

無線網路

愈來愈熱門的無線網路（wireless networking）技術，有著許多傳統有線技術無法俾倪的優點，但也帶來了原本在有線網路上不必太顧忌的隱憂。最顯著的優點是使用便利與網路安裝成本的降低，因為你的電腦 — 特別是筆記型電腦 — 不再受網路線的牽絆，同時也省去的纜線、插座、接頭、集線器、佈線施工等等成本。在無線網路裡增加新主機，幾乎不費吹灰之力，只要將無線網卡插入電腦，然後開機，差不多就搞定了。無線技術將網路涵蓋範圍擴展到傳統技術難以企及之處 — 機場、車站、旅館、或任何你想像得到的地方。

無線網路技術能受到消費者的歡迎，有部份原因要歸功於筆記型電腦的暢銷，或者，應該反過來說，無線網路技術提昇了筆記型電腦的便利性，而促成暢銷。現在，幾乎每一款新機型，都整合了 802.11b 或 802.11g，好像沒有內建無線網路硬體，就沒資格成為筆記型電腦似的。無線風潮甚至吹向平常不太需要搬動的桌上型機種，在翻譯這篇文章的期間，無線網卡的售價已經快可以跟傳統的 Ethernet 網卡相比了。

然而，興一利生一弊，無線技術帶來了便利性，也帶來了安全隱憂。

源起

無線區域網路（Wireless LAN）的基礎是「展頻技術」（spread spectrum）【譯註】，這種技術是美軍於二戰期間研發出來的軍用通訊科技。軍事科學家會看中展頻，是因為這種技術較能夠抗拒干擾，另一項優點是可以提昇無線電資料率。1945 年之後，有商務企業發現展頻技術的潛力，因而開始擴展相關技術與應用。

●○.....

譯註 <http://www.sss-mag.com/ss.html#tutorial> 有一篇關於「展頻」的入門課。

展頻技術應用於無線區域網路，起始於夏威夷大學於 1971 發展的 AlohNet 計畫，其目的是讓各島上的七部電腦可透過位於 Oahu 的中央交換站進行雙向通訊。

夏威夷大學 AlohNet 計畫的研究成果，成為第一代的無線網路技術，當時所用的頻率範圍是 901-928 MHz，主要應用於軍事用途，鮮少出現於消費領域，因為這段頻帶相當擁擠，而且速度相對較低。

後來，FCC（美國聯邦通訊委員會）規定開放 2.4 GHz 頻率，讓任何人都可以不經申請而直接使用，所以無線技術也開始轉移到此段頻帶，並樹立了 802.11 標準規格，此規格後來演變成廣為接受的 802.11b 標準。

802.11 標準

應用於 PC 無線網路的規格標準，主要是由 IEEE（Institute of Electrical and Electronics Engineers）制定的。跟 LAN/MAN 有關的技術，長久以來都是使用 802 作為標準編號；有些無線工作團隊相當活躍，包括 802.15（Bluetooth 個人區網）、802.16（寬頻無線系統）、以及 802.11（無線區域網路）。其中 802.11 還定義了多種子規格：

802.11a

使用 5 GHz 頻帶，傳輸速度最高可達 54 MBps。但因為硬體價格相對偏高，而且距離較短，所以沒造成流行。

802.11b

使用 2.4 GHz 頻帶，傳輸速度最高可達 11 MBps，傳輸範圍可達 500 公尺。在 802.11g 之前，這是最多人使用的無線網路標準。

802.11g

使用 2.4 GHz 頻帶，傳輸速度最高可達 54 MBps，引進 WPA（WiFi Protected Access）來提昇安全性。802.11g 正在逐漸取代 802.11b，是未來最有希望成為主流的標準規格。

802.11i

這是仍在研議中的未完成標準，其重點在於解決 802.11b 引發的諸多安全顧慮，並提供更為強健的驗證與加密系統。

802.11n

802.11n 預期將提供 100 Mbps 傳輸速度 — 將近目前最快的 802.11g 的兩倍快 — 但同時維持 b 與 g 的回溯相容性。在翻譯這本書的時候，802.11n 還不是一個完整的標準，但是已經有好幾家廠商依據早期的規格書開始偷跑了，提供所謂的「pre-n」產品，從這個現象可看出 802.11n 的未來潛力。

