

# CNLab1 - NAT & Firewall

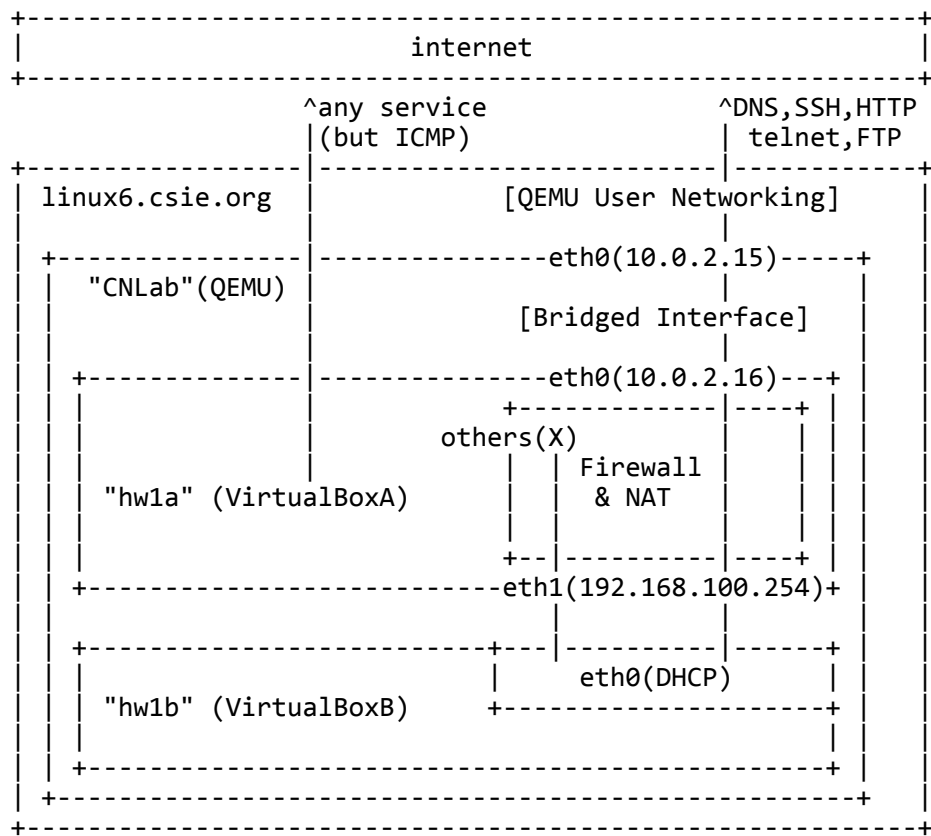
Team #1

b04303128 經濟四 吳海韜  
b04705001 資工四 陳約廷  
b05902093 資工三 顏睿楠  
b03901056 電機四 孫凡耕  
b05202043 物理三 呂佳軒  
b06902021 資工二 吳聖福  
(貢獻度平均各1/6)

## Virtual Machines in Virtual Machine

本次的實驗是利用VirtualBox模擬網路上兩台電腦的設定，其中：

- 電腦A作為NAT&Firewall，連接外部網路和子網路
- 電腦B透過電腦A提供private IP連上internet



上圖是我們這組實作本次實驗的架構說明圖（雙引號表 hostname）

具體方式是在工作站 linux6 架設 qemu 遠端操作來代替實際使用R204教室的電腦，也就是虛擬機裡面有兩台虛擬機，這種方式的優點有：

1. 所有六位組員可以用 ssh 或 vncviewer 同時進行設定和測試，提高分工效率

2. 在qemu裡面有root權限，設定port forwarding較方便，可以通過嵌套多層ssh的方式從各自的筆電直接對電腦A或電腦B下達shell command

- 以連進電腦B為例，透過linux6轉給cnlab1轉給hw1a再轉給hw1b
- `ssh -p 20022 cnlab@linux6.csie.org "ssh -p 21022 127.0.0.1 hostname"`

