

Multiband Atheros Driver for Wireless Fidelity

實驗名稱：Multiband Atheros Driver for Wireless Fidelity Project (MadWifi) [1]

實驗目標：

在 Linux 的環境下實作出 MadWifi project。讓 Atheros wireless NIC 能夠模擬底下數種模式，藉此了解各種不同架構下 wireless network 的運作。

- sta : typical WLAN client station
- ap : Access point
- adhoc : IBSS mode
- ahdemo : Ad-hoc Demo
- monitor : This device can be used to "sniff" raw 802.11 frames
- wds : Wireless Distribution System

實驗環境 [2]：

■ **硬體 (Infrastructure Mode)**

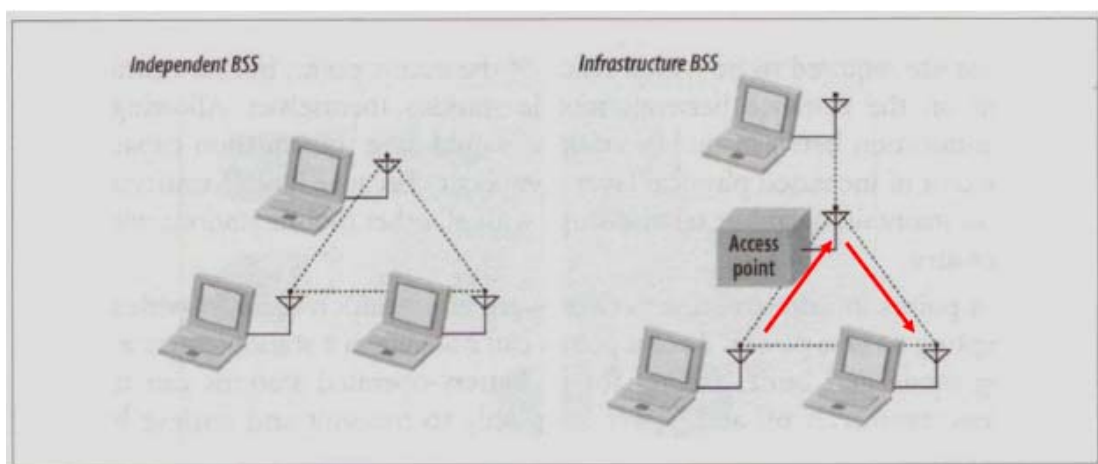
- Access Point * 1
 - ▶ 筆記型電腦 (Intel® Centrino® Duo 行動運算技術)
 - v CPU (Intel Pentium M or better)
 - v RAM (512MB or better)
 - v HD (80G or better)
 - v Display Card
 - v Sound Card
 - v Wire NIC
 - v Wireless NIC (Atheros chipset) [3]
- Station 1 * 1
 - ▶ 筆記型電腦 (Intel® Centrino® Duo 行動運算技術)
 - v CPU (Intel Pentium M or better)
 - v RAM (512MB or better)
 - v HD (80G or better)
 - v Display Card
 - v Sound Card
 - v Wire NIC
 - v Wireless NIC (Atheros chipset) [3]
- Station 2 * 1
 - ▶ 筆記型電腦 (Intel® Centrino® Duo 行動運算技術)
 - v CPU (Intel Pentium M or better)
 - v RAM (512MB or better)

- v HD (80G or better)
- v Display Card
- v Sound Card
- v Wire NIC
- v Wireless NIC (Atheros chipset) [3]

■ 軟體

- 作業系統：
 - ▶ Linux 2.4.23+ or Linux 2.6.x series
 - v Debian [4]
 - v Gentoo [5]
 - v Mandrake [6]
 - v RedHat (including Fedora Core) [7, 8]
 - v SlackWare [9]
 - v SuSE [10]
 - v Ubuntu [11]
 - MadWifi Driver [12]

實驗架構圖 [13]：



實驗步驟 [5]：

1. 在相關硬體上安裝 Linux 與 Atheros Driver，讓 wireless NIC 可以正常運作。
2. 在 sta mode 下進行實驗，藉此了解 typical WLAN client station 的運作過程。
3. 在 ap mode 下進行實驗，藉此了解 Infrastructure Mode 的運作過程。
4. 在 adhoc mode 下進行實驗，藉此了解 IBSS Mode 的運作過程。
5. 在 ahdemo mode 下進行實驗，藉此了解 IBSS Mode demo 的運作過程。
6. 在 monitor mode 下進行實驗，藉此了解 sniff 802.11 frames 的運作過程。
7. 在 wds mode 下進行實驗，藉此了解 Wireless Distribution System 運作過程 (optional)。