802.11 的安全顧慮

無線網路的傳輸媒介是電波，一種沒有明確邊界與屏障的傳媒。只要在電波涵蓋範圍內，就可以接收到網路信號。當初 IEEE 制定 802.11 標準，就已經瞭解到無線網路的開放本質，一定需要某種保護機制來保障網路資料的完整性，於是他們另外制定了 WEP (Wired Equivalent Privacy) 標準。此標準以 40 / 128-bits 加密技術來保證使用者可獲得等同於傳統有線網路的資料完整性。

然而，對這種安全性的期望，很快地就破滅了。Scott Fluhrer、Itsik Mantin 和 Adi Shamir 三人聯名發表的一篇論文《Weaknesses in the Key Scheduling Algorithm of RC4》，詳述了 WEP 所採用的金鑰產生演算法的弱點，在這篇論文公佈後沒多久，Rice 大學的一位學生，Adam Stubblefield，就將理論化為實際，以行動証實 WEP 確實是可被破解的。雖然他從未公開他所用的工具，但是現在已經有許多類似的 Linux 工具可用來破解 WEP，使得 WEP 成為一個不值得信任的機制。

儘管如此，我們也不必過度菲薄 WEP 的安全性。因為破解 WEP key 仍需要相當長的時間，而且攻擊者必須要事先蒐集相當大量的密文樣本才行。使用 AirSnort 破解工具，大約需要事先蒐集將近一千萬個密文封包，才有足夠的資料樣本可供分析。就算是在一個持續保持滿檔的超繁忙無線區網上，至少也要 10 小時以上才能夠蒐集到足夠的資料樣本。然而，大多數網路都不太可能持續維持滿載如此之久，所以，要蒐集到足夠資料樣本所需的時間，甚至可能長達好幾天甚至數週。

總結來說，對於小型的無線區網，WEP 的保護其實也夠了，若真的有顧忌，每隔一兩週換一次密鑰，應該足以使攻擊者放棄念頭。相對地，當真要阻擋竊聽與惡意破壞，則應該使用 VPN 技術，而且絕對不應該讓無線網路與內部有線網路相通。

802.11 網路拓樸

802.11 網路有多種拓樸模式，你可能遇到的主要兩種是基地台模式 (**infrastructure mode**) 與對等模式 (**ad-hoc mode**)。基地台模式 (參考圖 17-1) 需要一個中央存取點 (**central access point**) — 也就是所謂的「無線基地台」— 來仲介各節點之間的通訊，所以又稱為**中控模式 (managed mode)**。在對等模式 (參考圖 17-2) 下，各節點之間可以彼此直接通聯，形成魚網狀的點對點網路。一般而言，基地台模式的效率比較好，其傳輸速率接近傳統的有線網路，很適合部署於辦公室環境，讓無線用戶端也可以存取有線網路上的伺服器。另一方面，如果只是想讓 PC 之間可以傳輸檔案 (或 PC 與無線週邊裝置，像是印表機、PDA、手機、數位相機 ...)，而不存取無線網路之外的資源，對等模式是比較好的選擇。Bluetooth (藍芽) 技術採用的就是對等式拓樸。



