0716306 NSC lab2 report

Part 1 question

1.

(a) h2 可以 ping h3,因為 h2 和 h3 之間利用 switch 相連並在同一個網域中,不需要用到 routing

```
topology.py
h1 doesn't have connectivity to 192.168.1.65
h1 doesn't have connectivity to 192.168.1.66
   doesn't have connectivity to 192.168.3.1
             t have connectivity to 192.168.3.2
t have connectivity to 192.168.3.1
   doesn't
   doesn'
    doesn't have connectivity to 192.168.
                have connectivity
                                               192.168.
    doesn't
               have connectivity to 192.168.3.2
    doesn't have connectivity to 192.168.1.65
               have connectivity to 192.168.1.66 have connectivity to 192.168.1.65
    doesn't have connectivity to 192.168.1.66
mininet> h2 ping h3
PING 192.168.1.66 (192.168.1.66) 56(84) bytes of data.
   bytes from 192.168.1.66: icmp_seq=1 ttl=64
bytes from 192.168.1.66: icmp_seq=2 ttl=64
bytes from 192.168.1.66: icmp_seq=3 ttl=64
                                                                    time=0.351
                                                                   time=0.115
                                                                                    ms
                                                                   time=0.099
    bytes from 192.168.1.66: icmp_seq=4 ttl=64
--- 192.168.1.66 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3238ms
rtt min/avg/max/mdev = 0.099/0.169/0.351/0.105 ms
```

(b) h2 不能 ping h4,因為 h2 到 h4 的 path 需要經過 router,但此時所有 router 的 routing table 並沒有到其他網域的 routing path,因此無法成功

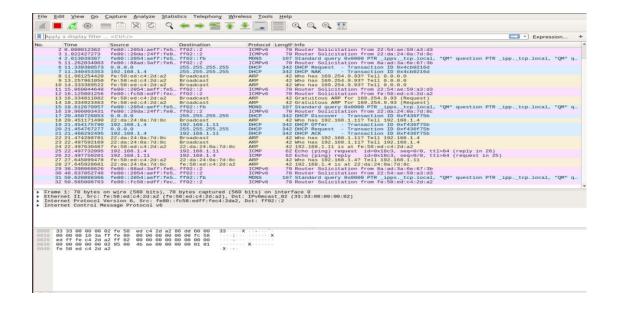
```
mininet> h2 ping h4
connect: Network is unreachable
mininet>
```

2. 在完成除 h1 以外的 topology 的部分後,執行 topology 的 check()結果如下圖

```
demo@SDN-NFV:~/Desktop/NSC_lab2$ sudo python topology.py
h1 doesn't have connectivity to 192.168.1.65
h1 doesn't have connectivity to 192.168.1.66
h1 doesn't have connectivity to 192.168.3.1
h1 doesn't have connectivity to 192.168.3.2
WRONG ANSWER
```

Part 2 question

3. 執行完 h1 dhclient h1-eth0 後,wireshark 的截圖如下

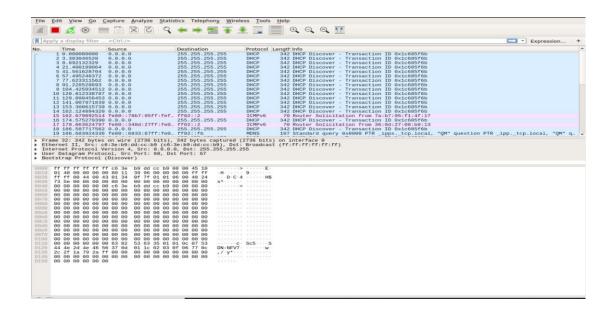


在 cmd 輸入 h1 ifconfig 後如下(原本我將 DHCP 給予的 IP range 設為從 192.168.1.0 開始,但因為分配到的 IP 是代表這個網段的 IP,因此後來 設為從 192.168.1.10 開始,到 192.168.1.61 結束)

```
mininet> h1 ifconfig
h1-eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.11 netmask 255.255.292 broadcast 192.168.1.63
    inet6 fe80::fc58:edff:fec4:2da2 prefixlen 64 scopeid 0x20<link>
    ether fe:58:ed:c4:2d:a2 txqueuelen 1000 (Ethernet)
    RX packets 53 bytes 5816 (5.8 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 22 bytes 2248 (2.2 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 20 bytes 1000 (1000.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 20 bytes 1000 (1000.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

4. 然後開始嘗試讓 DHCP server 將 IP 指派給其他的 host,此處以 h2 為例, 結果如下,可以看見 h2 的 eth0 介面並無變化,並且參照 wireshark 的介面,觀察 h2-eth0 收到的封包中也並無變化,而因為 dhcpd.conf 中沒有宣告除 192.168.1.0/26 以外的 subnet,因此 DHCP server 不會知道如何分配 IP 給除 h1 以外的 host



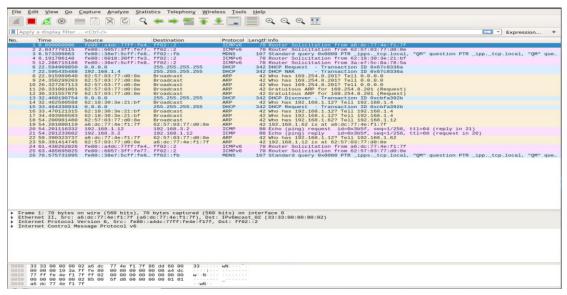
下圖為 h2 ifconfig 後的畫面,可看見 h2 的 IP 並未被修改

