Wireless Howto

Roberto Arcomano berto@bertolinux.com v1.7, 23 Agosto 2002

Wireless è una nuova tecnologia nelle schede di rete ad alta velocità (fino a 11 Mbps). Questo documento illustra come configurare Wireless in Linux, problemi di compatibilità, requisiti sulla visibilità tra le antenne e altro ancora. L'ultima versione di questo documento puo' essere trovata su *http://www.bertolinux.com* `http://www.bertolinux.com` .

Contents

1 Introduzione

1.1 Introduzione

Questo documento illustra le reti Wireless, loro configurazione e relative problematiche. Diversamente da come accade per le reti classiche, le reti Wireless richiedono particolari accorgimenti per funzionare. E' necessario avere un minimo di esperienza sulle antenne, sul loro puntamento e così via. Ogni critica (costruttiva) è la benvenuta. Un'ottima fonte é anche questa: *Jean Tourrilhes Wireless Howto* `http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux`

Per qualunque suggerimento o critica scrivetemi su questo *indirizzo email* `mailto:berto@bertolinux.com`

1.2 Copyright

1.3 Traduzioni

Sei libero di tradurre questo documento, devi soltanto rispettare 2 regole:

1. Controllare che non esista gia' un'altra versione del documento nel tuo LDP locale
2. Mantenere la sezione 'Introduzione' (che include i paragrafi 'Introduzione', 'Copyright', 'Traduzioni', 'Ringraziamenti').

Attenzione! Non bisogna tradurre il file TXT o HTML, bensi' il file LYX, cosicche' sia possibile convertirlo negli altri formati (TXT, HTML, RIFF, ecc.): puoi utilizzare l'applicativo LyX scaricabile da *http://www.lyx.org* `http://www.lyx.org` .

Non c'e' bisogno di chiedermi l'autorizzazione! E' sufficiente comunicarmelo.

Grazie per la traduzione.

1.4 Ringraziamenti

Ringraziamenti a *Fatamorgana Computers* `http://www.fatamorgana.com` per l'equipaggiamento hardware e per la sperimentazione.

Grazie anche al *Progetto Pluto* `http://www.pluto.linux.it` per la pubblicazione.

2 Conoscenze di base

2.1 Cos'é il Wireless?

Il Wireless é una nuova tecnologia che può aiutare nella comunicazione tra computers distanti. Si basa su schede Wireless con un TX/RX che opera sui 2.4 GHz, mentre l'interfaccia dal lato software é di tipo Ethernet, con un indirizzo fisico diverso per ogni scheda nel mondo. Potenze di trasmissione tipiche sono 10-20 mW fino a 100mW (vedi standard IEEE 802.11 e FCC/CEPT licenze).

2.2 Qual'é la massima distanza tra le schede radio?

Il requisito più importante, nelle comunicazioni Wireless, é la visibilità senza ostacoli delle antenne: e' necessario "vedere" l'altra antenna (a occhio nudo o con un binocolo) o al massimo é concesso un piccolo albero come ostacolo.

La distanza dipende dall'antenna (eventualmente amplificata): 2-300 metri con una omnidirezionale; 1 km con direttiva; 2-3 km con omnidirezionale amplificata (200mW); alcuni km con parabola. 50-60 km con parabola o direttiva ampificata (alcuni Watts).

Attenzione che non é sempre legale amplificare schede Wireless, in quanto potresti violare le specifiche dell' FCC (America) o del CEPT (Europa) o quelle del proprio stato.

2.3 Qual'e' la differenza tra schede classiche e schede Wireless?

Le schede classiche sono molto semplici da configurare (almeno a livello fisico). La configurazione delle reti Wireless risulta invece molto complessa come anche la loro gestione e relativo debugging... I tipici problemi di installazione hardware, software, debug e cosi' via diventano critici in ambiente Wireless:

1. Devi scegliere la giusta scheda Wireless: ce ne sono si svariati tipi, di svariati rivenditori e con diversi requisiti. Se vuoi creare una piccola rete LAN/WAN devi comprare schede Wireless compatibili IEEE 802.11 con un Access Point.
2. Molte schede hanno un plug PCMCIA, quindi necessitano, come prima cosa, l'installazione dei sorgenti pcmcia per Linux.
3. E' necessario testare le schede con 2 sistemi funzionanti e opportunamente configurati, all'inizio vicini, poi via via sempre piu' lontani.
4. Dovresti testare la configurazione in ogni condizione climatica (tipicamente pioggia).
5. Se tutto funziona puoi essere soddisfatto e rilassarti!