(註：因為QEMU架在linux6上設定的限制，hw1a 和 hw1b 網路服務正常，但最多只能ping到 10.0.2.15 )

## Firewall & NAT Settings

### Who is computer B?

hw1b 透過 hw1a 連到外部網路，我們在 hw1a 架設DHCP Server中設定了發放IP address的範圍是  
`range 192.168.100.20 192.168.100.100`

在這個實驗架構下 hw1b 每次都會拿到 192.168.100.20 這個IP

*dhcpd.conf*

```
15 # option definitions common to all supported networks...
16 option domain-name "cnlab.org";
17 option domain-name-servers 8.8.8.8, 140.112.254.4;
18
19 default-lease-time 6000;
20 max-lease-time 72000;
21
22 subnet 192.168.100.0 netmask 255.255.255.0 {
23     ... range 192.168.100.20 192.168.100.100;
24     ... option routers 192.168.100.254;
25     ... option broadcast-address 192.168.100.255;
26 }
```

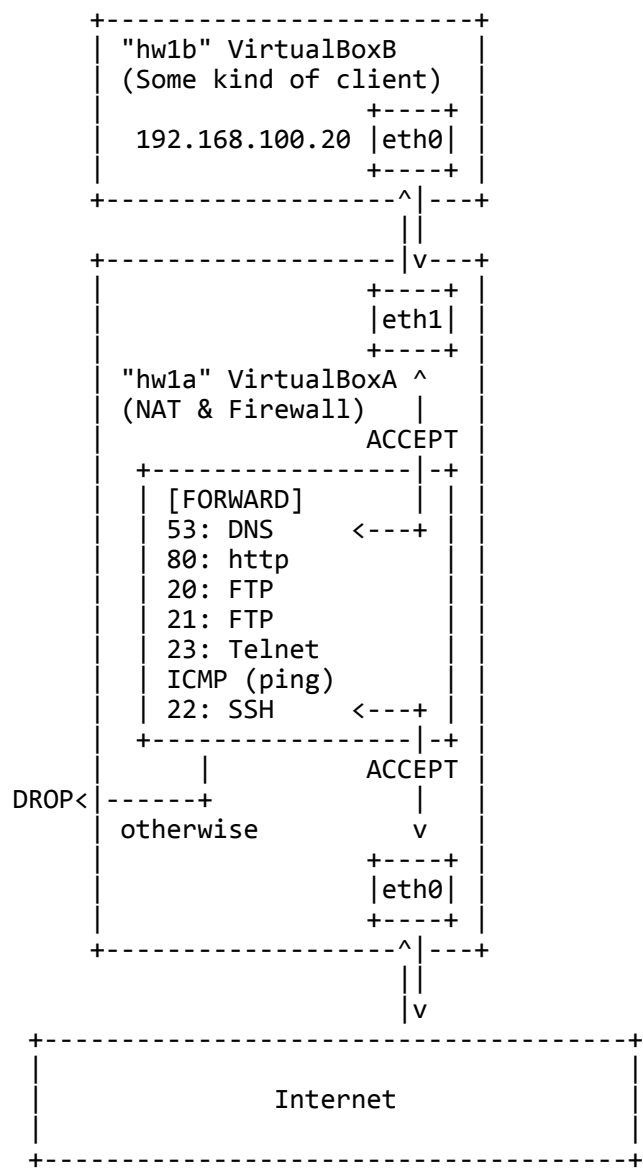
hw1a 為了將來自子網路的封包轉發出去，在透過 iptables 處理防火牆的最後，加入以下指令修改NAT Table：

*firewall.sh* (節錄，詳見文末附錄)

```
42 # nat
43 sudo iptables -t nat -A POSTROUTING -s 192.168.100.0/24 -o eth0 -j MASQUERADE
44
45 # forward
46 sudo sh -c "echo 1 > /proc/sys/net/ipv4/ip_forward"
```

