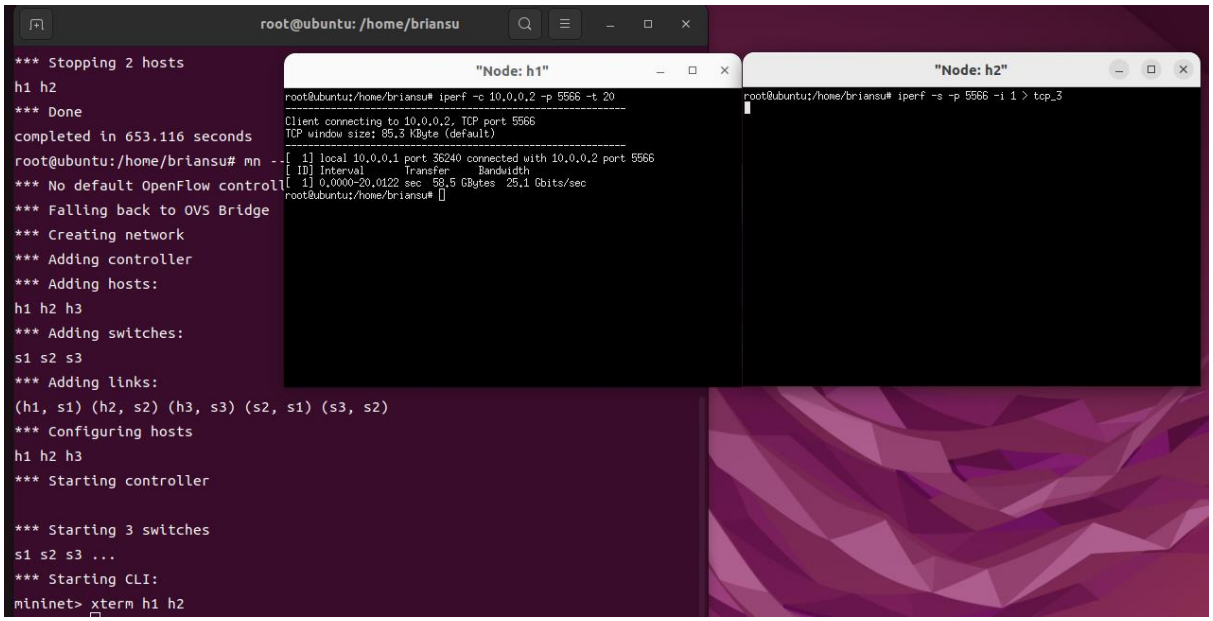
4-2

(依序為 TCP 1hop, 3hops, 5hops)

iperf -s -p 5566 -i 1
iperf -c 10.0.0.2 -p 5566 -t 20 > tcp_1
cat tcp | grep sec | head -n 20 | tr - " " | awk '{print $4,$6}' > tcp_res