圖 17-1：基地台模式

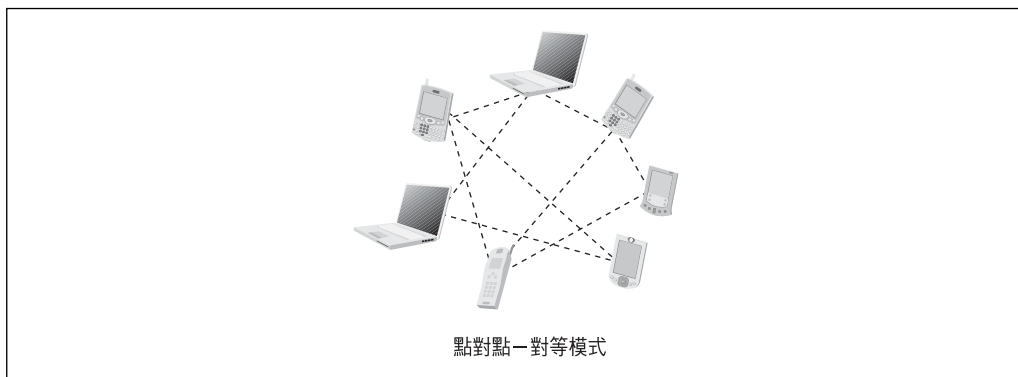


圖 17-2：對等模式

802.11b 網路使用一組預定的頻率，稱為**頻道 (channel)**。規格書定義了 14 個分離的頻道，但北美地區只能使用前 11 個頻道（從 2400 Mhz ~ 2483 Mhz，每個頻道各 22 Mhz 寬），台灣與歐洲地區可使用前 13 個頻道（2412 ~ 2472 Mhz），日本可使用全部 14 個頻道。很顯然，各頻道之間有部份頻帶重疊，所以，在選定頻道之前，有必要先掃描鄰近地區的無線網路（這種動作稱為「站台調查」，site survey），選擇一個與鄰近無線網路相隔較遠的頻道，以免受到不必要的干擾。

由於基地台模式是網路管理者最可能遇到的拓樸形式，所以本章重點放在這種模式。基地台（存取點）可以是一台從商店購置的整合性裝置，也可以是一台跑 HostAP 的 Linux 機器。

基地台會發送一種稱為信標 (beacon) 的信號，讓鄰近主機知道基地台的存在。信標的作用如同燈塔的旋轉燈光，每個信標本身都含有一小段訊息，其中含有讓用戶端知道如何連線的必要資訊 (先前提到的「站台調查」，就是蒐集鄰近基地台發出的信標)。收到信標的用戶端，可對基地台送出連結要求，若基地台接受，則回送一個表示同意的連結訊框給用戶端。基地台接受連結要求之前，可以對用戶端進行某種驗證，比方說，檢查用戶端的 MAC 位址，或是要求用戶端提供某種識別證明。

硬體

無線網路需要兩種硬體設備，一種是「基地台」(Access Point, AP)，另一種是「無線網卡」。不同廠家出品的基地台與無線網卡可以互通，前提是彼此要共同支援同一種 802.11 標準。有些公司標榜他們的產品可以提供比 802.11 標準更快的傳輸速度，不過，只有在基地台與無線網卡都必須是該公司的產品時，而且要搭配專屬的驅動程式，才會有這種效果。

市售的基地台通常含有幾個 Ethernet 連接埠，讓你的無線網路能夠與有線網路整合在一起，有些機種甚至內建 DHCP、NAT 等功能，讓你能將基地台當成閘道器使用。這類多功能型的基地台，都會提供一個 Web 操作介面，讓你能透過瀏覽器畫面來設定作業參數。

Linux 系統本身也可以當成基地台使用，但是需要使用特定廠牌的無線網卡，並搭配 HostAP 軟體 (<http://hostap.epitest.fi/hostapd/>)。

無線網卡的晶片組

無線網卡最主要的功能是收發無線電信號、在數位資料與電波信號之間的編解碼，承擔這些功能的主要元件是「晶片組」(chipset)。一般而言，晶片組製造商會提供「設計樣板」(reference design) 給無線網卡製造商；所謂的「設計樣板」通常包含電路設計圖 (俗稱「公板」) 與驅動程式，網卡製造商可略做修改 (或甚至完全不改)，然後印上自己的商標，開始銷售。很少有網卡製造商會大幅修改設計樣板，所以，從軟體的觀點來看，使用同一家晶片組的無線網卡，即使製造商不同，也可以用同樣的方式來驅動之。如同 Ethernet 網卡的支援方式，Linux 並未針對每一種無線網卡都提供專屬的驅動程式，而是針對無線晶片組來提供驅動程式。因此，如果你要為自己的 Linux 電腦選購無線網卡，最好事先打聽一下網卡所用的晶片組，然後到晶片組製造商的網站上 (或是 Linux 核心原始程式的說明文件中)，查看有沒有適當的 Linux 驅動程式可用。