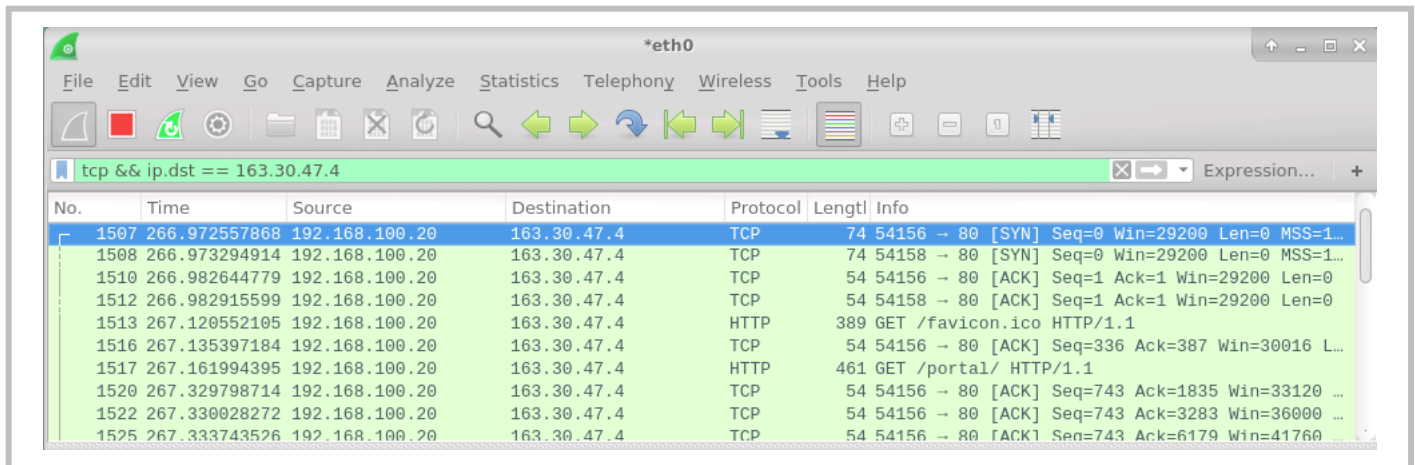
# The Firewall Rules

- 1. hw1a 自己的網路活動不受影響
- 2. 需要被轉發的封包預設DROP（也就是hw1b的封包），在逐一設定要通過的「白名單」
- 3. 根據SPEC，這次允許通過的程式是：
  - A. DNS
  - B. HTTP
  - C. FTP
  - D. Telnet
  - E. Ping
  - F. SSH —(∵ Team #1 ≡ 1 mod 4)



# Experiment with Wireshark

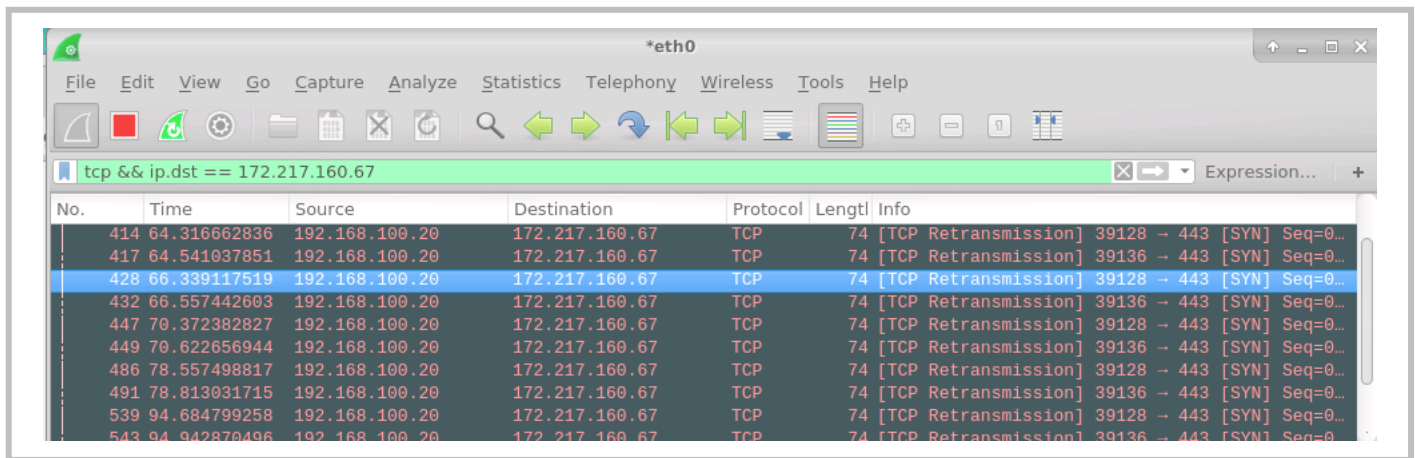
✅ http works (南門國小 <http://163.30.47.4/> )