預期成果：

透過本實驗的實作，除了讓學生習得實務操作上的經驗，更讓學生充分了解

到 wireless network 下各種不同 mode 的運作流程，達到實務與理論相互驗證的目的。

Documentations :

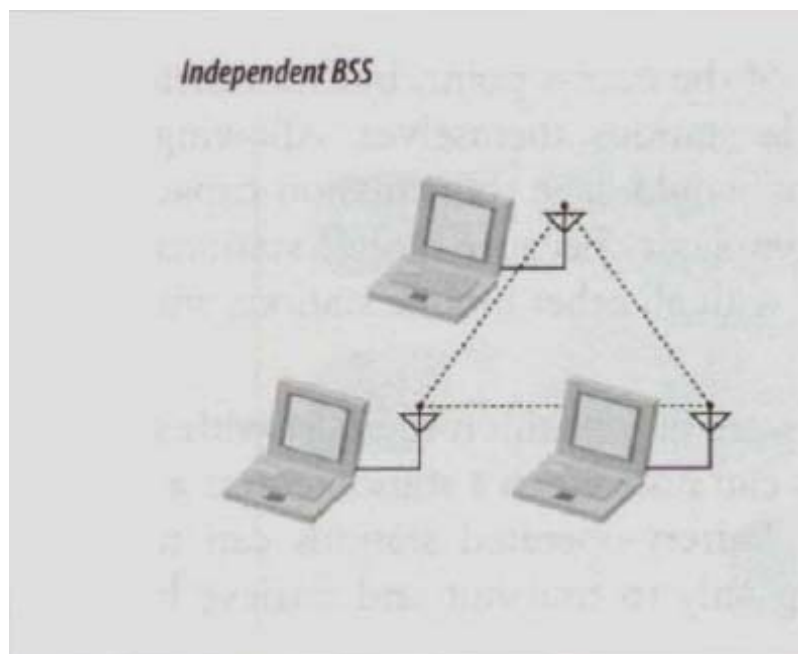
1. MadWifi User Documentation [14]
2. The Linux Documentations Project [15]

References :

- [1] <http://madwifi.org/>
- [2] <http://madwifi.org/wiki/Requirements>
- [3] <http://madwifi.org/wiki/Compatibility>
- [4] <http://www.debian.org/>
- [5] <http://www.gentoo.org/>
- [6] <http://mdk.linux.org.tw/>
- [7] <http://www.redhat.com/>
- [8] <http://www.redhat.com/fedora/>
- [9] <http://www.slackware.com/>
- [10] <http://www.novell.com/linux/>
- [11] <http://www.ubuntu.com/>
- [12] <http://madwifi.org/wiki/UserDocs/GettingMadwifi>
- [13] <http://www.csie.ncue.edu.tw/~ycchan/wl2006/WL-802.11-overview.pdf>
- [14] <http://madwifi.org/wiki/UserDocs>
- [15] <http://www.tldp.org/>
- [16] <http://pof.eslack.org/blog/2006/06/30/traient-li-el-suc-a-madwifi-ng/>
- [17] http://freebsd.ntut.idv.tw/document/freebsd_wireless_ap.html

實驗一. AD-HOC

架構圖



1. 第一台電腦設定如下

```
vi ad-hoc
```

按s表示做編輯動作

內容如下

以下兩行指令為啟動ad-hoc功能

```
wlanconfig ath0 destroy
```

```
wlanconfig ath0 create wlandev wifi0 wlanmode adhoc bssid
```

```
iwconfig ath0 essid "orz" //設定essid
```

```
iwconfig ath0 channel 3 //設定channel
```

```
ifconfig ath0 192.168.2.1 netmask 255.255.255.0 up //設定ip和遮罩
```

```
:wq //存檔離開
```

```
chmod 755 ad-hoc // 把權限定義成最高,然後,可以執行
```

```
./ad-hoc //執行啟動ad-hoc功能
```

