

教育部 5G 行動寬頻人才培育跨校教學聯盟計畫  
5G 行動網路協定與核網技術聯盟中心示範課程

## 4G/5G 行動寬頻協同網路

### 實驗一 開源碼 4G/5G 平台 建置與基本量測

副教授：吳俊興  
助教：林原進、吳振宇

國立高雄大學 資訊工程學系

## 目錄

實驗一.....	1
一、 平台架構.....	4
1. 實驗架構.....	4
二、 軟硬體需求.....	5
1. 硬體.....	5
2. 軟體.....	5
三、 環境安裝.....	7
1. Linux Kernel 安裝.....	7
1.1. 安裝 Kernel .....	7
1.2. 安裝過程.....	7
1.3. 修改開機選單和設定.....	7
1.4. 更新 grub 設定 .....	8
1.5. 檢查 Kernel 版本 .....	8
2. Linux 基本指令 .....	9
2.1. cd.....	9
2.2. pwd.....	9
2.3. ls .....	9
2.4. mkdir .....	10
2.5. rm .....	10
2.6. mv.....	10
2.7. cp.....	11
2.8. find .....	11
2.9. grep .....	11
2.10. 檔案權限.....	11
2.11. chmod .....	12
2.12. chown.....	12
2.13. chgrp .....	12
2.14. ifconfig.....	12
2.15. netstat .....	13
2.16. nslookup.....	14
2.17. traceroute .....	14
2.18. route.....	15
2.19. ping.....	16
2.20. ps.....	16
2.21. kill .....	17
2.22. jobs.....	17

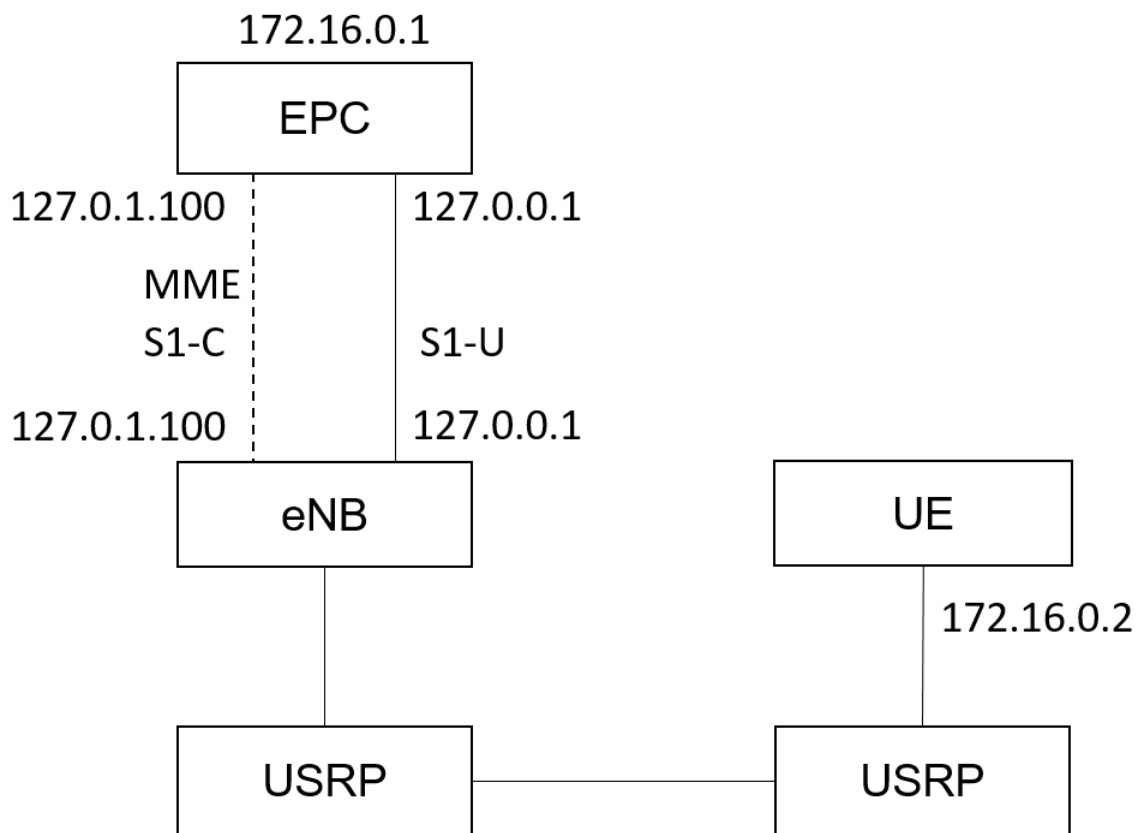
2.23.	fg.....	17
2.24.	bg.....	17
3.	安裝 Wireshark.....	18
3.1.	wireshark 無法正確抓取網卡.....	20
4.	安裝 srsLTE 及相關套件.....	20
4.1.	一般套件.....	20
4.2.	RF Front-end Driver.....	20
4.3.	mbd TLS.....	21
4.4.	srsGUI.....	21
4.5.	srsLTE.....	21
四、	執行 srsLTE.....	23
1.	EPC.....	23
2.	eNB.....	23
3.	UE.....	24
4.	檢查環境.....	26
4.1.	EPC 網卡.....	26
4.2.	UE 網卡.....	27
4.3.	EPC 程式-netstat.....	28
4.4.	偵測/重啟 USRP-B210.....	29
五、	測試.....	30
1.	互通測試.....	30
2.	Wireshark 介面查看.....	30



# 一、平台架構

## 1. 實驗架構

本實驗架構如圖所示，共分成兩個部分，第一部分由 EPC 和 eNB 組成，透過程式指定 ip 位址由同一台電腦開啟，並且藉由 USRP 達到實體層的效果，另一部分為 UE，同樣透過 USRP 實現實體層效果，並將兩片 USRP 透過 SMA 線對連，形成 4G 基本網路架構。



## 二、軟硬體需求

### 1. 硬體

名稱	規格	數量	目的
EPC+eNB	電腦型號： ASUS VivoMini UN65H	1	啟動 MME,S-GW,P-GW
	USRP B210	1	啟動 srsLTE eNB
UE	電腦型號： ASUS NB M580V	1	模擬 UE
	USRP B210	1	啟動 srsLTE UE

### 2. 軟體

名稱	軟體	版本	目的
EPC+ eNB	Ubuntu	Ubuntu 16.04	啟動 HSS, MME, S-GW, P-GW 功能
		Kernel: 4.15.0-041500-lowlatancy	
	srsLTE	srsLTE 19.03 5343b33f8ab2edf7319b6abb 07bbc3970541517a	
UE	Ubuntu	Ubuntu 16.04	啟動 UE 功 能
		Kernel: 4.15.0-041500-lowlatancy	
	srsLTE	srsLTE 19.03 5343b33f8ab2edf7319b6abb 07bbc3970541517a	
PC	Wireshark	2.6.8 (Ubuntu)	觀察協定
	Iperf	3.1.3 (Client) 3.1.3(Server)	測量網路 頻寬



## 三、環境安裝

### 1. Linux Kernel 安裝

#### 1.1. 安裝 Kernel

開啟一個終端機(Terminal)，並且依序輸入

wget	-P	~/Downloads/kernel
https://kernel.ubuntu.com/~kernel-ppa/mainline/v4.4.15/linux-headers-4.4.15-040415_4.4.15-040415.201607111333_all.deb		
wget	-P	~/Downloads/kernel
https://kernel.ubuntu.com/~kernel-ppa/mainline/v4.4.15/linux-headers-4.4.15-040415-lowlatency_4.4.15-040415.201607111333_amd64.deb		
wget	-P	~/Downloads/kernel
https://kernel.ubuntu.com/~kernel-ppa/mainline/v4.4.15/linux-image-4.4.15-040415-lowlatency_4.4.15-040415.201607111333_amd64.deb		

#### 1.2. 安裝過程

※請先確認是否連接網路正常

```
asus-medium@asusmedium-UN65H: ~  
asus-medium@asusmedium-UN65H:~$ wget -P ~/Downloads/kernel https://kernel.ubuntu.com/~kernel-ppa/mainline/v4.4.15/linux-headers-4.4.15-040415_4.4.15-040415.201607111333_all.deb  
--2019-07-12 10:23:21-- https://kernel.ubuntu.com/~kernel-ppa/mainline/v4.4.15/linux-headers-4.4.15-040415_4.4.15-040415.201607111333_all.deb  
Resolving kernel.ubuntu.com (kernel.ubuntu.com)... 91.189.94.216  
Connecting to kernel.ubuntu.com (kernel.ubuntu.com)|91.189.94.216|:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 9755644 (9.3M) [application/x-debian-package]  
Saving to: '/home/asus-medium/Downloads/kernel/linux-headers-4.4.15-040415_4.4.15-040415.201607111333_all.deb'  
linux-headers-4.4.15-0 100%[=====] 9.30M 1.62MB/s in 7.2s  
2019-07-12 10:23:29 (1.30 MB/s) - '/home/asus-medium/Downloads/kernel/linux-headers-4.4.15-040415_4.4.15-040415.201607111333_all.deb' saved [9755644/9755644]  
asus-medium@asusmedium-UN65H:~$ wget -P ~/Downloads/kernel https://kernel.ubuntu.com/~kernel-ppa/mainline/v4.4.15/linux-headers-4.4.15-040415-lowlatency_4.4.15-040415.201607111333_amd64.deb  
--2019-07-12 10:23:29-- https://kernel.ubuntu.com/~kernel-ppa/mainline/v4.4.15/linux-headers-4.4.15-040415-lowlatency_4.4.15-040415.201607111333_amd64.deb  
Resolving kernel.ubuntu.com (kernel.ubuntu.com)... 91.189.94.216  
Connecting to kernel.ubuntu.com (kernel.ubuntu.com)|91.189.94.216|:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 748090 (731K) [application/x-debian-package]  
Saving to: '/home/asus-medium/Downloads/kernel/linux-headers-4.4.15-040415-lowlatency_4.4.15-040415.201607111333_amd64.deb'  
linux-headers-4.4.15-0 100%[=====] 730.56K 505KB/s in 1.4s  
2019-07-12 10:23:32 (505 KB/s) - '/home/asus-medium/Downloads/kernel/linux-headers-4.4.15-040415-lowlatency_4.4.15-040415.201607111333_amd64.deb' saved [748090/748090]  
asus-medium@asusmedium-UN65H:~$ wget -P ~/Downloads/kernel https://kernel.ubuntu.com/~kernel-ppa/mainline/v4.4.15/linux-image-4.4.15-040415-lowlatency_4.4.15-040415.201607111333_amd64.deb
```