Wireshark capture showing HTTP traffic to 163.30.47.4. The filter is `tcp && ip.dst == 163.30.47.4`. The capture shows a successful GET request for `/favicon.ico` and `/portal/`.

No.	Time	Source	Destination	Protocol	Length	Info
1507	266.972557868	192.168.100.20	163.30.47.4	TCP	74	54156 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1...
1508	266.973294914	192.168.100.20	163.30.47.4	TCP	74	54158 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1...
1510	266.982644779	192.168.100.20	163.30.47.4	TCP	54	54156 → 80 [ACK] Seq=1 Ack=1 Win=29200 Len=0
1512	266.982915599	192.168.100.20	163.30.47.4	TCP	54	54158 → 80 [ACK] Seq=1 Ack=1 Win=29200 Len=0
1513	267.120552105	192.168.100.20	163.30.47.4	HTTP	389	GET /favicon.ico HTTP/1.1
1516	267.135397184	192.168.100.20	163.30.47.4	TCP	54	54156 → 80 [ACK] Seq=336 Ack=387 Win=30016 L...
1517	267.161994395	192.168.100.20	163.30.47.4	HTTP	461	GET /portal/ HTTP/1.1
1520	267.329798714	192.168.100.20	163.30.47.4	TCP	54	54156 → 80 [ACK] Seq=743 Ack=1835 Win=33120 ...
1522	267.330028272	192.168.100.20	163.30.47.4	TCP	54	54156 → 80 [ACK] Seq=743 Ack=3283 Win=36000 ...
1525	267.333743526	192.168.100.20	163.30.47.4	TCP	54	54156 → 80 [ACK] Seq=743 Ack=6179 Win=41760 ...

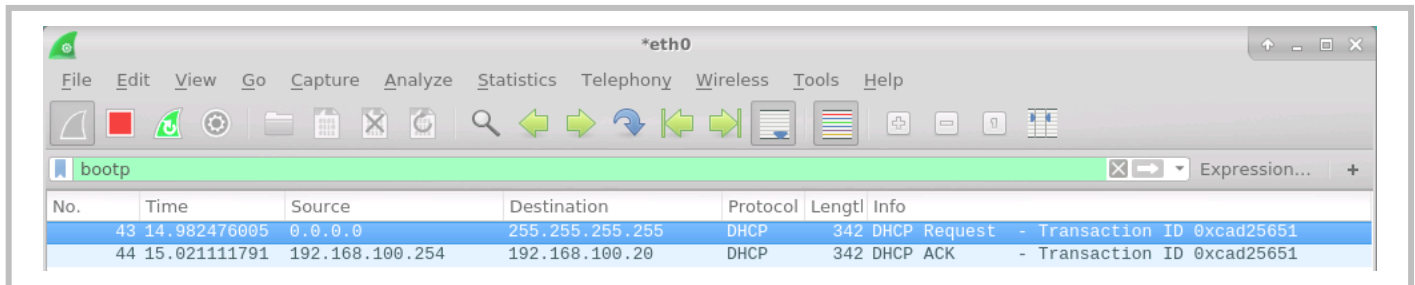
❌ https does not work (Google <https://172.217.160.78/> )



Wireshark capture showing failed HTTPS traffic to 172.217.160.67. The filter is `tcp && ip.dst == 172.217.160.67`. The capture shows multiple TCP retransmissions, indicating a connection failure.