2. 第二台電腦設定如下

```
vi ad-hoc
```

按s表示做編輯動作

內容如下

以下兩行指令為啟動ad-hoc功能

```
wlanconfig ath0 destroy
wlanconfig ath0 create wlandev wifi0 wlanmode adhoc bssid
iwconfig ath0 essid "orz" //設定essid
iwconfig ath0 channel 3 //設定channel
ifconfig ath0 192.168.2.2 netmask 255.255.255.0 up //設定ip和遮罩
:wq //存檔離開
chmod 755 ad-hoc // 把權限定義成最高,然後,可以執行
./ad-hoc //執行啟動ad-hoc功能
```

ps. 兩台電腦 essid 與 channel 務必設為相同

3. Ad-hoc 模式測試

If 兩台電腦可以互 ping 就表示實驗完成

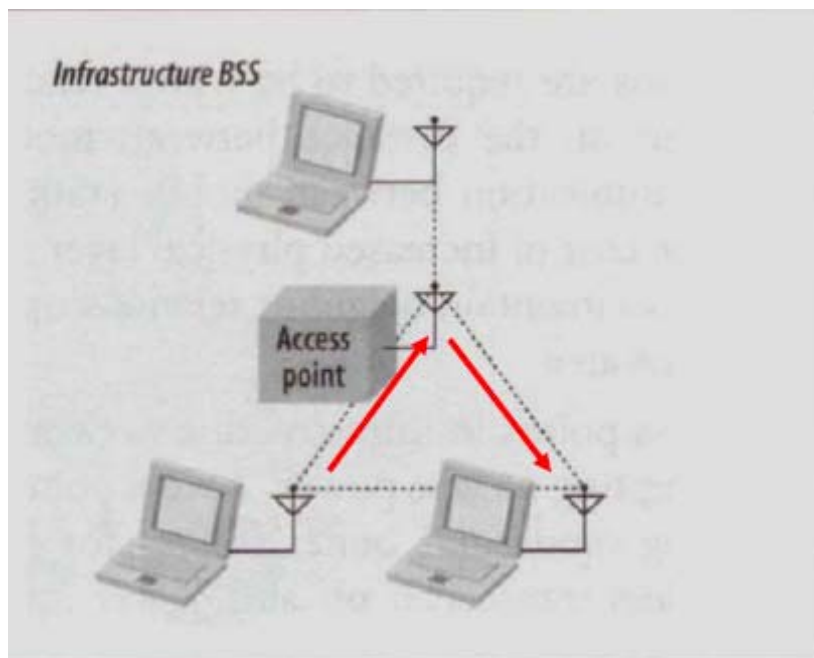
例如第一台電腦想要 ping 第二台電腦則在第一台電腦裡面終端機模式下輸入指令 ping 192.168.2.2

4. 安全移除網卡

```
ifconfig ath0 down
cardctl eject
cardctl status
```

實驗二.AP~使用 NAT+DHCP 方法

架構圖



安裝 linux 請選擇『完整安裝』，在安裝過程中遇到設定防火牆選項，請選擇關閉防火牆

1. 安裝網卡驅動從 madwifi 下載

下指令 cd Desktop/

```
tar zxvf madwifi-0.9.2.tar.gz //解壓縮
cd madwifi-0.9.2
make //編譯
make install //安裝
modprobe ath_pci //載入網卡
ifconfig ath0 up //如果燈會閃動代表驅動成功
```

2. 設定 ap

在終端機模式下輸入

下指令 vi ap

按s表示做編輯動作

//以下兩行指令為Wireless Setup把ap功能打開

```
wlanconfig ath0 destroy
```

```
wlanconfig ath0 create wlandev wifi0 wlanmode ap bssid
```

```
iwconfig ath0 essid "orz" //設定essid
```

```
iwconfig ath0 channel 6 //設定channel
```

```
ifconfig ath0 192.168.2.254 netmask 255.255.255.0 up //內部 ip 設 192.168.2.254
```

```
ifconfig eth0 163.23.227.139 netmask 255.255.255.224 up //外部 ip 設 163.23.227.139
route add default gw 163.23.227.158 //gateway ip
modprobe iptable_nat //載入 nat 模組
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE //設定 ip 偽裝規則
echo 1 > /proc/sys/net/ipv4/ip_forward //啟動封包轉送
```

:wq //存檔離開

chmod 755 ap // 把權限定義成最高,然後,可以執行

3.//以下為編輯 DHCP

vi /etc/dhcpd.conf

```
ddns-update-style none;
#default-lease-time 259200;
#max-lease-time 518400;
option routers 192.168.2.254; //ap的網路位置
option broadcast-address 192.168.2.255;
option domain-name-servers 168.95.1.1;

subnet 192.168.2.0 netmask 255.255.255.0
{
    range 192.168.2.1 192.168.2.253; // ip range 192.168.2.1~192.168.2.253
}
```