Se installi un ripetitore (macchina Linux con diverse schede Wireless e Wired) potresti avere dei problemi editando la sua configurazione a distanza.

2.4 Cosa devo sapere per configurare una rete Wireless?

Ci sono un certo numero di requisiti da soddisfare.

`requisiti software:`

1. Conoscenze generice di reti classiche: indirizzi IP, netmask, routing... tutto descritto dal generico NET3-4-HOWTO;
2. Conoscenze specifiche di reti classiche: proxy arp, bridging, file system proc, descritte nei documenti Proxy-ARP-Subnet, Bridge Mini-Howto e nei sorgenti di Linux (2.2.x or 2.4.x) sotto Documentation/networking/ ip-sysctl.txt)
3. Conoscenze di base delle reti Wireless: concetti di modalita' di accesso (ADHOC, INFRASTRUCTURE and ACCESS POINT), di canale, terminologia outdoor and indoor e cosi' via: puoi trovare tutto cio' in ogni documento che tratta di Wireless: IEEE standard 802.11, CEPT, ecc.

`requistiti non software:`

1. Esperienza minima di antenne, loro montaggio e puntamento
2. Esperienza in installazione di schede su Pc con particolare attenzione nell'evitare le interferenze elettromagnetiche tra 2 differenti schede Wireless (nel caso sia richiesta l'installazione di più schede).

Infine una grande fortuna!

2.5 Perché mettermi a configurare una rete Wireless e che cosa mi aspetto?

Perché? Perché non sei soddisfatto delle reti classiche!

Con le schede Wireless puoi attraversare giardini, parchi, case (l'importante é poter "vedere" l'altra estremità)

I protocolli di alto livello sono gli stessi delle schede classiche: TCP/IP over Wireless Ethernet-like ma fai molta attenzione alla condivisione risorse di Windows, perché se usi Linux per instradare i pacchetti, non potrai vedere gli altri computer su "Risorse di Rete", in quanto un router (ip forwarder) non fa passare i pacchetti di broacast (vedi protocollo NetBIOS): in tal caso potresti servirti di un server WINS (vedi documentazione di Samba).

Il sistema Wireless ti permette di creare una piccola LAN/WAN con un punto di accesso (magari con accesso ad Internet) e dando connettività a chiunque via etere!

Immagina un paese tutto cablato via radio.

Immagina una rete che può interconnettere tutte gli abitanti di un paese, far loro condividere files, suoni, video ad altissima velocità (comparabile con quella delle reti classiche fino a 11Mbps).

Tutto ciò può essere fatto (ed è già una realtà in alcune città) usando schede Wireless con relativi AccessPoint e macchine Linux che fungono da ripetitori allorché la distanza sia troppo elevata per essere coperta con un solo collegamento (a livello IP come ad esempio un router oppure, se vuoi, a livello data-link, con il supporto bridging per Linux , vedi *http link* `http://openrock.net/bridge` o *ftp link* `ftp://openrock.net/bridge`)

2.6 Quali schede Wireless vediamo in questo Howto?

In questo Howto iniziamo con una configurazione generica (per introdurre le reti Wireless), in seguito descrivo un esempio per ogni scheda di cui ho esperienza diretta, con piccoli trucchi per migliorare le performance ed evitare conflitti.

`Lista schede Wireless:`

1. Proxim Symphony - *http://www.proxim.com* `http://www.proxim.com`
2. Webgear AviatorPRO 2.4 (supporto pcmcia richiesto) - *http://www.webgear.com* `http://www.webgear.com`
3. Lucent Wavelan I, II, Orinoco - *http://www.lucent.com* `http://www.lucent.com` and *http://www.orinocowireless.com* `http://www.orinocowireless.com`
4. Cabletron - *http://www.cabletron.com* `http://www.cabletron.com`
5. YDI am930_isa - *http://www.ydi.com* `http://www.ydi.com`
6. Siemens Radio Modem (Dect) - *http://www.siemens.com* `http://www.siemens.com`

Per una lista più completa vedi *Jean Tourrilhes Wireless Howto* `http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux` .