No.	Time	Source	Destination	Protocol	Length	Info
414	64.316662836	192.168.100.20	172.217.160.67	TCP	74	[TCP Retransmission] 39128 → 443 [SYN] Seq=0...
417	64.541037851	192.168.100.20	172.217.160.67	TCP	74	[TCP Retransmission] 39136 → 443 [SYN] Seq=0...
428	66.339117519	192.168.100.20	172.217.160.67	TCP	74	[TCP Retransmission] 39128 → 443 [SYN] Seq=0...
432	66.557442603	192.168.100.20	172.217.160.67	TCP	74	[TCP Retransmission] 39136 → 443 [SYN] Seq=0...
447	70.372382827	192.168.100.20	172.217.160.67	TCP	74	[TCP Retransmission] 39128 → 443 [SYN] Seq=0...
449	70.622656944	192.168.100.20	172.217.160.67	TCP	74	[TCP Retransmission] 39136 → 443 [SYN] Seq=0...
486	78.557498817	192.168.100.20	172.217.160.67	TCP	74	[TCP Retransmission] 39128 → 443 [SYN] Seq=0...
491	78.813031715	192.168.100.20	172.217.160.67	TCP	74	[TCP Retransmission] 39136 → 443 [SYN] Seq=0...
539	94.684799258	192.168.100.20	172.217.160.67	TCP	74	[TCP Retransmission] 39128 → 443 [SYN] Seq=0...
543	94.942870496	192.168.100.20	172.217.160.67	TCP	74	[TCP Retransmission] 39136 → 443 [SYN] Seq=0...

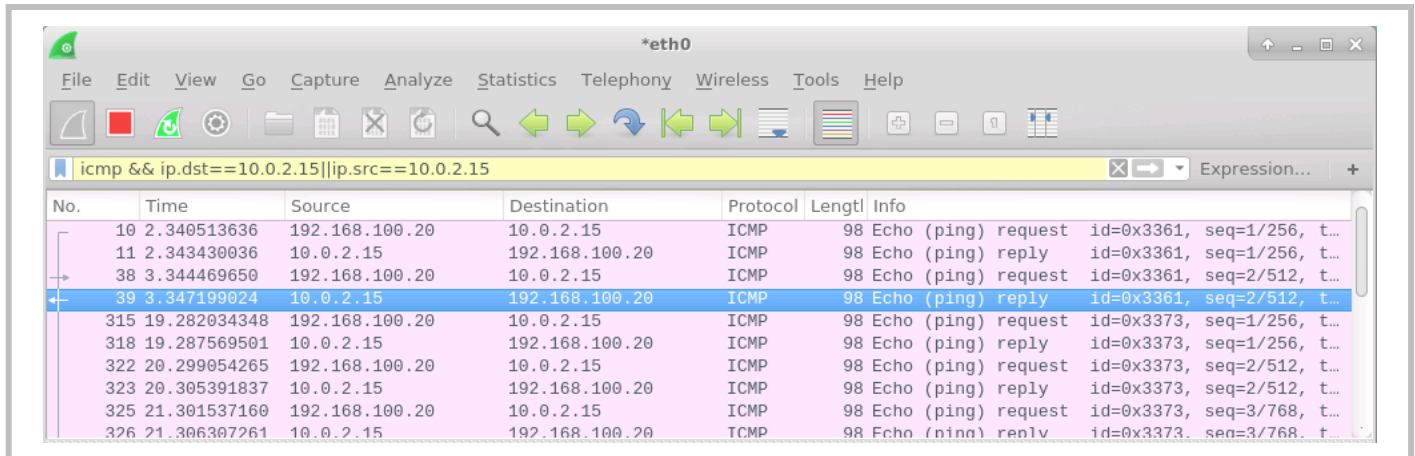
✅ DHCP from VirtualBoxA



Wireshark capture showing DHCP traffic. The filter is `bootp`. The capture shows a DHCP Request from 0.0.0.0 to 255.255.255.255 and a DHCP ACK from 192.168.100.254 to 192.168.100.20.