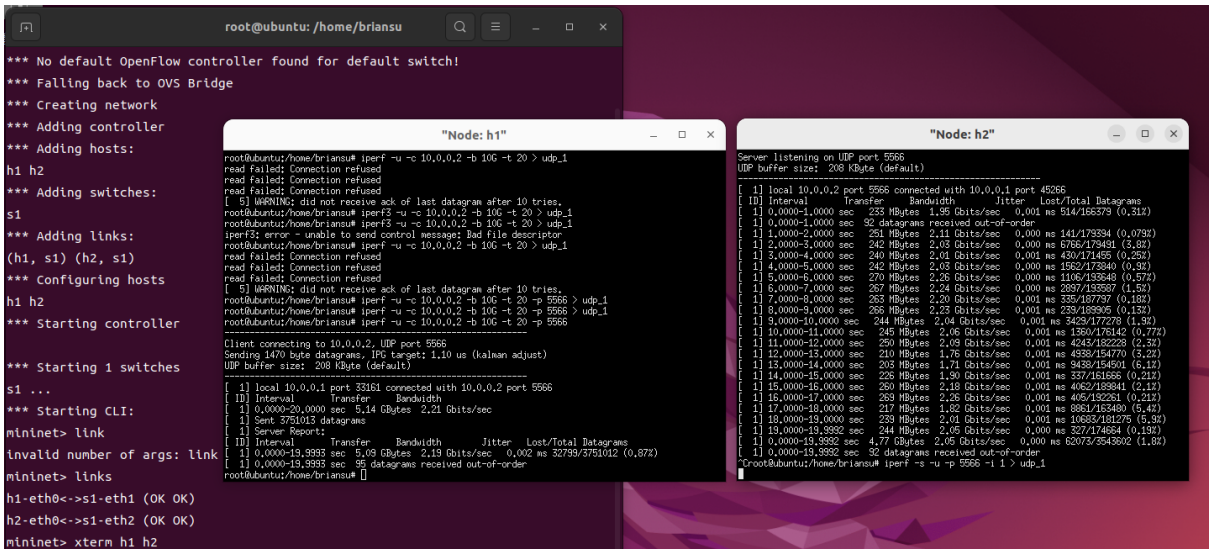


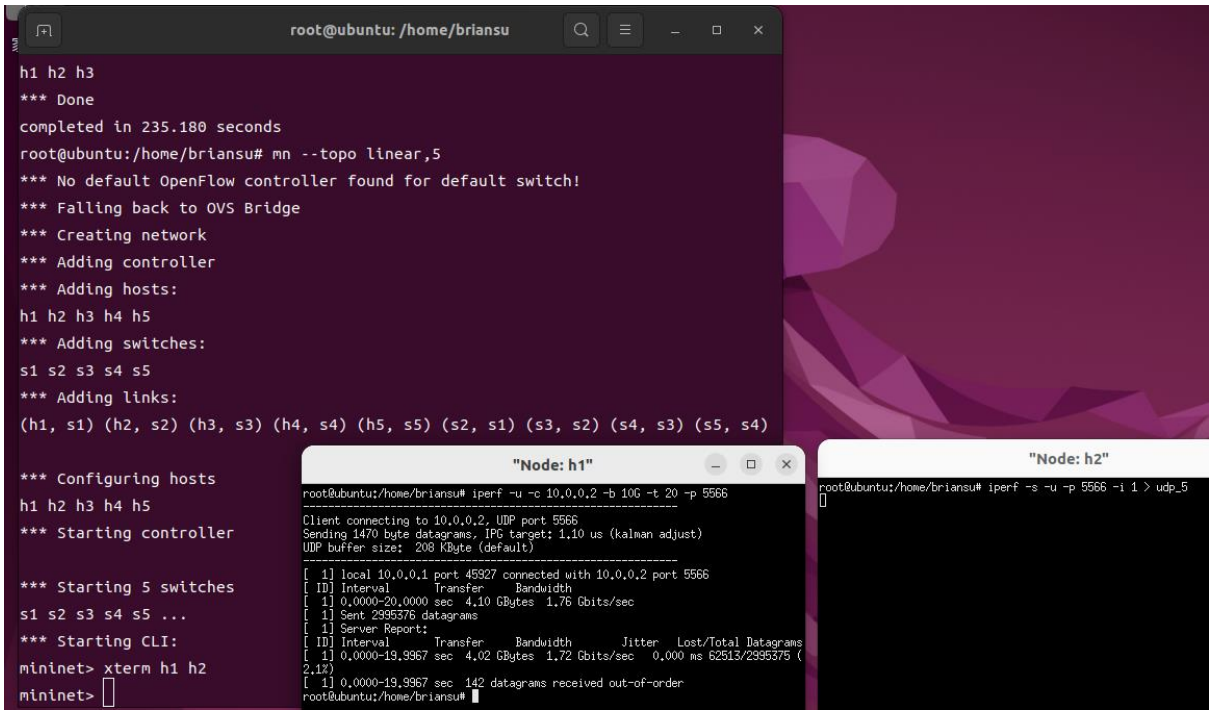


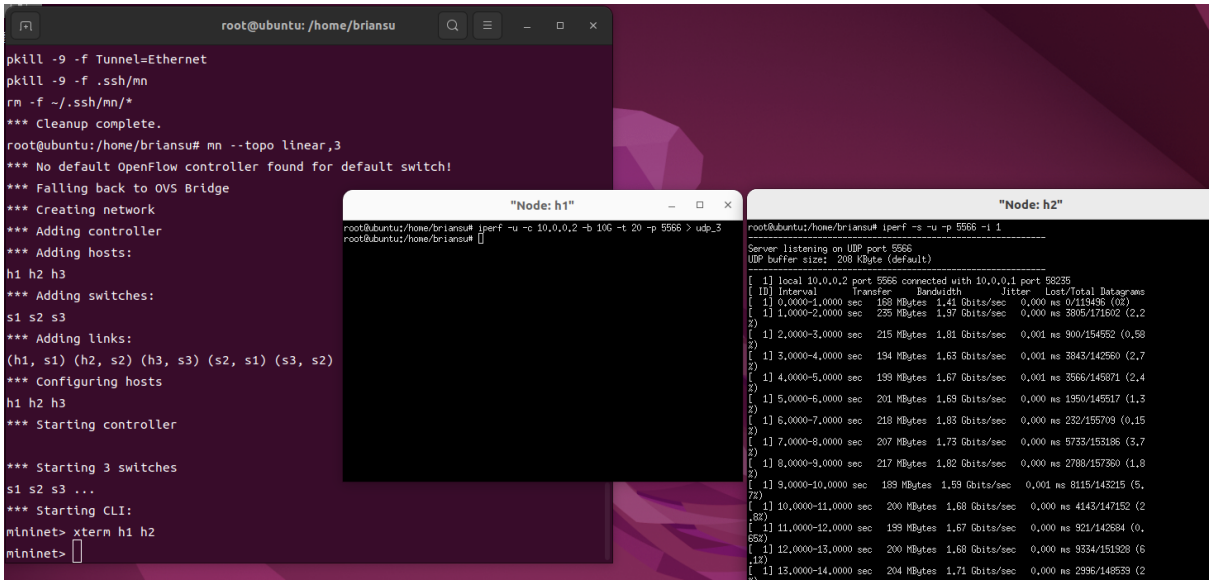
(依序為 UDP 1hop, 3hops, 5hops)