I Siemens Radio Modem non sono propriamente delle schede Wireless IEEE 802.11, ma sono dei modems da collegare sulla seriale (frequenza operativa 1800 MHz, tecnologia DECT). Vedi Appendice B.

2.7 Quanto costano?

Le suddette schede hanno un costo relativamente basso: partono da poche centinaia di migliaia di lire fino ad arrivare ad alcuni milioni per gli Access Point predisposti per schede 2 Wireless (Lucent, ad esempio) funzionanti come bridge.

3 Informazioni tecniche sul Wireless

Qui di seguito riporto alcune informazioni di base necessarie per comprendere la tecnologia Wireless

3.1 Livello fisico

A livello fisico ISO/OSI possiamo avere 3 tipi di specifiche:

1. FHSS, Frequency Hopping Spread Spectrum
2. DSSS, Direct Sequence Spread Spectrum
3. Infrarossi, che qui non vediamo. Vedi relativo Howto Linux-IR-HOWTO

3.2 Configurazioni

2 tipi di configurazioni:

1. AdHoc mode (anche noto come Independent mode), dove si hanno reti indipendenti ognuna con un BSS (Basic Service Set) . Ogni stazione della rete utilizza lo stesso BSS.
2. Infrastructure mode, dove un certo numero di reti (ognuna con un proprio BSS) può comunicare con le altre tramite il proprio Access Point (uno per ogni BSS) per creare un ESS (Extended Service Set). Vi sono anche delle funzionalità di roaming che permettono ad una stazione di scegliere l'Access Point più vicino cui allacciarsi.

Adhoc mode é il metodo più semplice (ed anche il meno scalabile!) e permette a diversi hosts di comunicare direttamente uno con l'altro. Il requisito (un po' restrittivo) é che ogni macchina debba vedere tutte le altre perché le cose funzionino (questo in teoria, perché in pratica questo problema può essere risolto a livello IP! Vedi Par 5.4).

```
                    Adhoc mode

                  A - - - - - C
                    \       /
                  |   \   /   |
                        /\
                  |   /    \  |
                    /       \
                  B - - - - - D
```

In Infrastructure mode vengono utilizzati gli Access Point attraverso cui tutte le stazioni devono connettersi per comunicare con gli altri.

```
Infrastructure mode
                                 ESS

     A - - - | - Access Point - -  Access Point - | - - - D

     B - - - |   BSS1                   BSS2      | - - - E

     C - - - |                                    | - - - F
```

A, B e C non possono vedere D,E and F, direttamente, ma possono comunicare con loro usando lo stesso ESS. Nota: A,B and C potrebbero anche non vedersi tra di loro, ma grazie all'Access Point comunicano lo stesso.

Vengono inoltre utilizzati temini come indoor e outdoor per distinguere le reti a corto raggio da quelle a lungo raggio o più propriamente per interni e per esterni.

3.3 Compatibilità

Bisogna tenere conto che esistono un gran numero di schede Wireless nel mondo e non tutte possono comunicare tra di loro.

Per dialogare e' necessario che utilizzino:

1. la stessa modalità di accesso: Adhoc o Infrastructure
2. lo stesso livello fisico: DSSS o FHSS
3. lo stesso protocollo (ad esempio Proxim utilizza il suo protocollo proprietario OpenAir e, benché il livello fisico sia FHSS, non può comunicare con le altre schede di tipo FHSS).

3.4 Devo usare Adhoc o Infrastructure?

E' noto che gli Access Point sono molto utili e evitano molte problematiche, ma hanno lo svantaggio di costare non poco. In teoria, per una rete molto concentrata conviene usare Infrastructure mode, mentre per pochi si può tranquillamente optare per Adhoc: in effetti perché spendere soldi inutilmente per poche macchine?