#### 1.3. 修改開機選單和設定

開啟終端機(Terminal)輸入以下指令

```
sudo gedit /etc/default/grub
```

找到下列文字

```
GRUB_HIDDEN_TIMEOUT=0
```

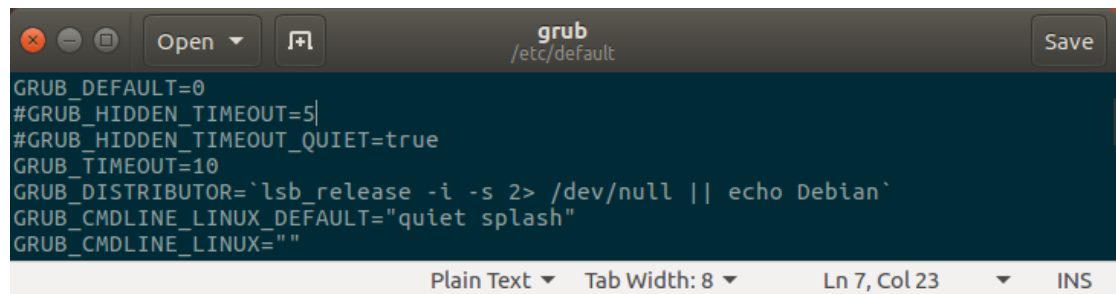
```
GRUB_HIDDEN_TIMEOUT_QUIET=true
```

改成

```
#GRUB_HIDDEN_TIMEOUT=0
```

```
#GRUB_HIDDEN_TIMEOUT_QUIET=true
```

儲存後離開



```
grub
/etc/default

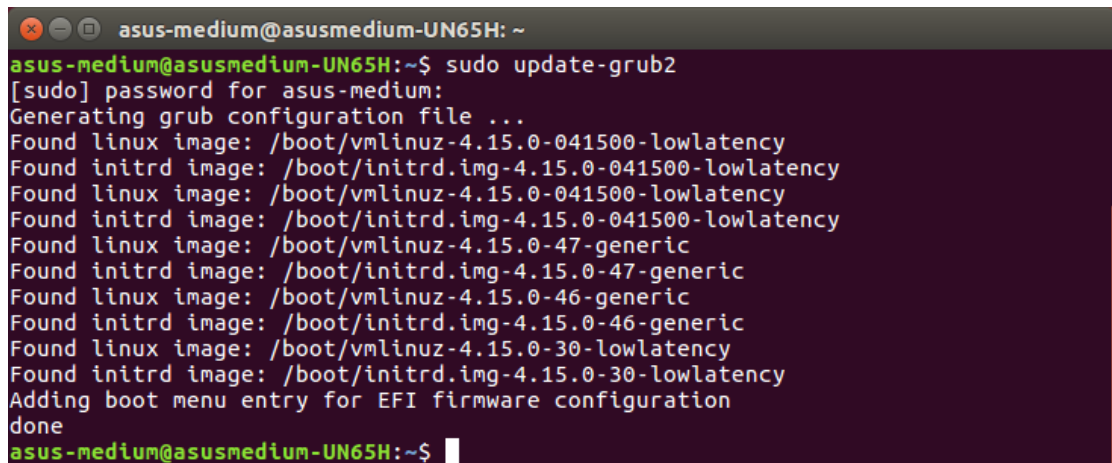
GRUB_DEFAULT=0
#GRUB_HIDDEN_TIMEOUT=5|
#GRUB_HIDDEN_TIMEOUT_QUIET=true
GRUB_TIMEOUT=10
GRUB_DISTRIBUTOR=`lsb_release -i -s 2> /dev/null || echo Debian`
GRUB_CMDLINE_LINUX_DEFAULT="quiet splash"
GRUB_CMDLINE_LINUX=""

Plain Text Tab Width: 8 Ln 7, Col 23 INS
```

## 1.4. 更新 grub 設定

開啟終端機並輸入以下指令

```
sudo update-grub2
```



```
asus-medium@asusmedium-UN65H: ~
asus-medium@asusmedium-UN65H:~$ sudo update-grub2
[sudo] password for asus-medium:
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-4.15.0-041500-lowlatency
Found initrd image: /boot/initrd.img-4.15.0-041500-lowlatency
Found linux image: /boot/vmlinuz-4.15.0-041500-lowlatency
Found initrd image: /boot/initrd.img-4.15.0-041500-lowlatency
Found linux image: /boot/vmlinuz-4.15.0-47-generic
Found initrd image: /boot/initrd.img-4.15.0-47-generic
Found linux image: /boot/vmlinuz-4.15.0-46-generic
Found initrd image: /boot/initrd.img-4.15.0-46-generic
Found linux image: /boot/vmlinuz-4.15.0-30-lowlatency
Found initrd image: /boot/initrd.img-4.15.0-30-lowlatency
Adding boot menu entry for EFI firmware configuration
done
asus-medium@asusmedium-UN65H:~$
```

接著輸入以下指令，重新啟動電腦

```
sudo reboot
```

然後在開機選單選擇剛才安裝的 lowlatency

## 1.5. 檢查 Kernel 版本

重新開機後在終端機輸入指令，確認版本

```
uname -r
```



```
asus-medium@asusmedium-UN65H: ~  
asus-medium@asusmedium-UN65H:~$ uname -r  
4.15.0-041500-lowlatency  
asus-medium@asusmedium-UN65H:~$
```

## 2. Linux 基本指令

在 Linux 環境，指令格式分三部分：[Command] [Options] [Arguments]，

例如：ping 127.0.0.1 -c 10。

選項(Options) 前面必須加“-”作為前導，多個選項可以合併，

例如：ls -a -l -t 等同 ls -alt。

系統對於指令大小寫是有區別的，對於指令、位址、檔案名稱，皆需注意是否有拼錯的情況。

### 2.1. cd

cd 是更改目前工作路徑的指令。

路徑有分絕對路徑及相對路徑，

絕對路徑是從根目錄開始：/etc/...，

相對路徑則是從目前開啟的路徑開始，

Ubuntu 開啟終端機預設就是從：/home/user/ 開始。

cd ..           #回到上一層

cd ~            #在 Ubuntu 預設是 /home/user\_name/

cd /usr/bin     #移動到指定路徑

```
asus-medium@asusmedium-UN65H: /usr/local  
asus-medium@asusmedium-UN65H:~$ cd /usr/local/  
asus-medium@asusmedium-UN65H:/usr/local$
```

### 2.2. pwd

pwd 是顯示目前的工作路徑。

```
asus-medium@asusmedium-UN65H: ~  
asus-medium@asusmedium-UN65H:~$ pwd  
/home/asus-medium  
asus-medium@asusmedium-UN65H:~$
```

### 2.3. ls

ls 是顯示檔案名稱與屬性的指令。

ls -al     #顯示該路徑下的全部檔案包括隱藏檔案以及詳細的屬性內容

ls --help #ls 指令的詳細用法

```

asus-medium@asusmedium-UN65H: ~
asus-medium@asusmedium-UN65H:~$ ls -al
total 212
drwxr-xr-x 27 asus-medium asus-medium 4096 七  4 14:59 .
drwxr-xr-x  3 root        root        4096  19 18:45 ..
-rw-r----- 1 asus-medium asus-medium 22657 七  4 16:20 .bash_history
-rw-r--r--  1 asus-medium asus-medium  220  19 18:45 .bash_logout
-rw-r--r--  1 asus-medium asus-medium  3771  19 18:45 .bashrc
drwx----- 22 asus-medium asus-medium 4096  七  1 15:35 .cache
drwx-----  3 asus-medium asus-medium 4096  23 16:18 .compiz
drwx----- 25 asus-medium asus-medium 4096  七  1 15:35 .config
drwx-----  3 root        root        4096  三 14 09:27 .dbus

```

## 2.4. mkdir

mkdir 是建立一個資料夾的指令。

mkdir [folder\_name] #建立名為[folder\_name]的資料夾

mkdir --help #mkdir 指令的詳細用法

```

asus-medium@asusmedium-UN65H: ~
asus-medium@asusmedium-UN65H:~$ ls
Desktop  Downloads  Music  Pictures  scitools  srsGUI  Understand_project
Documents examples.desktop Mytest  Public  SoapySDR  Templates Videos
asus-medium@asusmedium-UN65H:~$ mkdir Mydir
asus-medium@asusmedium-UN65H:~$ ls
Desktop  examples.desktop Mytest  scitools  Templates
Documents Music  Pictures SoapySDR  Understand_project
Downloads Mydir  Public  srsGUI  Videos
asus-medium@asusmedium-UN65H:~$