在本書翻譯期間，市場上大概有四家主要的晶片組：

Atheros

大多數支援 802.11a 的無線網卡，幾乎都是使用 Atheros 晶片組。有些 802.11g 裝置也使用這家公司的晶片組。

Broadcom

內建 802.11g 介面的筆記型電腦，若不是使用 Intel Centrino 晶片組，大概就是使用 Broadcom 的 802.11g 晶片組。Apple 的 AirPort Extreme 就是使用 Broadcom 的晶片組。

Conexant (Prism)

在 802.11 a/b 發展初期，Prism 晶片組曾經紅極一時，因為這種晶片組只要搭配適當的驅動軟體，就可以當成基地台使用。Prism 公司後來經過數度轉手，現在落入 Conexant 的手中。Linux 對 Prism 晶片組的支援頗為齊全。

Intel (Centrino)

許多內建無線網卡的筆記型電腦，貼有 Intel 的「Centrino」商標。其實 Centrino 是一個行銷名詞，它代表包括 CPU 在內的整組 Intel 系統晶片。Intel/PRO 2200 是 Centrino 的 802.11g 介面。由於 Centrino 涉及太多 Intel 專利技術，所以 Intel 以另一種授權方式來提供他們的驅動程式。詳情請見 IPW 2200 專案的網站 (<http://ipw2200.sourceforge.net/>)。

驅動程式

平心而論，Linux 核心目前提供的無線裝置驅動程式還不是很齊全，幸好，大多數晶片組製造商都很願意提供 Linux 驅動程式 — 雖然並非全部都是採用 Open Source 授權方式。儘管取得無線網卡的 Linux 驅動程式並不難，然而，多變的匯流排形式，卻使得驅動程式的安裝與設定工作變得頗為複雜。

依據匯流排形式而分，無線網卡可分成 PCI、CardBus (PCMCIA)、USB 三種。筆記型電腦內建的無線網路介面是 PCI (或 mini-PCI)，因為從作業系統的觀點來看，它們就像直接安插在 PCI slot 上。其實，市面上可找到的「PCI 無線網卡」(可安插在桌上型電腦的 PCI 插槽那種)，反而不多。

Linux 對於 PCI 與 CardBus 匯流排的支援頗為完備，但是對於 USB-Net 的支援尚不算成熟。大多數驅動程式都支援 PCI 或/與 CardBus，但是只有少數驅動程式支援 USB。

如同一般的 Ethernet 網卡，無線網卡的驅動程式可以被編譯進核心裡，也可以編譯成模組形式。由於大多數無線網卡驅動程式都仍在積極研發當中，其改版頻率相當高，所以建議你採用模組形式，如此便可以不必每次都要重新編譯整個核心。此外，CardBus 與 USB 網卡的驅動程式，也應該編譯成模組形式，讓 `cardmgr` 或 `USB-hotplug` 可以自動載入相關模組。

一個支援無線網路的 Linux 核心，必須具備 LWE (Linux Wireless Extensions) 子系統，如果你要使用模組，則核心也必須具備掛載、卸載模組的能力。大多數 Linux 版本隨附的核心，應該都具備前述兩項特性。若你的核心有釋出組態資訊（有 `/proc/config.gz` 檔），或是你有核心的 `.config` 檔（通常可在 `/boot` 或 `/usr/src/linux` 目錄下找到），則可以用下列方式來檢查：

```
rox ~ # zcat /proc/config.gz | grep -i CONFIG_NET_RADIO
CONFIG_NET_RADIO=y
rox ~ # zcat /proc/config.gz | grep -i CONFIG_MODULE
CONFIG_MODULES=y
CONFIG_MODULE_UNLOAD=y
# CONFIG_MODULE_FORCE_UNLOAD is not set
# CONFIG_MODULE_SRCVERSION_ALL is not set
```

若你的核心的 `CONFIG_NET_RADIO`、`CONFIG_MODULES`、`CONFIG_MODULE_UNLOAD` 不是設定為 `y`，則需要重新編譯一個支援這些選項的核心。