Vale comunque sempre la regola: mangi come spendi!

3.5 Una macchina Linux può fungere da AccessPoint?

Ottima domanda!

Di recente e' stato prodotto un nuovo tipo di schede che permettono questa caratteristica: Prism2

Per saperne di piu' si veda il sito web *http://people.ssh.com/jkm/Prism2/* `http://people.ssh.com/jkm/Prism2/`

4 Gli strumenti

4.1 Hardware

Hai bisogno di un Linux Box (486 o, meglio, un Pentium 100+ con 16MB+ ram), della scheda Wireless, di un antenna per i 2.4GHz (vedi Par 2.2). Ovviamente hai bisogno di tutto ciò anche nell'altra estremità (con Win9x o WinNT se preferisci...) in quanto dobbiamo simulare una comunicazione.

4.2 Software

Hai bisogno di:

1. Sorgenti recenti del kernel di Linux (2.2.x). Vedi dopo

2. Sorgenti recenti del pacchetto pcmcia (pcmcia-cs), nel caso tu abbia comprato una scheda pcmcia. Vedi dopo

3. Driver della scheda Wireless: se non ce l'hai puoi scaricarlo dal sito dove hai preso la scheda o dal sito del costruttore. Se proprio non lo trovi puoi provare a cercare all'indirizzo *Jean Tourrilhes Wireless Howto* `http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux` .

Se non lo trovi neache qui devi convertire un driver di Windows per farlo girare sotto Linux!! (buona fortuna!)

Dopodiché devi ricompilare il kernel, ricompilare i sorgenti pcmcia (se richiesto) e finalmente compilare il driver Wireless. Tieni conto che questa é un situazione del tutto generale, in molti casi ti basterà seguire il passo 3 oppure i passi 1 e 3, dipende dal driver che hai.

5 Generalità sulla configurazione di una rete Wireless.

5.1 Passi fondamentali

Una volta che hai recuperato il materiale e hai ricompilato tutto il necessario sei pronto per seguire i 3 passi per la configurazione di una rete Wireless.

1. Configurazione di basso livello: é necessario far riconoscere al kernel la nostra scheda Wireless (cose del tipo ioport, interrupts, dma...): devi cercare un qualche tipo di messaggio, sui file di log, che ti confermi l'avvenuto riconoscimento e la relativa configurazione.

2. Configurazione a livello Data-link: per ogni particolare scheda Wireless esiste un'utility in grado di settare i tipici valori Wireless data-link. Ad esempio Proxim Symphony utilizza "rl2cfg" mentre nelle schede pcmcia i settaggi sono presenti nei files di configurazione del supporto pcmcia. Devi settare tutte le tue schede Wireless in modo coerente, altrimenti non potranno dialogare.

3. Settaggi IP. A questo dovresti essere in grado di utilizzare i comandi ifconfig e route per cambiare i settaggi IP.

4. Trucchetti per migliorare le performance ed evitare conflitti. Adesso la tua rete fondamentalmente funziona: puoi aggiustare alcuni settaggi riguardanti proxy arp, icmp echo redirect, bridging, cambio dei canali e così via per ottimizzare la configurazione ed evitare conflitti uccidi-banda!

N.B.: passi 1, 2 e 3 corrispondono ai livelli 1, 2 and 3 dello standard ISO/OSI, mentre il passo 4 é un addendum per risolvere alcune problematiche, tra cui quelle generate dall'adozione della maschera 255.255.255.255. Infatti la maschera 32 bit viola lo standard TCP/IP forzando una rete ad usare lo stesso indirizzo per il broadcast e per l'host, mentre l'indirizzo di rete non esiste.

Qualcuno potrebbe criticare questo punto di vista, ma se seguissimo alla lettera lo standard inizieremmo a perdere una marea di indirizzi IP (sempre più preziosi); per ogni sottorete 2 indirizzi sono riservati (network and broadcast) e comunque, seguendo lo standard, non si avrebbe certamente la necessaria flessibilità sull'assegnazione di indirizzi IP alle interfacce richiesta dai sistemi Wireless (allocazione di tipo geografica!). Per maggiori informazioni vedi Appendix A.