```

## 2.5. rm

rm 是移除的指令。

rm -r #連同目錄裡面包含的檔案一併刪除(recursive)

rm -i #在每次刪除檔案前，都會確認一次

rm -f #強制執行，並且不會確認

rm --help #rm 指令的詳細用法

```

asus-medium@asusmedium-UN65H: ~
asus-medium@asusmedium-UN65H:~$ ls
Desktop  examples.desktop Mytest  scitools  Templates
Documents Music  Pictures SoapySDR  Understand_project
Downloads Mydir  Public  srsGUI  Videos
asus-medium@asusmedium-UN65H:~$ rm -rf Mydir/
asus-medium@asusmedium-UN65H:~$ ls
Desktop  Downloads  Music  Pictures  scitools  srsGUI  Understand_project
Documents examples.desktop Mytest  Public  SoapySDR  Templates Videos
asus-medium@asusmedium-UN65H:~$

```

## 2.6. mv

mv 是移動檔案或是目錄的指令。

mv [file\_name] [dir\_path] #將檔案移動至指定目錄

mv [file\_name] [file\_rename] #重新命名檔案

mv --help #mv 指令的詳細用法

```

asus-medium@asusmedium-UN65H: ~/command_dir
asus-medium@asusmedium-UN65H:~$ ls
command_dir  Documents      Music          Public         srsGUI          Videos
command_test Downloads      Mytest         scitools       Templates
Desktop      examples.desktop Pictures        SoapySDR       Understand_project
asus-medium@asusmedium-UN65H:~$ mv command_test ./command_dir/
asus-medium@asusmedium-UN65H:~$ cd command_dir/
asus-medium@asusmedium-UN65H:~/command_dir$ ls
command_test
asus-medium@asusmedium-UN65H:~/command_dir$ mv command_test command_test_rename
asus-medium@asusmedium-UN65H:~/command_dir$ ls
command_test_rename
asus-medium@asusmedium-UN65H:~/command_dir$ █

```

## 2.7. cp

cp 是複製檔案的指令。

cp [file\_name] [path] #複製檔案至指定目錄

cp --help #cp 指令的詳細用法

```

asus-medium@asusmedium-UN65H: ~
asus-medium@asusmedium-UN65H:~/command_dir$ cp command_test_rename ~/
asus-medium@asusmedium-UN65H:~/command_dir$ cd ../
asus-medium@asusmedium-UN65H:~$ ls
command_dir  Documents      Music          Public         srsGUI          Videos
command_test_rename Downloads      Mytest         scitools       Templates
Desktop      examples.desktop Pictures        SoapySDR       Understand_project
asus-medium@asusmedium-UN65H:~$ █

```

## 2.8. find

find 是尋找檔案的指令。

find [path] [file\_name] #在指定目錄下尋找檔名為 file\_name 的檔案

find --help #find 指令的詳細用法

```

asus-medium@asusmedium-UN65H: ~/command_dir
asus-medium@asusmedium-UN65H:~/command_dir$ find ./ command_test_rename
./
./command_test_rename
command_test_rename
asus-medium@asusmedium-UN65H:~/command_dir$ █

```

## 2.9. grep

grep 是搜尋某些特定字元的指令，通常搭配其他指令使用。

grep [OPTION]... PATTERN [FILE]...

grep --help #grep 指令的詳細用法

```

asus-medium@asusmedium-UN65H: ~
asus-medium@asusmedium-UN65H:~$ grep "Hello" ./command_dir/command_test_rename
Hello test for command
asus-medium@asusmedium-UN65H:~$ ls | grep com
command_dir
command_test_rename
asus-medium@asusmedium-UN65H:~$ █

```

## 2.10. 檔案權限

在 Unix 系統中，共分成三種身分: owner/group/others，

並且各自擁有三種權限: read/write/execute，

檔案的屬性對應至使用者，決定可以做什麼。

#若是不具有該權限則會用 - 表示，

權限分數: r = 4 分, w = 2 分, x = 1 分, - = 0 分

## 2.11.chmod

更改檔案權限的指令，可分兩種方式設定

數字 chmod 740 [file\_name]

文字 chmod u/g/o/a +/-= r/w/x [file/folder]

```
asus-medium@asusmedium-UN65H: ~/command_dir
asus-medium@asusmedium-UN65H:~/command_dir$ ls -l
total 4
-rw-rw-r-- 1 asus-medium asus-medium 23  七  4 16:36 command_test_rename
asus-medium@asusmedium-UN65H:~/command_dir$ chmod 777 command_test_rename
asus-medium@asusmedium-UN65H:~/command_dir$ ls -l
total 4
-rwxrwxrwx 1 asus-medium asus-medium 23  七  4 16:36 command_test_rename
asus-medium@asusmedium-UN65H:~/command_dir$ chmod u=rw,g=r,o=- command_test_rename
asus-medium@asusmedium-UN65H:~/command_dir$ ls -l
total 4
-rw-r----- 1 asus-medium asus-medium 23  七  4 16:36 command_test_rename
asus-medium@asusmedium-UN65H:~/command_dir$
```

## 2.12.chown

更改檔案的擁有人及擁有群組的指令

chown owner:group [file]

```
asus-medium@asusmedium-UN65H: ~/command_dir
asus-medium@asusmedium-UN65H:~/command_dir$ sudo chown asus-medium:test1 command_test_r
ename
[sudo] password for asus-medium:
asus-medium@asusmedium-UN65H:~/command_dir$ ls -l
total 4
-rw-r----- 1 asus-medium test1 23  七  4 16:36 command_test_rename
asus-medium@asusmedium-UN65H:~/command_dir$
```

## 2.13.chgrp

改變檔案的擁有群組

chgrp group [file]

```
asus-medium@asusmedium-UN65H: ~/command_dir
asus-medium@asusmedium-UN65H:~/command_dir$ ls -l
total 4
-rw-r----- 1 asus-medium test1 23  七  4 16:36 command_test_rename
asus-medium@asusmedium-UN65H:~/command_dir$ sudo chgrp asus-medium command_test_rename
asus-medium@asusmedium-UN65H:~/command_dir$ ls -l
total 4
-rw-r----- 1 asus-medium asus-medium 23  七  4 16:36 command_test_rename
asus-medium@asusmedium-UN65H:~/command_dir$
```

## 2.14.ifconfig

顯示網路介面卡狀況的指令

ifconfig #顯示使用中的網卡參數

ifconfig -a #顯示全部的網卡，包含關閉的

```

asus-medium@asusmedium-UN65H: ~
asus-medium@asusmedium-UN65H:~$ ifconfig
dongle  Link encap:Ethernet  HWaddr 98:de:d0:13:9b:1a
        UP BROADCAST MULTICAST  MTU:1500  Metric:1
        RX packets:0 errors:0 dropped:0 overruns:0 frame:0
        TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

eth0    Link encap:Ethernet  HWaddr 78:24:af:04:55:03
        inet addr:192.168.128.101  Bcast:192.168.128.255  Mask:255.255.255.0
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
        RX packets:10009 errors:0 dropped:0 overruns:0 frame:0
        TX packets:3767 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:9801668 (9.8 MB)  TX bytes:292349 (292.3 KB)

```

ifconfig --help #ifconfig 指令的詳細用法

```

asus-medium@asusmedium-UN65H: ~
asus-medium@asusmedium-UN65H:~$ ifconfig --help
Usage:
  ifconfig [-a] [-v] [-s] <interface> [[<AF>] <address>]
  [add <address>[/<prefixlen>]]
  [del <address>[/<prefixlen>]]
  [[-]broadcast [<address>]] [[-]pointopoint [<address>]]
  [netmask <address>] [dstaddr <address>] [tunnel <address>]
  [outfill <NN>] [keepalive <NN>]
  [hw <HW> <address>] [metric <NN>] [mtu <NN>]
  [[-]trailers] [[-]arp] [[-]allmulti]
  [multicast] [[-]promisc]
  [mem_start <NN>] [io_addr <NN>] [irq <NN>] [media <type>]
  [txqueuelen <NN>]
  [[-]dynamic]
  [up|down] ...

<HW>=Hardware Type.
List of possible hardware types:
  loop (Local Loopback) slip (Serial Line IP) cslip (VJ Serial Line IP)
  slip6 (6-bit Serial Line IP) cslip6 (VJ 6-bit Serial Line IP) adaptive (Adap
tive Serial Line IP)
  ash (Ash) ether (Ethernet) ax25 (AMPR AX.25)
  netrom (AMPR NET/ROM) rose (AMPR ROSE) tunnel (IPIP Tunnel)
  ppp (Point-to-Point Protocol) hdlc ((Cisco)-HDLC) lapb (LAPB)
  arcnet (ARCnet) dlci (Frame Relay DLCI) frad (Frame Relay Access Device)
  sit (IPv6-in-IPv4) fddi (Fiber Distributed Data Interface) hippi (HIPPI)
  irda (IrLAP) ec (Econet) x25 (generic X.25)
  eui64 (Generic EUI-64)
<AF>=Address family. Default: inet
List of possible address families:
  unix (UNIX Domain) inet (DARPA Internet) inet6 (IPv6)
  ax25 (AMPR AX.25) netrom (AMPR NET/ROM) rose (AMPR ROSE)
  ipx (Novell IPX) ddp (Appletalk DDP) ec (Econet)
  ash (Ash) x25 (CCITT X.25)
asus-medium@asusmedium-UN65H:~$