LWE 與相關工具

從使用者的觀點來看，802.11 網路介面與一般的 Ethernet 介面無異，只不過它們有許多 Ethernet 所沒有的作業參數。與其讓每個驅動程式各自實作自己的設定工具，不如將所有共同的功能性定義成一個 Wireless Extension API，讓驅動程式有一個統一的軟體介面來提供功能性，如此使用者便可以有一套一致的設定工具。Linux Wireless Extensions（用於設定無線網卡的作業參數）與 `xsupplicant`（用於設定 WEP/WPA key）就是這種構想下的產物。

如果你的系統還沒安裝 LWE，請到 http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Tools.html 下載最新版本。

以下幾個小節示範如何設定無線網卡的作業參數。我們假設你已經有一個支援無線網路的 Linux 核心，而且驅動程式與相關工具都安裝好了。很抱歉這裡無法實際示範如何安裝驅動模組，因為這方面的操作程序隨驅動程式與無線網卡的匯流排形式而異，但大致上與一般 Ethernet 網卡差不多。如果你遇到困難，請參考《Linux Wireless Howto》（http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Wireless.html）。

使用 iwconfig 設定無線介面

當無線網卡的驅動模組被載入核心，並順利偵測出硬體之後，它會產生一個介面。此介面的名稱隨驅動程式而異。有些驅動程式會沿用 Ethernet 的命名慣例 (eth0、eth1 ...)，有些會自有一套命名方式，你可能會見到 wlanX、athX 之類的怪名稱。驅動程式的說明文件會告訴你他們所使用的名稱；或者，你可以查看 `dmesg` 訊息，或是使用 `iwconfig` 列出所有已知介面。

`iwconfig` 是 LWE 的主要工具，它能讓你用類似 `ifconfig` 的語法來設定支援 Wireless Extension API 的任何介面。直接執行它而不加任何參數時，它會列出所有網路介面，特別是支援 802.11 的無線網路介面：

```
[root@rox]# iwconfig
lo          no wireless extensions.

eth0       no wireless extensions.

eth1       unassociated  ESSID:off/any
           Mode:Managed Channel=0 Access Point: 00:00:00:00:00:00
           Bit Rate=0 kb/s Tx-Power=20 dBm
           RTS thr:off Fragment thr:off
           Encryption key:off
           Power Management:off
           Link Quality:0 Signal level:0 Noise level:0
           Rx invalid nwid:0 Rx invalid crypt:0 Rx invalid frag:0
           Tx excessive retries:0 Invalid misc:4519 Missed beacon:0

sit0       no wireless extensions.
```

在此例中，可看到 eth1 與 ath0 是無線網卡，你甚至可看到各介面的作業模式、頻率、基地台的 MAC 位址、信號品質、電源管理狀態、密鑰值等等。

搜尋網路

某些驅動程式提供信號掃描功能，對於這類裝置，你可用 `iwlist` 來掃描鄰近基地台的信號：

```
[root@rox]~# iwlist eth1 scan
eth1      Scan completed :
           Cell 01 - Address: 00:11:09:BF:BB:9F
           ESSID:"LINHOME"
           Protocol:IEEE 802.11bg
           Mode:Master
```



```

Channel:8
Encryption key:on
Bit Rate:54 Mb/s
Extra: Rates (Mb/s): 1 2 5.5 9 11 6 12 18 24 36 48 54
Extra: RSSI: -53 dBm
Extra: Last beacon: 24ms ago
Cell 02 - Address: 00:90:96:87:7E:D8
ESSID:"wireless"
Protocol:IEEE 802.11bg
Mode:Master
Channel:11
Encryption key:off
Bit Rate:54 Mb/s
Extra: Rates (Mb/s): 1 2 5.5 6 9 11 12 18 24 36 48 54
Extra: RSSI: -53 dBm
Extra: Last beacon: 1145ms ago

```

在此例中，`iwlist` 找到兩個無線網路，你可看到各網路的識別名稱 (ESSID)、拓模模式、頻率、信號品質、加密與否。請記下你想參與的無線網路之識別名稱、頻道、加密方式；對於有加密的網路，你還得知道 WEP key 為何。以下的示範以此例中的「LINHOME」無線網路為例。