No.	Time	Source	Destination	Protocol	Length	Info
43	14.982476005	0.0.0.0	255.255.255.255	DHCP	342	DHCP Request - Transaction ID 0xcad25651
44	15.021111791	192.168.100.254	192.168.100.20	DHCP	342	DHCP ACK - Transaction ID 0xcad25651

✅ ICMP (ping to QEMU)



Wireshark capture showing ICMP ping traffic to 10.0.2.15. The filter is `icmp && ip.dst == 10.0.2.15 | ip.src == 10.0.2.15`. The capture shows multiple Echo (ping) requests and replies between 192.168.100.20 and 10.0.2.15.

No.	Time	Source	Destination	Protocol	Length	Info
10	2.340513636	192.168.100.20	10.0.2.15	ICMP	98	Echo (ping) request id=0x3361, seq=1/256, t...
11	2.343430036	10.0.2.15	192.168.100.20	ICMP	98	Echo (ping) reply id=0x3361, seq=1/256, t...
38	3.344469650	192.168.100.20	10.0.2.15	ICMP	98	Echo (ping) request id=0x3361, seq=2/512, t...
39	3.347199024	10.0.2.15	192.168.100.20	ICMP	98	Echo (ping) reply id=0x3361, seq=2/512, t...
315	19.282034348	192.168.100.20	10.0.2.15	ICMP	98	Echo (ping) request id=0x3373, seq=1/256, t...
318	19.287569501	10.0.2.15	192.168.100.20	ICMP	98	Echo (ping) reply id=0x3373, seq=1/256, t...
322	20.299054265	192.168.100.20	10.0.2.15	ICMP	98	Echo (ping) request id=0x3373, seq=2/512, t...
323	20.305391837	10.0.2.15	192.168.100.20	ICMP	98	Echo (ping) reply id=0x3373, seq=2/512, t...
325	21.301537160	192.168.100.20	10.0.2.15	ICMP	98	Echo (ping) request id=0x3373, seq=3/768, t...
326	21.306307261	10.0.2.15	192.168.100.20	ICMP	98	Echo (ping) reply id=0x3373, seq=3/768, t...

```
Applications Places

cnlab@CNLab ( 0 ) :~$ ifconfig
eth3: valid lft 86018sec preferred lft 86018sec
      inet6 fe80::3a99:ffc4:d0bd:1eb0/64 scope link
      valid lft forever preferred_lft forever
      Link encap:Ethernet HWaddr 52:54:00:12:35:79
      inet addr:10.0.2.15 Bcast:10.0.2.255 Mask:255.255.255
      inet6 addr: fec0::2d57:6afa:fa16:3e4b/64 Scope:Link
      inet6 addr: fec0::9161:c32:a3e2:1c34/64 Scope:Link
      inet6 addr: fec0::9187:ce20:755c:1c34/64 Scope:Link
      inet6 addr: fec0::219c:7bf6:f916:3e4b/64 Scope:Link
      inet6 addr: fec0::852d:9c06:f163:2f3d/64 Scope:Link
      inet6 addr: fec0::49c5:df7d:4816:3e4b/64 Scope:Link
      inet6 addr: fec0::b4a3:9756:a616:3e4b/64 Scope:Link
      inet6 addr: fec0::d846:6324:c216:3e4b/64 Scope:Link
      inet6 addr: fe80::3a99:ffc4:d0bd:1eb0/64 scope link
      UP BROADCAST RUNNING MULTICAST
      RX packets:7127813 errors:0 dropped:0 overruns:0 frame:0
      TX packets:3639418 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:1000
      RX bytes:5972341828 (5.9 GB)
      TX bytes:2650657 (2.6 MB)

      Link encap:Local Loopback
      inet addr:127.0.0.1 Mask:255.0.0.0
      inet6 addr: ::1/128 Scope:Host
      UP LOOPBACK RUNNING MTU:65536
      RX packets:1427304 errors:0 dropped:0 overruns:0 frame:0
      TX packets:1427304 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:1000
      RX bytes:2650657 (2.6 MB)
      TX bytes:2650657 (2.6 MB)

cnlab@hw1b:~$ ifconfig
eth0: Link encap:Ethernet HWaddr 08:00:27:8f:69:8f
      inet addr:192.168.100.20 Bcast:192.168.100.255 Mask:255.255.255
      inet6 addr: fe80::7b34:782c:1bae:7286/64 Scope:Link
      UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
      RX packets:125494 errors:0 dropped:0 overruns:0 frame:0
      TX packets:34115 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:1000
      RX bytes:138233081 (138.2 MB) TX bytes:4632793 (4.6 MB)

lo: Link encap:Local Loopback
      inet addr:127.0.0.1 Mask:255.0.0.0
      inet6 addr: ::1/128 Scope:Host
      UP LOOPBACK RUNNING MTU:65536 Metric:1
      RX packets:22699 errors:0 dropped:0 overruns:0 frame:0
      TX packets:22699 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:1000
      RX bytes:2650657 (2.6 MB) TX bytes:2650657 (2.6 MB)

cnlab@hw1b:~$ ping 10.0.2.15
PING 10.0.2.15 (10.0.2.15) 56(84) bytes of data:
64 bytes from 10.0.2.15: icmp_seq=1 ttl=63 time=4.86 ms
64 bytes from 10.0.2.15: icmp_seq=2 ttl=63 time=5.21 ms
64 bytes from 10.0.2.15: icmp_seq=3 ttl=63 time=4.11 ms
^C
--- 10.0.2.15 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2014ms
rtt min/avg/max/mdev = 4.115/4.734/5.219/0.460 ms
cnlab@hw1b:~$
```