```

## 2.15.netstat

顯示 Linux 網路系統的詳細資訊

netstat -a #顯示所有開啟的 Socket

netstat -p #顯示程式名稱

netstat -s #顯示每個協定的統計結果

```
asus-medium@asusmedium-UN65H: ~  
asus-medium@asusmedium-UN65H:~$ netstat -s  
Ip:  
 9778 total packets received  
 2 with invalid addresses  
 0 forwarded  
 0 incoming packets discarded  
9775 incoming packets delivered  
4234 requests sent out  
40 outgoing packets dropped
```

netstat --help #netstat 指令的更詳細用法

```
ue@ue-X580VD:~$ netstat --help  
usage: netstat [-vWeenNcCF] [<Af>] -r netstat {-V|--version|-h|--help}  
       netstat [-vWnNcaeol] [<Socket> ...]  
       netstat { [-vWeenNac] -i | [-cWnNe] -M | -s }  
  
-r, --route display routing table  
-i, --interfaces display interface table  
-g, --groups display multicast group memberships  
-s, --statistics display networking statistics (like SNMP)  
-M, --masquerade display masqueraded connections  
  
-v, --verbose be verbose  
-W, --wide don't truncate IP addresses  
-n, --numeric don't resolve names  
--numeric-hosts don't resolve host names  
--numeric-ports don't resolve port names  
--numeric-users don't resolve user names  
-N, --symbolic resolve hardware names  
-e, --extend display other/more information  
-p, --programs display PID/Program name for sockets  
-c, --continuous continuous listing  
  
-l, --listening display listening server sockets  
-a, --all, --listening display all sockets (default: connected)  
-o, --timers display timers  
-F, --fib display Forwarding Information Base (default)  
-C, --cache display routing cache instead of FIB  
  
<Socket>={-t|--tcp} {-u|--udp} {-w|--raw} {-x|--unix} --ax25 --ipx --netrom  
<AF>=Use '-6|-4' or '-A <af>' or '--<af>'; default: inet  
List of possible address families (which support routing):  
inet (DARPA Internet) inet6 (IPv6) ax25 (AMPR AX.25)  
netrom (AMPR NET/ROM) ipx (Novell IPX) ddp (Appletalk DDP)  
x25 (CCITT X.25)  
ue@ue-X580VD:~$
```

## 2.16.nslookup

查詢 DNS 的指令

nslookup 網址

```
asus-medium@asusmedium-UN65H: ~  
asus-medium@asusmedium-UN65H:~$ nslookup www.google.com  
Server: 127.0.1.1  
Address: 127.0.1.1#53  
  
Non-authoritative answer:  
Name: www.google.com  
Address: 172.217.24.4  
asus-medium@asusmedium-UN65H:~$
```

## 2.17.traceroute

追蹤封包流向



traceroute dst

```
asus-medium@asusmedium-UN65H: ~  
asus-medium@asusmedium-UN65H:~$ traceroute www.nuk.edu.tw  
traceroute to www.nuk.edu.tw (140.127.234.77), 30 hops max, 60 byte packets  
 1  192.168.128.1 (192.168.128.1)  0.168 ms  0.202 ms  0.236 ms  
 2  10.1.208.254 (10.1.208.254)  16.947 ms  17.253 ms  17.594 ms  
 3  192.168.1.254 (192.168.1.254)  2.094 ms  2.231 ms  2.258 ms  
 4  192.168.249.253 (192.168.249.253)  0.773 ms  0.768 ms  0.776 ms
```

traceroute --help #traceroute 指令的詳細用法

```
asus-medium@asusmedium-UN65H: ~  
asus-medium@asusmedium-UN65H:~$ traceroute --help  
Usage:  
  traceroute [ -4dFITnreAUDV ] [ -f first_ttl ] [ -g gate,... ] [ -i device ] [ -m max_ttl ] [ -N squeries ] [ -p port ] [ -t tos ] [ -l flow_label ] [ -w wait time ] [ -q nqueries ] [ -s src_addr ] [ -z sendwait ] [ --fwmark=num ] host [ p  
acketlen ]  
Options:  
  -4                                Use IPv4  
  -6                                Use IPv6  
  -d --debug                        Enable socket level debugging  
  -F --dont-fragment               Do not fragment packets  
  -f first_ttl --first=first_ttl    Start from the first_ttl hop (instead from 1)  
  -g gate,... --gateway=gate,...    Route packets through the specified gateway  
                                     (maximum 8 for IPv4 and 127 for IPv6)  
  -I --icmp                        Use ICMP ECHO for tracerouting  
  -T --tcp                         Use TCP SYN for tracerouting (default port is 80)  
  -i device --interface=device      Specify a network interface to operate with  
  -m max_ttl --max-hops=max_ttl     Set the max number of hops (max TTL to be  
                                     reached). Default is 30  
  -N squeries --sim-queries=squeries Set the number of probes to be tried  
                                     simultaneously (default is 16)  
  -n                                Do not resolve IP addresses to their domain names  
  -p port --port=port               Set the destination port to use. It is either  
                                     initial udp port value for "default" method  
                                     (incremented by each probe, default is 33434), or  
                                     initial seq for "icmp" (incremented as well,  
                                     default from 1), or some constant destination  
                                     port for other methods (with default of 80 for  
                                     "tcp", 53 for "udp", etc.)  
  -t tos --tos=tos                 Set the TOS (IPv4 type of service) or TC (IPv6
```

## 2.18.route

顯示目前的 Routing table

route -n #不會自動將 ip 位址轉成文字

```
asus-medium@asusmedium-UN65H: ~  
asus-medium@asusmedium-UN65H:~$ route  
Kernel IP routing table  
Destination Gateway Genmask Flags Metric Ref Use Iface  
default 192.168.128.1 0.0.0.0 UG 100 0 0 eth0  
link-local * 255.255.0.0 U 1000 0 0 eth0  
192.168.128.0 * 255.255.255.0 U 100 0 0 eth0  
asus-medium@asusmedium-UN65H:~$
```

route --help #route 指令的詳細用法

```

ue@ue-X580VD:~$ route --help
Usage: route [-nNvee] [-FC] [<AF>]          List kernel routing tables
       route [-v] [-FC] {add|del|flush} ...  Modify routing table for AF.

       route {-h|--help} [<AF>]             Detailed usage syntax for specified AF.
       route {-V|--version}                 Display version/author and exit.

       -v, --verbose                        be verbose
       -n, --numeric                        don't resolve names
       -e, --extend                         display other/more information
       -F, --fib                           display Forwarding Information Base (default)
       -C, --cache                         display routing cache instead of FIB

<AF>=Use '-A <af>' or '--<af>'; default: inet
List of possible address families (which support routing):
inet (DARPA Internet) inet6 (IPv6) ax25 (AMPR AX.25)
netrom (AMPR NET/ROM) ipx (Novell IPX) ddp (Appletalk DDP)
x25 (CCITT X.25)
ue@ue-X580VD:~$

```

## 2.19.ping

ps 利用 ICMP 協定，發出 ECHO\_REQUEST 到目的地

ping dst -c 10 #發出 10 個封包

```

asus-medium@asusmedium-UN65H: ~
asus-medium@asusmedium-UN65H:~$ ping 192.168.128.1 -c 10
PING 192.168.128.1 (192.168.128.1) 56(84) bytes of data.
64 bytes from 192.168.128.1: icmp_seq=1 ttl=64 time=0.253 ms
64 bytes from 192.168.128.1: icmp_seq=2 ttl=64 time=0.217 ms
64 bytes from 192.168.128.1: icmp_seq=3 ttl=64 time=0.231 ms
64 bytes from 192.168.128.1: icmp_seq=4 ttl=64 time=0.224 ms
64 bytes from 192.168.128.1: icmp_seq=5 ttl=64 time=0.256 ms
64 bytes from 192.168.128.1: icmp_seq=6 ttl=64 time=0.218 ms
64 bytes from 192.168.128.1: icmp_seq=7 ttl=64 time=0.226 ms
64 bytes from 192.168.128.1: icmp_seq=8 ttl=64 time=0.223 ms
64 bytes from 192.168.128.1: icmp_seq=9 ttl=64 time=0.221 ms
64 bytes from 192.168.128.1: icmp_seq=10 ttl=64 time=0.235 ms

--- 192.168.128.1 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9218ms
rtt min/avg/max/mdev = 0.217/0.230/0.256/0.018 ms
asus-medium@asusmedium-UN65H:~$

```

ping --help #ping 指令的詳細用法

```

ue@ue-X580VD:~$ ping --help
ping: invalid option -- '-'
Usage: ping [-aAbBdDfhLnOqrRUvV] [-c count] [-i interval] [-I interface]
          [-m mark] [-M pmtudisc_option] [-l preload] [-p pattern] [-Q tos]
          [-s packetsize] [-S sndbuf] [-t ttl] [-T timestamp_option]
          [-w deadline] [-W timeout] [hop1 ...] destination
ue@ue-X580VD:~$

```

## 2.20.ps

查看執行中的程式指令

ps -l #查看目前 bash 的程序

ps aux #查看所有運行中的程序



```
asus-medium@asusmedium-UN65H: ~
asus-medium@asusmedium-UN65H:~$ ps -l
F S  UID  PID  PPID  C PRI  NI ADDR SZ WCHAN  TTY  TIME CMD
0 S  1000  7572  6855  0  80   0  -  5823 wait  pts/4  00:00:00 bash
0 R  1000  7769  7572  0  80   0  -  7379 -  pts/4  00:00:00 ps
asus-medium@asusmedium-UN65H:~$
```