```
mininet> h2 ifconfig
h2-eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.65 netmask 255.255.255.192 broadcast 192.168.1.127
    inet6 fe80::c43e:b9ff:fedd:ccb9 prefixlen 64 scopeid 0x20<link>
    ether c6:3e:b9:dd:cc:b9 txqueuelen 1000 (Ethernet)
    RX packets 65 bytes 6062 (6.0 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 54 bytes 12364 (12.3 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

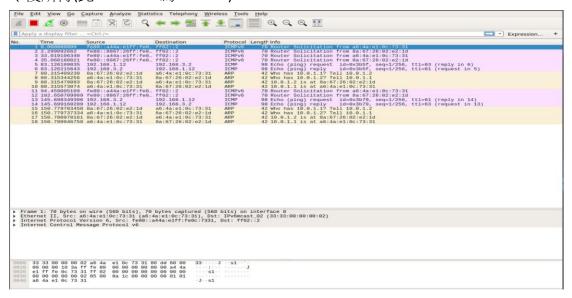
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 22 bytes 1168 (1.1 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 22 bytes 1168 (1.1 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Part 3 question

5. 首先是 h1 ping h5 -c 1 的指令,圖中編號為 20 和 21 的 ICMP 封包為執 行後所得(此 interface 為 r1-eth1)

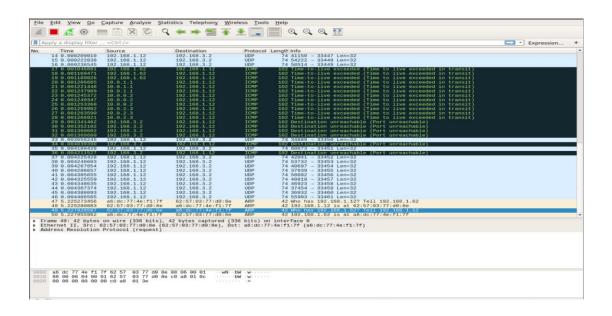


下圖為 h5 ping h1 -c 1 的指令,編號 13 和 14 的 ICMP 封包為執行完指令後所得(此 interface 為 r1-eth0)



在 h1 ping h5 的過程中,r1 的 eth1 會偵測到 h1 發出的 request,而在 h5 ping h1 的過程中 r1 的 eth0 會 5 偵測到 h5 發出的 request,並個別對 request 做 reply

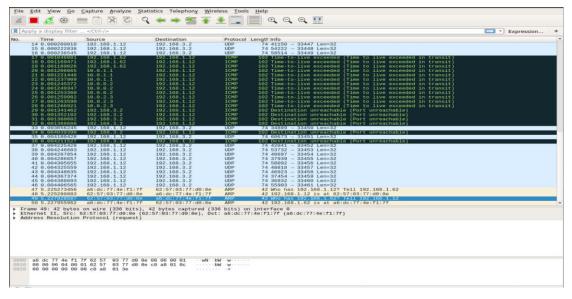
6. 下圖包含所有 h1 收到的 ICMP 封包



下圖為 h1 traceroute h5 的結果,但據替為甚麼只能看到第一、第二和 第五條 hop detail 我不是很清楚,因為我第一次執行就是這樣的結果

```
mininet> h1 traceroute h5
QStandardPaths: XDG_RUNTIME_DIR not set, defaulting to '/tmp/runtime-root'
traceroute to 192.168.3.2 (192.168.3.2), 30 hops max, 60 byte packets
1 _gateway (192.168.1.62) 1.068 ms 1.134 ms 1.134 ms
2 10.0.1.1 (10.0.1.1) 1.139 ms 1.140 ms 1.141 ms
3 10.0.0.2 (10.0.0.2) 1.135 ms 1.126 ms 1.114 ms
4 10.0.2.3 (10.0.2.3) 1.107 ms 1.098 ms 1.088 ms
5 192.168.3.2 (192.168.3.2) 1.149 ms 1.146 ms 1.140 ms
mininet>
```

7. 使用第 6 題中的圖片做說明,下圖中編號為 17 到 19 的封包是 r1 的 gateway(也就是 192.168.1.62/26,是 r1 對 h1 和 DHCP server 的 gateway) 傳給 h1 的 ICMP,而編號為 20 到 22 的封包是 r2 的 eth1 傳給 h1 的 ICMP,這兩個 sender 送出的 ICMP 可以幫助 h1 找到抵達 h5 的路徑(1st hop 和 2nd hop)



8. 而關於第五個 hop,則是由編號 29 到 32、34、36 的 ICMP 封包組成, sender 皆為 h5