:wq //存檔離開

4.//以下兩行為設定 DNS

vi /etc/resolv.conf

```
nameserver 168.95.1.1
```

:wq //存檔離開

./ap //啟動 ap 功能

dhcpd -cf /etc/dhcpd.conf //啟動 DHCP 功能

5.ap 實驗測試~

只要有無線網卡的電腦打開無線網路搜尋 ap，可以容易自動取得 ip 並透過有 ap 功能那台電腦連上網際網路。達成上述方法代表實驗成功！

6.安全移除網卡

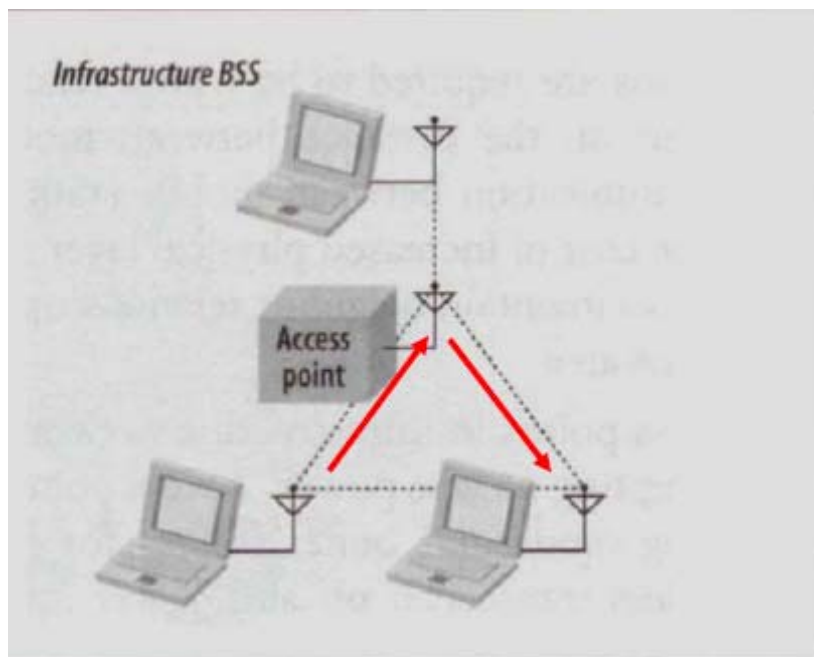
```
ifconfig ath0 down
```

```
cardctl eject
```

```
cardctl status
```

實驗二.AP~使用橋接的方法

架構圖



安裝 linux 請選擇『完整安裝』，在安裝過程中遇到設定防火牆選項，請選擇關閉防火牆

2. 安裝網卡驅動從 madwifi 下載

下指令 cd Desktop/

```
tar zxvf madwifi-0.9.2.tar.gz //解壓縮
cd madwifi-0.92
make //編譯
make install //安裝
modprobe ath_pci //載入網卡
ifconfig ath0 up //如果燈會閃動代表驅動成功
```

3. vi /etc/resolv.conf

輸入nameserver DNS

例如 203.68.220.1

4. 從光碟安裝 bridge 套件

查詢 bridge 套件指令

```
cd /media/cdrom/
cd Fedora/
cd RPMS/
```

ls br 按 Tab 鍵 2 次

安裝套件指令 rpm -ivh bridge-utils-1.0.4-4.i386.rpm //橋接程式
 rpm -ivh bridge-utils-devel-1.0.4-4.i386.rpm //橋接工具