## 2.21.kill

結束執行中的程式的指令

kill [PID]

```
asus-medium@asusmedium-UN65H:~$ ps
PID TTY  TIME CMD
6859 pts/18  00:00:00 bash
7871 pts/18  00:00:00 ps
asus-medium@asusmedium-UN65H:~$ kill 78603
asus-medium@asusmedium-UN65H:~$ ps
PID TTY  TIME CMD
7860 pts/4  00:00:00 bash
7872 pts/4  00:00:00 ps
asus-medium@asusmedium-UN65H:~$
```

## 2.22.jobs

查看目前背景有哪些工作

jobs #顯示工作狀態

jobs -l #顯示工作狀態外，同時顯示 PID

```
Terminal
asus-medium@asusmedium-UN65H: ~
asus-medium@asusmedium-UN65H:~$ jobs
asus-medium@asusmedium-UN65H:~$ gedit test
^Z
[1]+  Stopped                  gedit test
asus-medium@asusmedium-UN65H:~$ jobs
[1]+  Stopped                  gedit test
asus-medium@asusmedium-UN65H:~$
```

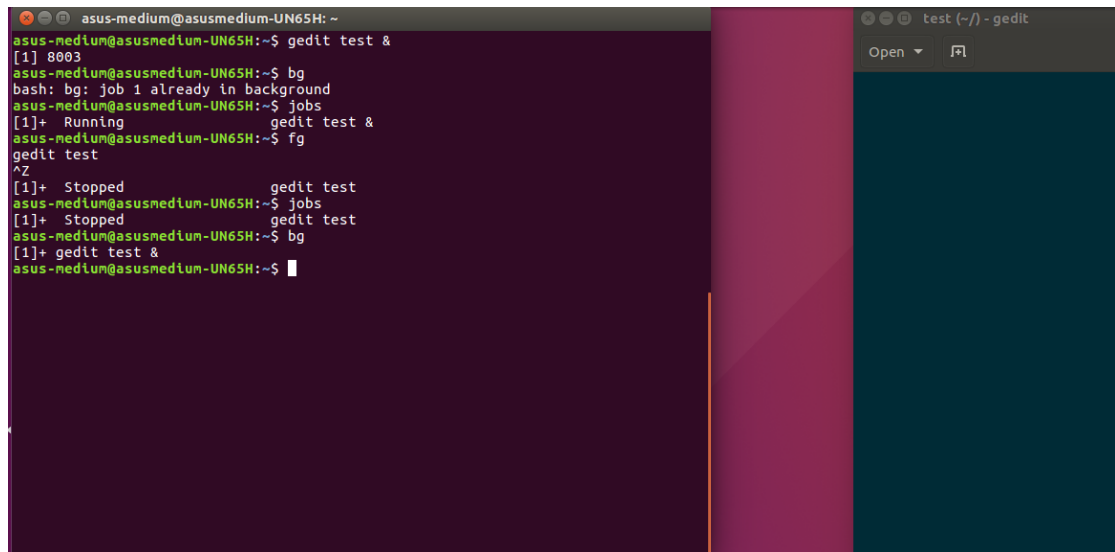
## 2.23.fg

將背景工作拿至前景處理

```
asus-medium@asusmedium-UN65H: ~
asus-medium@asusmedium-UN65H:~$ jobs
asus-medium@asusmedium-UN65H:~$ gedit test
^Z
[1]+  Stopped                  gedit test
asus-medium@asusmedium-UN65H:~$ jobs
[1]+  Stopped                  gedit test
asus-medium@asusmedium-UN65H:~$ fg
gedit test
```

## 2.24.bg

將目前執行的工作放置背景執行，效果等同指令後面加上”&”

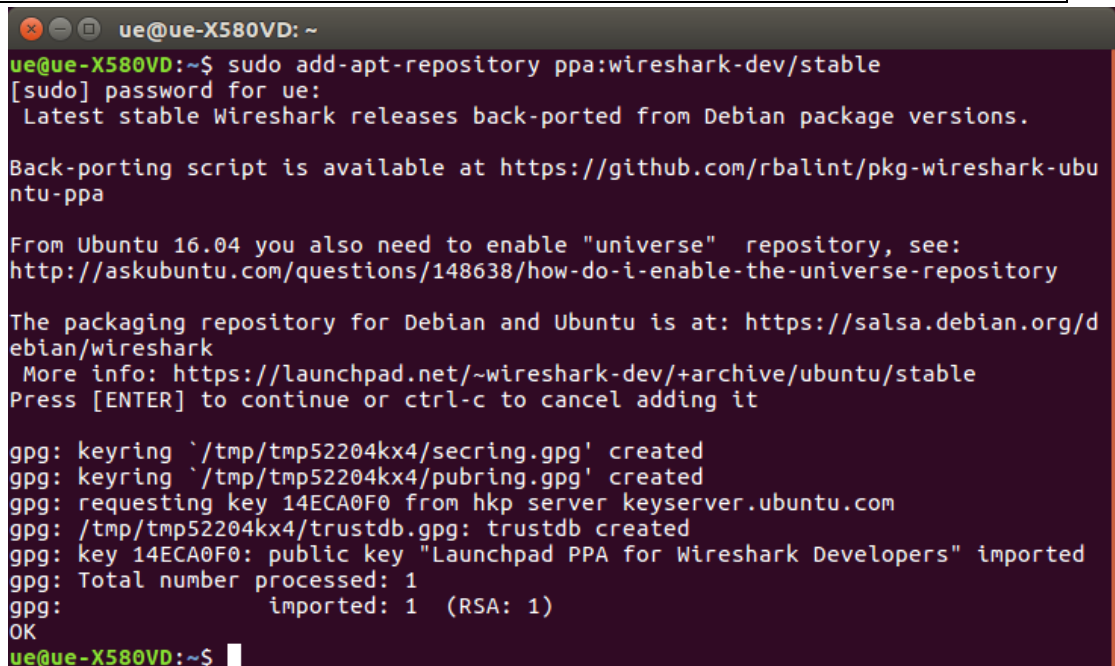


```
asus-medium@asusmedium-UN65H: ~  
asus-medium@asusmedium-UN65H:~$ gedit test &  
[1] 8003  
asus-medium@asusmedium-UN65H:~$ bg  
bash: bg: job 1 already in background  
asus-medium@asusmedium-UN65H:~$ jobs  
[1]+  Running                  gedit test &  
asus-medium@asusmedium-UN65H:~$ fg  
gedit test  
^Z  
[1]+  Stopped                  gedit test  
asus-medium@asusmedium-UN65H:~$ jobs  
[1]+  Stopped                  gedit test  
asus-medium@asusmedium-UN65H:~$ bg  
[1]+ gedit test &  
asus-medium@asusmedium-UN65H:~$
```

### 3. 安裝 Wireshark

開啟終端機並且依序輸入指令

```
sudo add-apt-repository ppa:wireshark-dev/stable
```



```
ue@ue-X580VD: ~  
ue@ue-X580VD:~$ sudo add-apt-repository ppa:wireshark-dev/stable  
[sudo] password for ue:  
Latest stable Wireshark releases back-ported from Debian package versions.  
  
Back-porting script is available at https://github.com/rbalint/pkg-wireshark-ubuntu-ppa  
  
From Ubuntu 16.04 you also need to enable "universe" repository, see:  
http://askubuntu.com/questions/148638/how-do-i-enable-the-universe-repository  
  
The packaging repository for Debian and Ubuntu is at: https://salsa.debian.org/debian/wireshark  
More info: https://launchpad.net/~wireshark-dev/+archive/ubuntu/stable  
Press [ENTER] to continue or ctrl-c to cancel adding it  
  
gpg: keyring `/tmp/tmp52204kx4/secring.gpg' created  
gpg: keyring `/tmp/tmp52204kx4/pubring.gpg' created  
gpg: requesting key 14ECA0F0 from hkp server keyserver.ubuntu.com  
gpg: /tmp/tmp52204kx4/trustdb.gpg: trustdb created  
gpg: key 14ECA0F0: public key "Launchpad PPA for Wireshark Developers" imported  
gpg: Total number processed: 1  
gpg: imported: 1 (RSA: 1)  
OK  
ue@ue-X580VD:~$
```

```
sudo apt-get update
```

```

ue@ue-X580VD: ~
Get:27 http://tw.archive.ubuntu.com/ubuntu xenial-backports/main amd64 DEP-11 Me
tadata [3328 B]
Get:28 http://tw.archive.ubuntu.com/ubuntu xenial-backports/universe amd64 DEP-1
1 Metadata [5104 B]
Get:29 http://security.ubuntu.com/ubuntu xenial-security/main amd64 Packages [70
0 kB]
Get:30 http://security.ubuntu.com/ubuntu xenial-security/main i386 Packages [572
kB]
Get:31 http://security.ubuntu.com/ubuntu xenial-security/main Translation-en [27
9 kB]
Get:32 http://security.ubuntu.com/ubuntu xenial-security/main amd64 DEP-11 Metad
ata [73.9 kB]
Get:33 http://security.ubuntu.com/ubuntu xenial-security/main DEP-11 64x64 Icons
[73.2 kB]
Get:34 http://security.ubuntu.com/ubuntu xenial-security/universe amd64 DEP-11 M
etadata [121 kB]
Get:35 http://security.ubuntu.com/ubuntu xenial-security/universe DEP-11 64x64 I
cons [179 kB]
Get:36 http://security.ubuntu.com/ubuntu xenial-security/multiverse amd64 DEP-11
Metadata [2464 B]
Fetched 7566 kB in 4s (1683 kB/s)
Reading package lists... Done
ue@ue-X580VD:~$