Si può infine notare come il passo 2 non sia presente nelle schede Wired, in quanto non vi sono particolari settaggi da operare.

5.2 Configurazione di basso livello

E' il classico problema dell'amministratore del sistema: far riconoscere al PC il proprio hardware.

Le schede Wireless sono più complesse sotto questo punto di vista, perché molte hanno il plug Pcmcia e quindi prima di tutto é necessario "far vedere" al kernel il proprio adattare Pcmcia, in seguito sarà possibile provare ad installare il driver specifico per la scheda Wireless Pcmcia.

`Quindi nelle configurazioni Pcmcia è necessario:`

1. installare i sorgenti del kernel, scaricandoli da *http://www.kernel.org* `http://www.kernel.org` e scompattandoli su /usr/src/linux (vedi comandi tar and gzip)
2. installare i sorgenti per il supporto pcmcia, da *ftp://projects.sourceforge.net/pub/pcmcia-cs* `ftp://projects.sourceforge.net/pub/pcmcia-cs` e scompattarli su /usr/src/pcmcia (vedi sempre utilities tar and gzip)
3. configurare e ricompilare il kernel: leggi attentamente il README presente nella directory del Kernel (/usr/src/linux)
4. configurare e ricompilare i sorgenti pcmcia: sotto /usr/src/pcmcia usare configure eppoi make. Assicurati che ci sia il driver che ti interessa, altrimenti dovrai installarlo seguendo le relative istruzioni (di solito basta scompattare con "tar zxvf driver.tgz" il driver pcmcia direttamente sotto la directory pcmcia). Dopodiché digita "make all" per compilare. Alla fine digita "make install".
5. Dopo aver installato correttamente i drivers pcmcia potrai trovare i relativi file di configurazione sotto la directory /etc/pcmcia .

`mentre per la configurazione senza Pcmcia:`

1. Se il driver della tua scheda non é presente (in effetti non lo é mai!) nel Kernel, dovrai scaricarlo, installarlo e ricompilarlo in una nuova directory.

Una volta che conosci il nome del driver é il momento di caricarlo: nel caso di Pcmcia ci penserà il demone relativo (attivabile tramite /etc/rc.d/init.d/pcmcia start per le distribuzioni RedHat), mentre per le altre schede basterà dare "modprobe module_name options". Tra le opzioni vi sono ioport, irq e i settaggi datalink relativi al driver Wireless (vedi Par 5.3). Ad ogni modo ci sono una serie di strumenti molto comodi per controllare il riconoscimento a basso livello del driver:

1. "tail /var/log/messages" che mostra le ultime informazioni scritte sul log di sistema (syslog)
2. "dmesg" for ulteriori info sul log
3. /proc directory: files ioports, devices, irq e sottodirectories specifiche per il driver.

5.3 Configurazione a livello Data-link

`Cosa significa?`

Le reti classiche hanno bisogno solo di essere fisicamente connesse le une con le altre e di avere i giusti settaggi TCP/IP. Al contrario le reti Wireless networks richiedono settaggi Data-link che indicanti:

1. A che tipo di rete Wireless appartengo? (Adhoc o Infrastructure)

2. Che canale devo usare?
3. A che sottorete (BSSID) appartengo e qual'é il mio ESSID?
4. La mia comunicazione é protetta da qualche algoritmo crittografico? Qual'è la chiave?

Ci sono quindi una serie di settaggi da configurare, la ragione di ciò deriva dall'architettura Wireless: potrebbe esserci qualcuno, nel raggio di copertura Wireless, che potrebbe intercettare dei pacchetti e usare servizi relativi ad un altra rete wireless solamente puntanto la propria antenna nella giusta direzione e settando gli opportuni parametri TCP/IP.

Inoltre più reti Wireless (magari ognuna relativa ad un piano di un edificio) potrebbero interferire una con l'altra.

Ecco allora:

1. Opzioni a load-time module: "modprobe ray_cs essid='LINUX'" per esempio o
2. Utilities a run-time driver: "rl2cfg eth1 master".

5.4 Settaggi IP + trucchetti

Questo é il terzo problema che dobbiamo affrontare. Qui i problemi arrivano quando ci si evolve verso una rete più grossa.