連接無線網路

連接無線網路的第一步，是使用 `iwconfig` 的 `essid` 選項讓網卡知道你想參與的無線網路之識別名稱 (ESSID)：

```
[root@rox]# iwconfig eth1 essid "LINHOME"
```

每個無線網路都有一個識別名稱，而且大小寫有別。若名稱中含有空格，則必須以一對雙引號括住。設定識別名稱好之後，可以再用一次 `iwconfig` 檢查設定結果：

```

[root@rox]# iwconfig eth1
eth1      unassociated  ESSID:"LINHOME"
          Mode:Managed Channel=0 Access Point: 00:00:00:00:00:00
          Bit Rate=0 kb/s Tx-Power=20 dBm
          RTS thr:off Fragment thr:off
          Encryption key:off
          Power Management:off
          Link Quality:0 Signal level:0 Noise level:0
          Rx invalid nwid:0 Rx invalid crypt:0 Rx invalid frag:0
          Tx excessive retries:0 Invalid misc:4519 Missed beacon:0

```

下一步是使用 `channel` 設定介面所用的頻道：

```
[root@rox]# iwconfig eth1 channel 8
[root@rox]# iwconfig eth1
eth1      unassociated  ESSID:"LINHOME"
          Mode:Managed  Frequency=2.447 GHz Access Point: 00:00:00:00:00:00
          Bit Rate=0 kb/s   Tx-Power=20 dBm
          RTS thr:off   Fragment thr:off
          Encryption key:off
          Power Management:off
          Link Quality:0  Signal level:0  Noise level:0
          Rx invalid nwid:0  Rx invalid crypt:0  Rx invalid frag:0
          Tx excessive retries:0  Invalid misc:4519  Missed beacon:0
```

如你所見，`iwconfig` 會自動將頻道編號換算成對應的頻率。順便一提，使用 `iwlist` 可列出無線介面所支援的所有頻道：

```
[root@rox]# iwlist eth1 channel
eth1      14 channels in total; available frequencies :
          Channel 01 : 2.412 GHz
          Channel 02 : 2.417 GHz
          Channel 03 : 2.422 GHz
          Channel 04 : 2.427 GHz
          Channel 05 : 2.432 GHz
          Channel 06 : 2.437 GHz
          Channel 07 : 2.442 GHz
          Channel 08 : 2.447 GHz
          Channel 09 : 2.452 GHz
          Channel 10 : 2.457 GHz
          Channel 11 : 2.462 GHz
          Channel 12 : 2.467 GHz
          Channel 13 : 2.472 GHz
          Channel 14 : 2.484 GHz
          Current Channel=8
```

連接無線網路的最後關鍵 (key) 是設定 WEP key。WEP key 有兩種可能長度：40 (不建議)、或 128 bits。對於 WEP-40，你可以用 10 個十六進位數字或 5 個 ASCII 字元表示；對於 WEP-128，則是以 26 個十六進位數字或 13 個 ASCII 字元表示。由於一連串的十六進位數字很容易寫錯，所以 `iwconfig` 提供下列兩種等效格式：

```
# iwconfig eth1 key 123456789A
# iwconfig eth1 key 1234-4568-9A
```

如你所見，第一種格式是連續十個數字，沒有任何分隔符號。第二種格式是每四個數字一組，其間以「-」符號分隔。

對於支援 WEP-128 的驅動程式，其鍵值以 26 個十六進位數字表示：

```
# iwconfig eth1 key 12345678901234567890123456
# iwconfig eth1 key 1234-5678-9012-3456-7890-1234-56
```

如果使用 ASCII 字串來表示 WEP-Key，則必須冠上 `s:` 前綴詞，例如：

```
# iwconfig eth1 key s:wxyz8
# iwconfig eth1 key s:0123456789xyz
```