```

sudo apt-get install wireshark

```

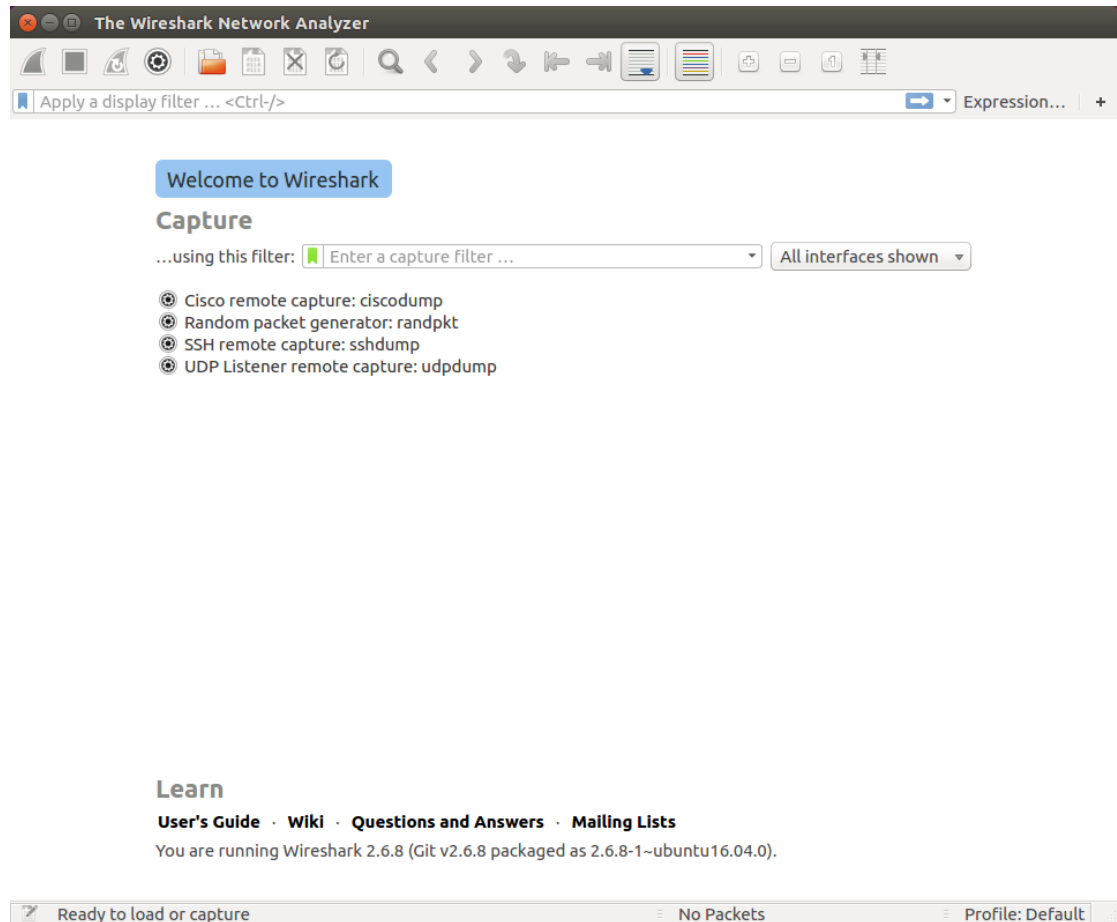
ue@ue-X580VD: ~
ue@ue-X580VD:~$ sudo apt-get install wireshark
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libllvm5.0 linux-headers-4.13.0-36 linux-headers-4.13.0-36-generic
  linux-headers-4.15.0-46 linux-headers-4.15.0-46-generic
  linux-headers-4.15.0-47 linux-headers-4.15.0-47-generic
  linux-headers-4.4.0-143 linux-headers-4.4.0-143-generic
  linux-headers-4.4.0-148 linux-headers-4.4.0-148-generic
  linux-image-4.13.0-36-generic linux-image-4.15.0-46-generic
  linux-image-4.15.0-47-generic linux-image-4.4.0-143-generic
  linux-image-4.4.0-148-generic linux-image-extra-4.13.0-36-generic
  linux-modules-4.15.0-46-generic linux-modules-4.15.0-47-generic
  linux-modules-4.4.0-143-generic linux-modules-4.4.0-148-generic
  linux-modules-extra-4.15.0-47-generic linux-modules-extra-4.4.0-148-generic
  snapd-login-service
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  wireshark-gtk wireshark-qt
The following NEW packages will be installed:
  wireshark wireshark-gtk wireshark-qt
0 upgraded, 3 newly installed, 0 to remove and 27 not upgraded.
Need to get 4201 kB of archives.
After this operation, 10.7 MB of additional disk space will be used.
Do you want to continue? [Y/n]

```

輸入“y”後按下 enter，接著繼續

安裝完成後，重新登入，並開啟 wireshark，

若是無法正確顯示網路介面卡資訊，請按照以下步驟解決



### 3.1. wireshark 無法正確抓取網卡

開啟終端機並輸入

```
sudo adduser $USER wireshark
```

接著登出後，在開啟 Wireshark

## 4. 安裝 srsLTE 及相關套件

### 4.1. 一般套件

開啟終端機並輸入

```
sudo apt-get install cmake libfftw3-dev libboost-all-dev libconfig++-dev  
libsctp-dev
```

### 4.2. RF Front-end Driver

開啟終端機並依序輸入

```
sudo add-apt-repository ppa:ettusresearch/uhd
```

```
sudo apt-get update
```

```
sudo apt-get install libuhd-dev libuhd003 uhd-host
```

<code>python3 /lib/uhd/uhd_images_downloader.py</code>
--

#### 4.3. mbed TLS

開啟終端機並依序輸入

<code>wget https://tls.mbed.org/download/start/mbedtls-2.16.0-apache.tgz</code>
<code>tar zxvf mbedtls-2.16.0-apache.tgz</code>
<code>sudo mv ~/Download/mbedtls-2.16.0 /usr/local</code>
<code>cd /usr/local/mbedtls-2.16.0</code>
<code>cmake .</code>
<code>make</code>
<code>make test</code>
<code>cmake -DENABLE_TESTING=Off .</code>
<code>cmake -DUSE_SHARED_MBEDTLS_LIBRARY=On .</code>
<code>sudo make install library</code>

#### 4.4. srsGUI

開啟終端機並依序輸入

<code>sudo apt-get install libboost-system-dev libboost-test-dev libboost-thread-dev libqwt-dev libqt4-dev</code>
<code>git clone https://github.com/srsLTE/srsGUI.git</code>
<code>cd ~/srsgui</code>
<code>mkdir build</code>
<code>cd build</code>
<code>cmake ../</code>
<code>make</code>
<code>make test</code>

#### 4.5. srsLTE

開啟終端機並依序輸入

<code>git clone https://github.com/srsLTE/srsLTE.git</code>
<code>cd srsLTE</code>
<code>mkdir build</code>
<code>cd build</code>
<code>cmake ../</code>
<code>make</code>
<code>make test</code>
<code>sudo make install</code>

```
sudo srslte_install_config.sh [user | service]
```

選擇 user option 會將全部 config 檔安裝至 \$HOME/.config/srslte

選擇 service option 會將全部 config 檔安裝至 /etc/srslte

## 四、執行 srsLTE

srsLTE 執行順序為 EPC->eNB->UE，USRP-B210 需用 USB 3.0 接口，若是使用 USB 2.0 可能會導致無法讀取。

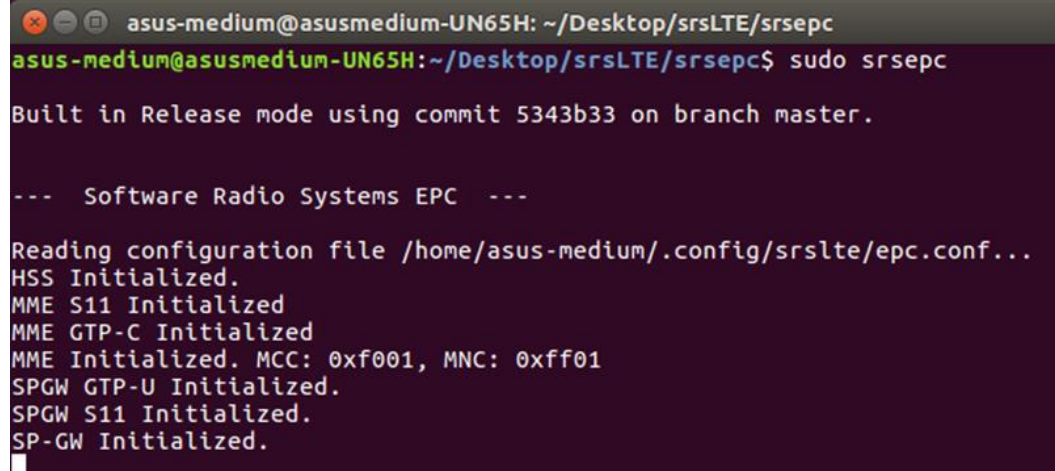
在本實驗中 EPC 跟 eNB 是使用同一台電腦執行，UE 使用另一台電腦。

### 1. EPC

開啟終端機並依序輸入

<code>cd ~/srsLTE/srsepc</code>
---------------------------------

<code>sudo srsepc</code>
--------------------------



```
asus-medium@asusmedium-UN65H: ~/Desktop/srsLTE/srsepc
asus-medium@asusmedium-UN65H:~/Desktop/srsLTE/srsepc$ sudo srsepc

Built in Release mode using commit 5343b33 on branch master.

--- Software Radio Systems EPC ---

Reading configuration file /home/asus-medium/.config/srslte/epc.conf...
HSS Initialized.
MME S11 Initialized
MME GTP-C Initialized
MME Initialized. MCC: 0xf001, MNC: 0xff01
SPGW GTP-U Initialized.
SPGW S11 Initialized.
SP-GW Initialized.
```