`Ricordati che la rete Wireless non ti stressa se non sei tu a stressarla!`

5.4.1 Una configurazione semplice

```
                      Tutti gli hosts si "vedono"

                              A - - - - - C
                                \       /
                              |   \   /   |
                                   /\
                              |   /    \  |
                                /       \
                              B - - - - - D
```

Una configurazione come questa é molto semplice e non richiede particolari attenzioni (almeno a livello IP): hai soltanto bisogno di assegnare un indirizzo IP ad ogni host e una maschera generale coerente con tutte le altre macchine.

5.4.2 Una configurazione più complessa

```
                       A non vede B direttamente

                               A <- - - -
                             NO\        |
                           TALK\        C
                               \        |
                               B <- - - -
```

In questo caso A e B possono comunicare tra loro soltanto tramite C.

Se la rete é in infrastructure mode e C é l'Access Point allora tutto ok. In Adhoc mode é invece designare un host per essere un "master" (so che il termine non é molto formale, ma l'importante é capirsi!), un host che sia cioé in grado di creare un BSS e a cui gli altri possano collegarsi specificando il relativo BSSID.

In questo caso la connettività globale viene raggiunta a livello IP: A e B usando la stessa interfaccia di C, quindi se provi anche solo a fare ping da A verso B riceverai una marea di pacchetti ICMP REDIRECT da C, con cui C t'informa che la destinazione é già sulla rete da cui proviene la richiesta.

Soluzione: digita "echo 0 > /proc/sys/net/ipv4/conf/ethx/send_redirects" (dove ethx é l'interfaccia su C verso A e B) per annullare il problema.

Un altro problema: che maschera assegnare ad A, B e C? Se si da una maschera ad A che includa B non funziona nulla, perché A non utilizzerà C come gateway ma farà la richiesta ARP con indirizzo MAC destinazione sconosciuto.

Potresti dire: uso il proxy arp! Peccato che il proxy arp serva per rispondere al sorgente solamente nel caso in cui il destinatario appartenga ad una diversa interfaccia rispetto al sorgente e questo non é il caso.

Conviene quindi settare la più piccola maschera possibile (Win9x permette 255.255.255.254, WinNT invece 255.255.255.248), e devi essere sicuro che A and B non appartengano alla stessa sottorete.

`Esempi:`

1. IP(A) = x.y.z.2/31, IP(B) = x.y.z.3/31. Così non va perché A chiede B nella sua sottorete (ARP request) e C non può rispondere perché, per lui, A e B appartengono alla stessa interfaccia (quindi niente proxy arp).
2. IP(A) = x.y.z.1/31, IP(B)= x.y.z.2/31. Adesso si, perché A chiede a C (A manda i suoi pacchetti a B usando l'indirizzo MAC di C) per arrivare a B.

In generale, con una maschera 255.255.255.254, il giochetto funziona tra 2 IP che NON differiscono soltanto per l'ultimo bit.

Mi rendo conto che tutto ciò é una forzatura del TCP/IP, ma é l'unico metodo per ottenere un elevato livello di flessibilità.

Nota: Usando un Access Point (rete in Infrastructure mode) non esiste più il problema dei pacchetti ICMP REDIRECT perché il problema viene già risolto a livello data-link (praticamente tutti gli Access Point sono dei bridge...). Ma gli Access Points sono costosi (qualche milione o più) ed é più economico usare un P133, 32MB Ram per forwardare i pacchetti, anche con 2 o più schede di rete.

5.4.3 Accesso ad Internet

```
A - - - - - C - - Internet
  \       /
|   \   /   |
      /\
|   /    \  |
  /       \
B - - - - - D
Ci sono una serie di possibilità:
```

1. C'é un unico indirizzo IP pubblico. Devi soltanto settare indirizzi IP privati (192.168.x.y per esempio) sulla rete Wireless e abilitare, su C, il forwarding e il masquering (NAT). A, B e D utilizzeranno C come default Gateway.

2. Possiedi una maschera pubblica (classe) visibile da Internet e C é il tuo default GW verso Internet per la tua rete. Hai bisogno, come prima, di abilitare il forwarding su C e di settare default GW su A, B ed D facendolo puntare su C.