✅ FTP（可以認得ftp server，但1024以上隨機port擋掉）

No.	Time	Source	Destination	Protocol	Length	Info
237	18.069757513	90.130.70.73	192.168.100.20	FTP	74	Response: 220 (vsFTPd 3.0.3)
239	18.072111249	192.168.100.20	90.130.70.73	FTP	70	Request: USER anonymous
247	18.389565326	90.130.70.73	192.168.100.20	FTP	88	Response: 331 Please specify the password.
248	18.391251441	192.168.100.20	90.130.70.73	FTP	60	Request: QUIT
257	18.684685043	90.130.70.73	192.168.100.20	FTP	68	Response: 221 Goodbye.
911	87.832791030	90.130.70.73	192.168.100.20	FTP	74	Response: 220 (vsFTPd 3.0.3)
913	87.833952879	192.168.100.20	90.130.70.73	FTP	70	Request: USER anonymous
915	88.119886543	90.130.70.73	192.168.100.20	FTP	88	Response: 331 Please specify the password.
916	88.123711878	192.168.100.20	90.130.70.73	FTP	60	Request: QUIT
921	88.410189427	90.130.70.73	192.168.100.20	FTP	68	Response: 221 Goodbye.

```
cnlab@hw1b:~$ ftp speedtest.tele2.net
Connected to speedtest.tele2.net.
220 (vsFTPd 3.0.3)
Name (speedtest.tele2.net:cnlab): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

## ✔ DNS, SSH (result from demo.sh )

```
testing http:
curl http://www.google.com
HTTP/1.1 200 OK

testing https:
curl https://www.google.com
curl: (28) Connection timed out after 1002 milliseconds

testing dns:
nslookup google.com 8.8.8.8
Server:      8.8.8.8
Address:     8.8.8.8#53

Non-authoritative answer:
Name:   google.com
Address: 172.217.27.142

testing ftp:
ftp -n speedtest.tele2.net
Connected to speedtest.tele2.net.

testing ssh:
ssh -t 10.0.2.15 hostname
CNLab
Connection to 10.0.2.15 closed.