### 2. eNB

開啟新的終端機並依序輸入

<code>cd ~/srsLTE/srsenb</code>
---------------------------------

<code>sudo srsenb</code>
--------------------------

```
asus-medium@asusmedium-UN65H: ~/Desktop/srsLTE/srsenb
asus-medium@asusmedium-UN65H:~/Desktop/srsLTE/srsenb$ sudo srsenb

Built in Release mode using commit 5343b33 on branch master.

--- Software Radio Systems LTE eNodeB ---

Reading configuration file /home/asus-medium/.config/srslte/enb.conf...
[INFO] [UHD] linux; GNU C++ version 5.4.0 20160609; Boost_105800; UHD_3.14.0.0-release
[INFO] [LOGGING] Fastpath logging disabled at runtime.
Opening USRP with args: type=b200,master_clock_rate=30.72e6
[INFO] [B200] Detected Device: B210
[INFO] [B200] Operating over USB 3.
[INFO] [B200] Initialize CODEC control...
[INFO] [B200] Initialize Radio control...
[INFO] [B200] Performing register loopback test...
[INFO] [B200] Register loopback test passed
[INFO] [B200] Performing register loopback test...
[INFO] [B200] Register loopback test passed
[INFO] [B200] Asking for clock rate 30.720000 MHz...
[INFO] [B200] Actually got clock rate 30.720000 MHz.
Setting frequency: DL=2685.0 Mhz, UL=2565.0 Mhz
[INFO] [B200] Asking for clock rate 11.520000 MHz...
[INFO] [B200] Actually got clock rate 11.520000 MHz.
Setting Sampling frequency 11.52 MHz

==== eNodeB started ====
Type <t> to view trace
```

當 eNB 成功啟動後，EPC 會接著顯示 S1 連線的相關資訊

```
asus-medium@asusmedium-UN65H: ~/Desktop/srsLTE/srsepc

--- exiting ---
asus-medium@asusmedium-UN65H:~/Desktop/srsLTE/srsepc$ clear

asus-medium@asusmedium-UN65H:~/Desktop/srsLTE/srsepc$ sudo srsepc

Built in Release mode using commit 5343b33 on branch master.

--- Software Radio Systems EPC ---

Reading configuration file /home/asus-medium/.config/srslte/epc.conf...
HSS Initialized.
MME S11 Initialized
MME GTP-C Initialized
MME Initialized. MCC: 0xf001, MNC: 0xff01
SPGW GTP-U Initialized.
SPGW S11 Initialized.
SP-GW Initialized.
Received S1 Setup Request.
S1 Setup Request - eNB Name: srsenb01, eNB id: 0x19b
S1 Setup Request - MCC:001, MNC:01, PLMN: 61712
S1 Setup Request - TAC 7, B-PLMN 0
S1 Setup Request - Paging DRX 2
Sending S1 Setup Response
```

### 3. UE

開啟終端機並依序輸入

```
cd ~/srsLTE/srsue
```



sudo srsue

```
ue@ue-X580VD: ~/Desktop/srsLTE/srsue
ue@ue-X580VD:~/Desktop/srsLTE/srsue$ sudo srsue
Reading configuration file /home/ue/.config/srslte/ue.conf...

Built in Release mode using commit 5343b33 on branch master.

--- Software Radio Systems LTE UE ---

Opening 1 RF devices with 1 RF channels...
[INFO] [UHD] linux; GNU C++ version 5.4.0 20160609; Boost_105800; UHD_3.14.0.0-release
[INFO] [LOGGING] Fastpath logging disabled at runtime.
Opening USRP with args: type=b200,master_clock_rate=30.72e6
[INFO] [B200] Detected Device: B210
[INFO] [B200] Operating over USB 3.
[INFO] [B200] Initialize CODEC control...
[INFO] [B200] Initialize Radio control...
[INFO] [B200] Performing register loopback test...
[INFO] [B200] Register loopback test passed
[INFO] [B200] Performing register loopback test...
[INFO] [B200] Register loopback test passed
[INFO] [B200] Asking for clock rate 30.720000 MHz...
[INFO] [B200] Actually got clock rate 30.720000 MHz.
Waiting PHY to initialize...
...
Attaching UE...
Searching cell in DL EARFCN=3400, f_dl=2685.0 MHz, f_ul=2565.0 MHz
.
Found Cell: Mode=FDD, PCI=1, PRB=50, Ports=1, CFO=0.7 KHz
[INFO] [B200] Asking for clock rate 11.520000 MHz...
[INFO] [B200] Actually got clock rate 11.520000 MHz.
Found PLMN: Id=00101, TAC=7
Random Access Transmission: seq=8, ra-rnti=0x2
RRC Connected
Random Access Complete. c-rnti=0x46, ta=1
Network attach successful. IP: 172.16.0.2
Software Radio Systems LTE (srsLTE)
```

當 UE 成功連線後，EPC 及 eNB 接會顯示相關的連線資訊

eNB :

```
==== eNodeB started ====
Type <t> to view trace
RACH: tti=8341, preamble=6, offset=1, temp_crnti=0x46
User 0x46 connected
```

EPC :

```

asus-medium@asusmedium-UN65H: ~/Desktop/srsLTE/srsepc
Sending S1 Setup Response
Initial UE message: LIBLTE_MME_MSG_TYPE_ATTACH_REQUEST
Received Initial UE message -- Attach Request
Attach request -- GUTI Style Attach request
Attach request -- M-TMSI: 0x2f9f5a6b
Attach request -- eNB-UE S1AP Id: 1
Attach request -- Attach type: 1
Attach Request -- UE Network Capabilities EEA: 11100000
Attach Request -- UE Network Capabilities EIA: 01100000
Attach Request -- MS Network Capabilities Present: false
PDN Connectivity Request -- EPS Bearer Identity requested: 0
PDN Connectivity Request -- Procedure Transaction Id: 1
PDN Connectivity Request -- ESM Information Transfer requested: false
UL NAS: Received Identity Response
ID Response -- IMSI: 001010123456789
Downlink NAS: Sent Authentication Request
UL NAS: Received Authentication Response
Authentication Response -- IMSI 001010123456789
UE Authentication Accepted.
Generating KeNB with UL NAS COUNT: 0
Downlink NAS: Sending NAS Security Mode Command.
UL NAS: Received Security Mode Complete
Security Mode Command Complete -- IMSI: 001010123456789
Getting subscription information -- QCI 7
Sending Create Session Request.
Creating Session Response -- IMSI: 1010123456789
Creating Session Response -- MME control TEID: 1
Received GTP-C PDU. Message type: GTPC_MSG_TYPE_CREATE_SESSION_REQUEST
SPGW: Allocated Ctrl TEID 1
SPGW: Allocated User TEID 1
SPGW: Allocate UE IP 172.16.0.2
Received Create Session Response
Create Session Response -- SPGW control TEID 1
Create Session Response -- SPGW S1-U Address: 127.0.1.100
SPGW Allocated IP 172.16.0.2 to IMSI 001010123456789
Adding attach accept to Initial Context Setup Request
Initial Context Setup Request -- eNB UE S1AP Id 1, MME UE S1AP Id 1
Initial Context Setup Request -- E-RAB id 5
Initial Context Setup Request -- S1-U TEID 0x1. IP 127.0.1.100
Initial Context Setup Request -- S1-U TEID 0x1. IP 127.0.1.100
Initial Context Setup Request -- QCI 7
Received Initial Context Setup Response
E-RAB Context Setup. E-RAB id 5
E-RAB Context -- eNB TEID 0x460003; eNB GTP-U Address 127.0.1.1
UL NAS: Received Attach Complete
Unpacked Attached Complete Message. IMSI 1010123456789
Unpacked Activate Default EPS Bearer message. EPS Bearer id 5
Received GTP-C PDU. Message type: GTPC_MSG_TYPE_MODIFY_BEARER_REQUEST
Sending EMM Information