有些無線網路輪替使用幾組固定的 WEP key，為了支援這種情況，`iwconfig` 容許你預先定義多組 WEP key，並為每組 WEP key 各指定一個索引編號。索引編號是定義在 WEP key 的後面的一對方括號裡，例如：

```
# iwconfig eth1 key 0123-4567-89
# iwconfig eth1 key 9876-5432-01 [2]
# iwconfig eth1 key 5432-1678-90 [3]
```

定義好多組鍵值之後，當你要更換鍵值時，可直接用索引編號來表示你想使用的 WEP key：

```
# iwconfig eth1 key [2]
```

使用 `key on` 或 `key off` 可讓特定的 WEP-key 生效或失效：

```
# iwconfig eth1 key [3] on
# iwconfig eth1 key off
```

除了 WEP-key 之外，還必須選擇 WEP 系統形式。開放式 (`open`) 系統接受明文形式的資料框，而受限 (`restricted`) 系統會丟棄明文形式的資料框。兩種選項都可以配合鍵值索引碼：

```
# iwconfig eth1 key [4] open
# iwconfig eth1 key [3] restricted
```

設定好所要用的 WEP-key 與 WEP 系統形式，你的無線網路介面可能還沒啟動。你可像啟動 Ethernet 網卡一樣來啟動無線網卡：

```
[root@rox]# ifconfig eth1 up
```

啟動無線介面可能要花一段時間，在這段期間，你可以重複使用 `iwconfig` 來查看介面狀態：

```
[root@rox]# iwconfig eth1
eth1      IEEE 802.11g  ESSID:"LINHOME"
          Mode:Managed  Frequency:2.447 GHz  Access Point: 00:00:00:00:00:00
          Bit Rate=54 Mb/s   Tx-Power=20 dBm
          ... (略) ...
```

```
[root@rox]# iwconfig eth1
eth1      IEEE 802.11g  ESSID:"LINHOME"
          Mode:Managed  Frequency:2.447 GHz  Access Point: 00:11:09:BF:BB:9F
          Bit Rate=54 Mb/s   Tx-Power=20 dBm
          RTS thr:off   Fragment thr:off
          Encryption key: (略) Security mode:open
          Power Management:off
          Link Quality=100/100  Signal level=-17 dBm  Noise level=-86 dBm
          Rx invalid nwid:0  Rx invalid crypt:0  Rx invalid frag:0
          Tx excessive retries:0  Invalid misc:4519  Missed beacon:2
```

當 `iwconfig` 的輸出訊息中有顯示基地台的 MAC 位址，就表示無線網卡終於找到了網路。

現在你已經接上無線網路了，只差最後一步：IP 位址。如果你的無線網路有提供 DHCP 服務，現在可以用 `dhclient`（或 `dhcpcd`、`pump`）來取得 IP 位址：

```
[root@rox]# dhclient eth1
Internet Systems Consortium DHCP Client V3.0.1
Copyright 2004 Internet Systems Consortium.
All rights reserved.
For info, please visit http://www.isc.org/products/DHCP

sit0: unknown hardware address type 776
sit0: unknown hardware address type 776
Listening on LPF/eth1/00:0e:35:58:3a:21
Sending on   LPF/eth1/00:0e:35:58:3a:21
Sending on   Socket/fallback
DHCPDISCOVER on eth1 to 255.255.255.255 port 67 interval 6
DHCPDISCOVER on eth1 to 255.255.255.255 port 67 interval 9
DHCPOFFER from 192.168.100.8
DHCPREQUEST on eth1 to 255.255.255.255 port 67
DHCPACK from 192.168.100.8
bound to 192.168.100.234 -- renewal in 120483 seconds.
```

若沒有 DHCP 服務，則應該 `ifconfig` 來設定無線介面的 IP 參數，就像設定 Ethernet 網卡那樣。

設定選擇性功能

有些無線網卡可讓你選擇網路拓樸模式。先前說過，無線網路有兩種拓樸模式，基地台 (Managed) 或對等 (Ad-hoc)。大多數驅動程式會預先將無線網卡設定為基地台模式，而且不容許改變。對於可以改變模式的網卡，你可使用 `iwconfig` 的 `mode` 選項來改變模式：

```
# iwconfig eth1 mode Managed
```