testing icmp:
ping -c 5 10.0.2.15
64 bytes from 10.0.2.15: icmp_seq=1 ttl=63 time=2.69 ms
64 bytes from 10.0.2.15: icmp_seq=2 ttl=63 time=4.22 ms
64 bytes from 10.0.2.15: icmp_seq=3 ttl=63 time=22.0 ms
64 bytes from 10.0.2.15: icmp_seq=4 ttl=63 time=4.97 ms
64 bytes from 10.0.2.15: icmp_seq=5 ttl=63 time=4.46 ms
5 packets transmitted, 5 received, 0% packet loss, time 4020ms

testing telnet:
telnet telehack.com
Connected to telehack.com.
```

# Applications

## 1. 公司/學校宿舍等公共網路管理

- 電腦A架設DHCP Server提供員工/學生的電腦B(s)使用上網
- 通過防火牆擋ports管理電腦B(s)可以使用的服務，例如不讓員工/學生打遊戲
- 如果想做權限劃分，可以讓不同權限群組連到電腦A的不同網卡，例如：
  - 電腦A eth0 聯外
  - 電腦A eth1 連電腦B1(s)，例如主管/老師，沒有擋port
  - 電腦A eth2 連電腦B2(s)，例如員工/學生，有擋port

## 2. 簡單過濾蠕蟲程式散布：透過對電腦B(s)之間的內網連線設定更嚴格的防火牆規則來達成

## 3. 設定流量配額限制

- 類似地通過修改filter表來達成（資料來源：[Blog - Yunfei](#))
  - `$ iptables -A OUTPUT -p tcp -m quota --quota 1024000 -j ACCEPT`
  - `$ iptables -A OUTPUT -p tcp -j DROP`

## Appendix: the firewall.sh file

```
# clean
sudo iptables -F
sudo iptables -X
sudo iptables -F -t nat
sudo iptables -X -t nat

# default drop
sudo iptables -t filter -P FORWARD DROP

# DNS
sudo iptables -A FORWARD -i eth0 -p tcp --sport 53 -j ACCEPT
sudo iptables -A FORWARD -o eth0 -p tcp --dport 53 -j ACCEPT
sudo iptables -A FORWARD -i eth0 -p udp --sport 53 -j ACCEPT
sudo iptables -A FORWARD -o eth0 -p udp --dport 53 -j ACCEPT

# http
sudo iptables -A FORWARD -i eth0 -p tcp --sport 80 -j ACCEPT
sudo iptables -A FORWARD -o eth0 -p tcp --dport 80 -j ACCEPT

# https
#sudo iptables -A FORWARD -i eth0 -p tcp --sport 443 -j ACCEPT
#sudo iptables -A FORWARD -o eth0 -p tcp --dport 443 -j ACCEPT

# FTP
sudo iptables -A FORWARD -i eth0 -p tcp --sport 20 -j ACCEPT
sudo iptables -A FORWARD -o eth0 -p tcp --dport 20 -j ACCEPT
sudo iptables -A FORWARD -i eth0 -p tcp --sport 21 -j ACCEPT
sudo iptables -A FORWARD -o eth0 -p tcp --dport 21 -j ACCEPT

# Telnet
sudo iptables -A FORWARD -i eth0 -p tcp --sport 23 -j ACCEPT
sudo iptables -A FORWARD -o eth0 -p tcp --dport 23 -j ACCEPT

# ICMP
sudo iptables -A FORWARD -i eth0 -p icmp -j ACCEPT
sudo iptables -A FORWARD -o eth0 -p icmp -j ACCEPT

# SSH
sudo iptables -A FORWARD -i eth0 -p tcp --sport 22 -j ACCEPT
sudo iptables -A FORWARD -o eth0 -p tcp --dport 22 -j ACCEPT

# nat
sudo iptables -t nat -A POSTROUTING -s 192.168.100.0/24 -o eth0 -j MASQUERADE

# forward
sudo sh -c "echo 1 > /proc/sys/net/ipv4/ip_forward"

# show info
sudo iptables -t filter -L -n
sudo iptables -t nat -L -n
```