```

## 4. 檢查環境

### 4.1. EPC 網卡

EPC 開啟後會產生新的網卡介面：srs\_spgw\_sgi

使用 ifconfig 指令查看

```
asus-medium@asusmedium-UN65H: ~
asus-medium@asusmedium-UN65H:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 78:24:af:04:55:03
          inet addr:192.168.128.101 Bcast:192.168.128.255 Mask:255.255.255.0
          UP BROADCAST RUNNING PROMISC MULTICAST  MTU:1500  Metric:1
          RX packets:31202 errors:0 dropped:0 overruns:0 frame:0
          TX packets:15571 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:22414352 (22.4 MB)  TX bytes:4384829 (4.3 MB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:2920 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2920 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:377570 (377.5 KB)  TX bytes:377570 (377.5 KB)

srs_spgw_sgi Link encap:UNSPEC  HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
          inet addr:172.16.0.1 P-t-P:172.16.0.1 Mask:255.255.255.0
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Metric:1
          RX packets:20 errors:0 dropped:0 overruns:0 frame:0
          TX packets:24 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:500
          RX bytes:1680 (1.6 KB)  TX bytes:1936 (1.9 KB)

asus-medium@asusmedium-UN65H:~$
```

epc 根據 epc.conf 設定裡的 sgi\_if\_name

產生新的網卡介面：srs\_spgw\_sgi

```
epc.conf (~/.config/srslte) - gedit
Open [F] Save

#####
# SP-GW configuration
#
# gtpu_bind_addr: GTP-U bind address.
# sgi_if_addr: SGI TUN interface IP address.
# sgi_if_name: SGI TUN interface name.
# max_paging_queue: Maximum packets in paging queue (per UE).
#
#####

[spgw]
gtpu_bind_addr = 127.0.1.100
sgi_if_addr = 172.16.0.1
sgi_if_name = srs_spgw_sgi
max_paging_queue = 100

#####
# PCAP configuration
#
# Packets are captured to file in the compact format decoded by
# the Wireshark s1ap dissector and with DLT 150.
# To use the dissector, edit the preferences for DLT_USER to
# add an entry with DLT=150, Payload Protocol=s1ap.
#
# enable: Enable or disable the PCAP.
# filename: File name where to save the PCAP.
#
#####

[pcap]
enable = false
filename = /tmp/epc.pcap

Matlab Tab Width: 8 Ln 1, Col 1 INS
```

## 4.2. UE 網卡

成功建立連線後，UE 會產生新的網卡介面：tun\_srsue

利用 ifconfig 指令查看

```
ue@ue-X580VD: ~  
ue@ue-X580VD:~$ ifconfig  
lo                Link encap:Local Loopback  
                  inet addr:127.0.0.1  Mask:255.0.0.0  
                  UP LOOPBACK RUNNING  MTU:65536  Metric:1  
                  RX packets:605 errors:0 dropped:0 overruns:0 frame:0  
                  TX packets:605 errors:0 dropped:0 overruns:0 carrier:0  
                  collisions:0 txqueuelen:1000  
                  RX bytes:49266 (49.2 KB)  TX bytes:49266 (49.2 KB)  
  
tun_srsue         Link encap:UNSPEC  HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00  
                  inet addr:172.16.0.2  P-t-P:172.16.0.2  Mask:255.255.255.0  
                  UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Metric:1  
                  RX packets:10 errors:0 dropped:0 overruns:0 frame:0  
                  TX packets:10 errors:0 dropped:0 overruns:0 carrier:0  
                  collisions:0 txqueuelen:500  
                  RX bytes:840 (840.0 B)  TX bytes:840 (840.0 B)
```

ue 根據 ue.conf 設定裡的 ip\_devname  
產生新的網卡介面 : tun\_srsue

```
ue.conf (~/.config/srslte) - gedit  
#  
# pdsch_8bit decoder:    Use 8-bit for LLR representation and turbo decoder trellis computation  
# (Experimental)  
#####  
[expert]  
#ip_netmask            = 255.255.255.0  
#ip_devname             = tun_srsue  
#mbms_service          = -1  
#rsssi_sensor_enabled  = false  
#rx_gain_offset        = 62  
#prach_gain            = 30  
#cqi_max               = 15  
#cqi_fixed             = 10  
#snr_ema_coeff         = 0.1  
#snr_estim_alg         = refs  
#pdsch_max_its         = 8    # These are half iterations  
#nof_phy_threads       = 3  
#equalizer_mode        = mmse  
#sfo_ema               = 0.1  
#sfo_correct_period    = 10  
#sss_algorithm         = full  
#estimator_fil_auto    = false  
#estimator_fil_stddev  = 1.0  
#estimator_fil_order   = 4  
#snr_to_cqi_offset     = 0.0  
#interpolate_subframe_enabled = false  
#sic_pss_enabled       = true  
#pregenerate_signals   = false  
#metrics_csv_enable    = false  
#metrics_period_secs   = 1  
#metrics_csv_filename  = /tmp/ue-metrics.csv  
Matlab Tab Width: 8 Ln 223, Col 70 INS
```

### 4.3. EPC 程式-netstat

利用 netstat 指令，確認 srsLTE 程式是否有開啟。

在終端機輸入

```
sudo netstat -alpn | grep srs
```

```

asus-medium@asusmedium-UN65H: ~
asus-medium@asusmedium-UN65H:~$ sudo netstat -alepn | grep srs
udp        0      0 127.0.1.1:2152      0.0.0.0:*           0          1698722      5337/srsenb
udp        0      0 127.0.1.100:2152    0.0.0.0:*           0          1700872      5267/srsepc
unix 2      [ ]          DGRAM               1700867  5267/srsepc      @mme_s11@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
unix 2      [ ]          DGRAM               1700873  5267/srsepc      @spgw_s11@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
asus-medium@asusmedium-UN65H:~$

```

#### 4.4. 偵測/重啟 USRP-B210

利用以下指令，讓電腦偵測是否可以讀取 usrp

uhd_find_devices	較簡單的訊息
uhd_usrp_probe	較詳細的訊息

使用下列指令，清除電腦之前讀取的相關檔案

/usr/lib/uhd/uhdutils/b2xx_fx3_utils -D
---

```

asus-medium@asusmedium-UN65H: ~
asus-medium@asusmedium-UN65H:~$ /usr/lib/uhd/uhdutils/b2xx_fx3_utils -D
Device opened (VID=0x2500,PID=0x0020)
B2xx detected... Control of B2xx granted...

Operation complete! I did it! I did it!
asus-medium@asusmedium-UN65H:~$

```

## 五、測試

### 1. 互通測試

根據 EPC 設定，EPC 預設 ip 為 172.16.0.1

UE 在終端機輸入

```
ping 172.16.0.1 -c 10
```

若是有收到回覆，則代表平台建置成功。

```
ue@ue-X580VD: ~  
ue@ue-X580VD:~$ ping 172.16.0.1 -c 10  
PING 172.16.0.1 (172.16.0.1) 56(84) bytes of data.  
64 bytes from 172.16.0.1: icmp_seq=1 ttl=64 time=179 ms  
64 bytes from 172.16.0.1: icmp_seq=2 ttl=64 time=16.0 ms  
64 bytes from 172.16.0.1: icmp_seq=3 ttl=64 time=14.0 ms  
64 bytes from 172.16.0.1: icmp_seq=4 ttl=64 time=11.8 ms  
64 bytes from 172.16.0.1: icmp_seq=5 ttl=64 time=18.0 ms  
64 bytes from 172.16.0.1: icmp_seq=6 ttl=64 time=15.8 ms  
64 bytes from 172.16.0.1: icmp_seq=7 ttl=64 time=12.8 ms  
64 bytes from 172.16.0.1: icmp_seq=8 ttl=64 time=9.98 ms  
64 bytes from 172.16.0.1: icmp_seq=9 ttl=64 time=16.9 ms  
64 bytes from 172.16.0.1: icmp_seq=10 ttl=64 time=13.8 ms  
  
--- 172.16.0.1 ping statistics ---  
10 packets transmitted, 10 received, 0% packet loss, time 9012ms  
rtt min/avg/max/mdev = 9.983/30.951/179.973/49.728 ms  
ue@ue-X580VD:~$
```

### 2. Wireshark 介面查看

另外也可以利用 Wireshark 軟體查看封包，可以直接選擇 srs 產生的網卡，也可以選擇 any 全部監看。

利用 Filter 過濾出想查看的封包：

ip.addr == 172.16.0.2 #符合 ip 位址為 172.16.0.2 的封包

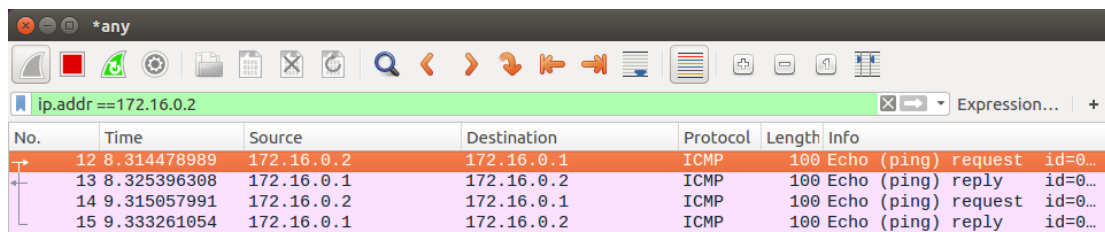
ip.dst == 172.16.0.1 #符合 ip 目的位址為 172.16.0.1 的封包

ip.src == 172.16.0.2 #符合 ip 來源位址為 172.16.0.2 的封包

tcp #符合協定為 tcp 的封包

udp #符合協定為 udp 的封包

eth.addr == 12:34:56:78:90:aa #符合 MAC 位址的封包



No.	Time	Source	Destination	Protocol	Length	Info
12	8.314478989	172.16.0.2	172.16.0.1	ICMP	100	Echo (ping) request id=0...
13	8.325396308	172.16.0.1	172.16.0.2	ICMP	100	Echo (ping) reply id=0...
14	9.315057991	172.16.0.2	172.16.0.1	ICMP	100	Echo (ping) request id=0...
15	9.333261054	172.16.0.1	172.16.0.2	ICMP	100	Echo (ping) reply id=0...