iperf -s -u -p 5566 -i 1
iperf -u -c 10.0.0.2 -b 10G -t 20 -p 5566 > udp_1

cat udp | grep MBytes | head -n 20 | tr - " " | awk '{print $4,$8}' > udp_res







4-3

如附檔 tcp.png, udp.png

4-4

可靠性：

TCP 是一種可靠的協議，它確保資料依序到達目的地，如果資料包遺失或損壞，它會重新傳輸。 這確保了資料的完整性和可靠性，但也導致了較高的延遲。

UDP 是一種不可靠的協議，它沒有資料包重新傳輸機制。 如果資料包遺失或損壞，不會進行任何修復。使得傳輸更快，但資料完整性不如 TCP 可靠。

連接性：

TCP 是面向連線的協議，建立連線需要三次握手，確保通訊的兩端都已經準備好發送和接收資料。 這種連接性適用於許多應用，如網頁瀏覽和文件傳輸。

UDP 是面向無連線的協議，它不需要建立連線。 這使得 UDP 適用於需要快速傳輸資料的應用，例如即時音視訊串流。

流量控制：

TCP 具有流量控制機制，可根據目標主機的處理能力和網路擁塞情況來控制資料的發送速率。

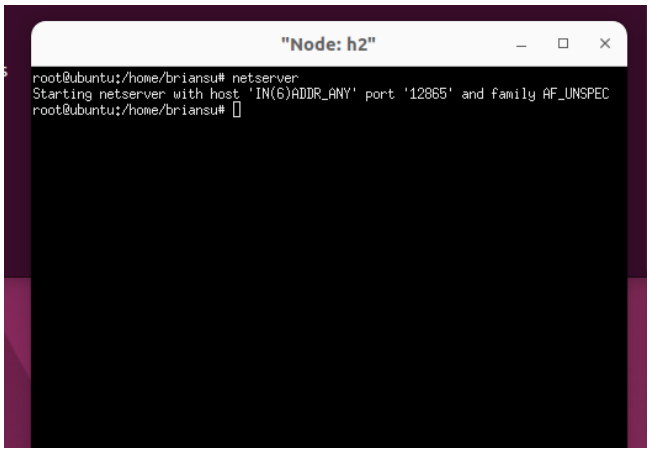
UDP 不提供流量控制，發送端會以最大速率發送數據，這可能導致網路擁塞。

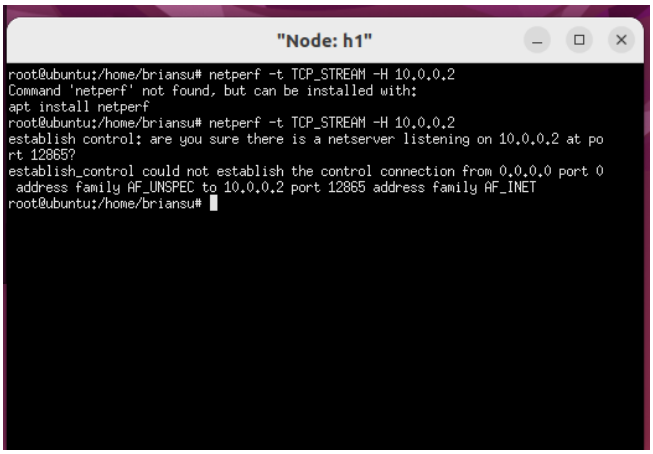
5. netperf

a.

netperf -t TCP_STREAM -H 10.0.0.2

netserver





b.

netperf -t UDP_STREAM -H 10.0.0.2

netserver