或

```
# iwconfig eth1 mode Ad-hoc
```

有些基地台會要求工作站必須向它們註冊 MAC 位址，在這種情況下，你可能需要用 `ap` 選項將你要註冊的 MAC 位址告訴基地台：

```
# iwconfig eth1 ap 01:02:03:04:05:06
```

無線網路的速度，隨基地台信號強度而變，愈遠愈慢。因此，無線網卡支援多種位元速率，`iwconfig` 提供 `rate` 選項讓管理者選擇固定速率，或是自動調整速率：

```
# iwconfig eth1 rate auto
```

並非每種無線網卡的驅動程式都本小節所說的 `mode`、`rate` 與 `ap` 選項。

故障排除

802.11 網路的「無線」本質，使得它比傳統有線網路更容易出錯。在規劃無線網路的部署時，有幾個議題必須事先考慮。

首先是信號強度。我們當然希望所有用戶端都在有效信號的涵蓋範圍內，但是卻也不想要廣播給全世界知道。信號品質可藉由天線與基地台放置位置來調整，必要時可以另外安裝指向天線 (**directional antenna**) 將電波信號推向信號死角。怎樣才是最好的配置方式，要看實際環境而定，只有實地實驗才能得到最佳效果。

另一項要考慮的因素是信號干擾。802.11b 所用的 2.4 Ghz 是頗為繁忙的頻帶，無線電話、微波爐、對講機都是可能的干擾源。隔壁公司的無線網路如果使用與你相同或相鄰的頻道，則你們倆家的網路也會互相干擾，雖然還不致於造成停擺，但是傳輸效率肯定會大受影響。再一次，解決問題的方法唯有實驗再實驗。

除了物理因素之外，軟體也可能造成問題。驅動程式與網卡不相容，是最常見的問題，避免這種問題的最好方法，是徹底搞清楚你所用的硬體。認清楚晶片組型號，絕對有助於找出正確的驅動程式。

Linux 系統上有許多工具可以幫助你找出晶片組型號。對於 PCI 裝置，可使用 `lspci`（或直接查看 `/proc/pci`）來查看：

```
# lspci -v
... (略) ...
0000:02:02.0 Network controller: Intel Corp. PRO/Wireless 2200BG (rev 05)
  Subsystem: Intel Corp.: Unknown device 2712
  Flags: bus master, medium devsel, latency 64, IRQ 11
  Memory at c0214000 (32-bit, non-prefetchable) [size=4K]
  Capabilities: [dc] Power Management version 2
```

對於 CardBus (PCMCIA) 無線網卡，可利用 `cardctl` 的 `ident` 命令找出答案：

```
[root@localhost ~]# cardctl ident
Socket 0:
  no product info available
Socket 1:
  product info: "Lucent Technologies", "WaveLAN/IEEE", "Version 01.01", ""
  manfid: 0x0156, 0x0002
  function: 6 (network)
[root@localhost etc]#
```

最棘手的是 USB 無線網卡。你可以用 `lsusb` 或直接查看 `/proc/bus/usb/devices` 檔案內容來尋找裝置資訊，不過，不見得有你要的晶片組編號：

```
rox ~ # cat /proc/bus/usb/devices
... (略) ...
T: Bus=01 Lev=01 Prnt=01 Port=04 Cnt=01 Dev#= 3 Spd=480 MxCh= 0
D: Ver= 2.00 Cls=00(>ifc ) Sub=00 Prot=00 MxPS=64 #Cfgs= 1
P: Vendor=0b05 ProdID=1706 Rev= 0.01
S: Manufacturer=ASUS
S: Product=802.11g WLAN + Pen Drive
C:* #Ifs= 1 Cfg#= 1 Atr=80 MxPwr=300mA
I: If#= 0 Alt= 0 #EPs= 2 Cls=ff(vend.) Sub=ff Prot=ff Driver=(none)
E: Ad=81(I) Atr=02(Bulk) MxPS= 512 Iv1=0ms
E: Ad=01(O) Atr=02(Bulk) MxPS= 512 Iv1=0ms
```

不過，如果你有 USB 無線裝置的 Windows 驅動程式，或許可從驅動程式的檔名看出端倪。