



---

# REPORT

---



109550031

李旻融

➤ **each step**

✓ **TASK 1**

1. 安裝 Wireshark 和 Oracle VM VirtualBox
2. 將 .ova file 加入 Oracle VM VirtualBox
3. 加入 github classroom lab1 並下載所需檔案
4. 測試 Mininet

✓ **TASK 2**

1. 測試 topo.py
2. 開兩個新檔案 (topo\_TCP.py / topo\_UDP.py)
3. 在新檔案裡，我分別使用了四個 hosts(h1, h2, h3, h4)和  
兩個 switches(s1, s2)。

✓ **TASK 3**

1. 在那兩個新檔案中，我使用了五條links來連接(h2->s1, h1->s1, s1->s2, s2->h3, s2->h4) 並設定每條links  
的頻寬。
2. 在 topo\_TCP.py 中，我產生了兩個 TCP 的flow，一條  
從 h2 到 h3，另一條則從 h1 到 h4。創造兩個 pcap  
檔案來記錄這些封包，再開了四個.txt檔來記錄一些和  
host相關的內容。

3. 在 `topo_UDP.py` 中，我產生了兩個 UDP 的flow，一條從 h2 到 h3，另一條則從 h1 到 h4。創造兩個 pcap 檔案來記錄這些封包，再開了四個.txt檔來記錄一些和 host相關的內容。

#### ✓ TASK 4

1. 開一個新檔案 `computRate.py`，分別讀取TCP和UDP 中的兩條flow，再去算每條的速率是多少Mbps，因為`len(packet)`的單位是byte，所以size要先乘8，接著因為時間總長是五秒，所以除以5，又因為單位是Mbps，所以需要再除以1000000，才會是最終的速率。

#### ✓ TASK 5

1. 將檔案從虛擬機上傳到github
2. 從github下載檔案至自己的電腦
3. 用 Wireshark 開pcap檔，利用filter找出所需檔案，比較`computRate`和wireshark上的答案是否一致。

➤ **How to run my program**

```
$ sudo python3 topo_TCP.py
```

```
$ sudo python3 topo_UDP.py
```

```
$ sudo python3 computRate.py
```

➤ **Observation from the results in this lab**

我發現TCP和UDP相比較慢一些，這可能是因為TCP須建立連線，進行偵錯並確保依照傳送順序接收檔案。除此之外，Wireshark 和 ComputRate.py 跑出來的資料也有些微的不同。

➤ **What does each iPerf command you used mean?**

-w: window size

-s: server

-c: client

-u: UDP

-b: bandwidth

-i: interval

-t: time

-p: port

- What is your command to filter each flow in Wireshark?

Flow1(h2->h3): Ip.src == 10.0.0.2 && Ip.dis == 10.0.0.3

Flow3(h1->h4): Ip.src == 10.0.0.1 && Ip.dis == 10.0.0.4

- the result of computeRate.py

```

---TCP---
Flow1(h1->h4):  2.0115104  Mbps
Flow3(h2->h3):  3.0290624  Mbps

---UDP---
Flow1(h1->h4):  2.0800384  Mbps
Flow3(h2->h3):  3.333184   Mbps

```

- the results of statistics of Wireshark

1. TCP\_h3.pcap

#### Statistics

<u>Measurement</u>	<u>Captured</u>	<u>Displayed</u>	<u>Marked</u>
Packets	1322	671 (50.8%)	—
Time span, s	8.576	5.047	—
Average pps	154.1	133.0	—
Average packet size, B	1438	2763	—
Bytes	1900724	1854294 (97.6%)	0
Average bytes/s	221k	367k	—
Average bits/s	1772k	2939k	—

## 2. TCP\_h4.pcap

### Statistics

<u>Measurement</u>	<u>Captured</u>	<u>Displayed</u>	<u>Marked</u>
Packets	898	446 (49.7%)	—
Time span, s	8.576	5.062	—
Average pps	104.7	88.1	—
Average packet size, B	1408	2761	—
Bytes	1264584	1231284 (97.4%)	0
Average bytes/s	147k	243k	—
Average bits/s	1179k	1945k	—

## 3. UDP\_h3.pcap

### Statistics

<u>Measurement</u>	<u>Captured</u>	<u>Displayed</u>	<u>Marked</u>
Packets	1444	1376 (95.3%)	—
Time span, s	11.964	5.296	—
Average pps	120.7	259.8	—
Average packet size, B	1449	1505	—
Bytes	2092682	2070370 (98.9%)	0
Average bytes/s	174k	390k	—
Average bits/s	1399k	3127k	—

## 4. UDP\_h4.pcap

### Statistics

<u>Measurement</u>	<u>Captured</u>	<u>Displayed</u>	<u>Marked</u>
Packets	927	859 (92.7%)	—
Time span, s	11.964	5.433	—
Average pps	77.5	158.1	—
Average packet size, B	1413	1499	—
Bytes	1310056	1287744 (98.3%)	0
Average bytes/s	109k	237k	—
Average bits/s	875k	1896k	—

- Does the throughput match the bottleneck throughput of the path?

是，Flow1 和 Flow3 的吞吐量相加大約等於s1, s2中間的頻寬

- Do you observe the same throughput from TCP and UDP?

Can both flows equally share the bandwidth?

Flow1 和 Flow3 無論在TCP還是UDP中，吞吐量都差不多，但

UDP會略大於TCP。Flow1 和 Flow3 並沒有恰好平分頻寬。

- What you have learned from this lab?

在這次的lab裡，我知道該如何建造出host, switch, link，並利

用他們創造出flow，我也學到該如何利用mininet來製造一個小

型網路，並寫一個code來計算吞吐量，再利用wireshark來做檢

測，除此之外，我還學到了TCP和UDP的相同與相異處。

- What difficulty you have met in this lab?

在寫computeRate.py的時候，我試了好多寫法都寫不出正確

的吞吐量，後來發現根本不用用到.time，是我自己想得太

複雜了，只要直接將五秒帶入就好。