5. 建立 ap 步驟

下指令 vi ap

按s表示做編輯動作

```
-----  
ifconfig ath0 down    //開始把ath0給關閉是為了之後設定,  
                      rmmod -w ath_pci //將無線網卡給移除~那只是為了確保它可以正常的喚醒  
                      我們的設定  
                      modprobe ath_pci    //這是將網卡給喚醒起來,因為無線網卡是可以做移除  
                      的動作  
ifconfig -a    //只是show 出目前所有的網路設定  
//以下三行指令為Wireless Setup把ap功能打開  
wlanconfig ath0 destroy  
wlanconfig ath0 create wlandev wifi0 wlanmode ap bssid  
wlanconfig ath0 create wlandev wifi0 wlanmode sta nosbeacon  
iwconfig ath0 essid "hahaha" channel 6 //設essid及channel  
brctl addbr br0            //新增的橋接器bridge,將之命名為br0,  
                            之後將bridge的一端連接到有線網卡  
                            卡,一端到無線網卡  
  
brctl addif br0 eth0  
brctl addif br0 ath0  
ifconfig ath0 down  
ifconfig br0 down  
ifconfig ath0 0.0.0.0 up    //ip分別歸0,是為了待會使用bridge橋接時將有線與  
無線橋接起來  
ifconfig eth0 0.0.0.0 up  
ifconfig br0 192.168.1.22 up            //有線網卡ip  
echo "1" > /proc/sys/net/ipv4/ip_forward //是否要核心轉送封包。預設是關閉  
                                          的  
route add default gw 192.168.1.1        //gateway ip  
:wq存檔離開  
chmod 755 ap    // 把權限定義成最高,然後,可以執行  
./ap            //執行啟動ap功能
```

6. ap 實驗測試~

- ▶ 上述方法是在虛擬 ip 環境下，用橋接的方式架 ap，所以只要有無線網卡的電腦打開無線網路搜尋 ap，可以容易自動取得 ip 並透過有 ap 功能那台電腦連上網際網路。達成上述方法代表實驗成功！
- ▶ 針對實體 ip 環境方面，必須要有兩個實體 ip，第一個 ip 給有 ap 功能那台電腦使用，第二個 ip 給想要連上網際網路電腦使用。請注意兩台電腦的閘道，遮罩，DNS 設定必須相同，只要能透過有 AP 功能那台電腦連上網路實驗就算成功。

7. 安全移除網卡

```
ifconfig ath0 down  
cardctl eject  
cardctl status
```

實驗二. ap~在 FreeBSD 環境下架設 AP

Setp 1

重新編譯 Kernel，FreeBSD6.0 在預設下裝完 OS 是抓不到無線網卡的，所以我們要自行 Compiler 新的 Kernel 才能驅動這個硬體。

```
#cd /usr/src/sys/i386/conf
#cp GENERIC /etc/WIFI
#ln -s /etc/WIFI
#vi WIFI # kernel 的社定檔要加入以下的設定
# wireless suport
device ath # Atheros IEEE 802.11 wireless network driver
device ath_hal # Atheros Hardware Access Layer
device ath_rate_sample # John Bicket's SampleRate control algorithm.
device wi
device wlan # 802.11 support (Required)
device wlan_wep # WEP crypto support for 802.11 devices
device wlan_ccmp # AES-CCMP crypto support for 802.11 devices
device wlan_tkip # TKIP and Michael crypto support for 802.11 devices
device wlan_xauth # External authenticator support for 802.11 devices
device wlan_acl # MAC-based ACL support for 802.11 devices
options IPFIREWALL
options IPDIVERT
options IPFIREWALL_DEFAULT_TO_ACCEPT
options IPFIREWALL_VERBOSE
options IPFIREWALL_VERBOSE_LIMIT=10
options IPFIREWALL_FORWARD

#config WIFI # 一切就緒開始編譯核心
#cd ../compile/WIFI
#make cleandepend; make depend all install
#vi /boot/loader.conf # 讓開機就自動載入無線網路的 funtion

wlan_wep_load="YES"
wlan_tkip_load="YES"
wlan_ccmp_load="YES"
wlan_xauth_load="YES"
wlan_acl_load="YES"

#sync;sync;sync;reboot # sync 是把 ram 的資料寫到 Disk 上
```

核心編譯完成後重新啟動 FreeBSD。

Setp 2

#vi hostap.sh

```
#!/bin/sh
Myip=192.168.100.1
Myopt=Hostap
Mychannel=11
ifconfig ath0 $myip netmask 255.255.255.0
ifconfig ath0 ssid AP channel $Mychannel
ifconfig ath0 mode 11g mediaopt $Myopt
echo "Wireless work in $Myopt Mode"
echo "IP Address : $myip"
echo "Work channel : $Mychannel"
```

#chmod 755 hostap.sh

#!/hostap.sh

測試方法

用一台 notebook (OS:WindowXP) 檢視無線網路，看看是否可以找到我們所架設的 AP，登入進去看看是否可以取得 IP 位址（表示 DHCP 功能正常），在打開瀏覽器測試看看可以連出 Internet 嗎？（表示 NAT 功能正常）。