3. Possiedi una maschera pubblica (classe) visibile da Internet e C NON é il tuo default GW verso Internet, Hai 2 possibili soluzioni: modificare il tuo default GW facendolo puntare a C per la tua rete. Simmetricamente C punterà sul tuo default GW per andare su Internet. Altrimenti potresti abilitare il proxy arp su C (echo 1 > /proc/sys/net/ipv4/conf/ethx/proxy_arp dove ethx é l'interfaccia verso il default GW)e settare default GW su C per puntare sul default GW per andare su Internet. Il Proxy arp é una forzatura TCP/IP ma lavora bene.

5.4.4 Reti miste: Wired e Wireless

```
        Internet
               \
                \        E
                 \     /
                  \  /      Wireless
       A - - - - - C - - F
         \       /   \
       |   \   /   |   \
Wired        /\          G
       |   /    \  |
         /       \
       B - - - - - D
```

Adesso C unisce 2 reti: a destra Wireless e a sinistra Wired.

In aggiunta puoi avere l'accesso ad Internet ed in totale hai bisogno di 3 schede di rete su C.

Che indirizzi devi assegnare alle macchine? 2 soluzioni possibili:

1. Dividere la rete in 2 sottoreti: ad esempio 192.168.1.0/24 e 192.168.2.0/24. Questa soluzione é molto veloce ma non é assolutamente scalabile né flessibile (almeno in ambito Wireless), perché se utilizzi indirizzi Internet pubblici rischi di perderne una marea.

2. Abilitare il proxy arp su C per tutte e 2 le interfacce. I parametri di rete sono, in questo caso, gli stessi (net address and netmask) per Wireless e Wired, ma con il proxy-arp abilitato é possbile scegliere quali indirizzi IP sono su Wired e quali su Wireless.

```
Esaminiamo ora la soluzione 2
```

Ad esempio considera di avere una sottorete pubblica su Internet del tipo x.y.z.0/24.

```
Le interfacce sono:
```

1. ifconfig eth0 x.y.z.C netmask 255.255.255.255 (Wired)

2. ifconfig eth1 x.y.z.C netmask 255.255.255.255 (Wireless)

3. ifconfig eth2 x.y.z.C netmask 255.255.255.255 (to Internet)

`Le routes statiche su eth2:`

1. route add IPGW dev eth2
2. route add default gw IPGW

Queste route servono per indirizzare tutto Internet verso il nostra Default GW: come puoi notare é necessario prima dire a Linux che il router é sull'interfaccia eth2, poi mandare la default route verso di esso.

`Routes statiche su eth0:`

1. route add x.y.z.A dev eth0
2. route add x.y.z.B dev eth0
3. route add x.y.z.D dev eth0

Hosts A,B e D sulla rete Wired

`Routes statiche su eth1:`

1. route add x.y.z.E dev eth1
2. route add x.y.z.F dev eth1
3. route add x.y.z.G dev eth1

Hosts E,F e G sulla rete Wireless.

Nota che la flessibilità é molto elevata, però devi settare manualmente ogni hosts (al limite puoi destinare un micro-classe su un'interfaccia).

6 Setup

Qui riporto alcuni esempi (spero utili!) per configurare le schede Wireless più diffuse.

6.1 Informazioni generali sul Setup

Le schede Wireless hanno un interfaccia molto simile alle schede Wired, quindi dovrei aggiungere, come al solito delle entries nel file /etc/conf.modules contenenti:

1. "alias ethx module", dove ethx é l'intefaccia che vuoi attribuire alla scheda Wirelss e module é il nome del modulo del kernel.
2. "options module io=0xAAA irq=I ...", dove 0xAAA é l'indirizzo io di base da assegnare alla scheda, I é l'IRQ e così via per gli altri parametri.

Dopo questo potrai direttamente usare i comandi ifconfig route per configurare i settaggi IP della tua scheda.

6.2 Proxim Symphony

Tipo di rete: FHSS,solo Adhoc e con protocollo proprietario OpenAir.

Web site: *http://www.proxim.com* `http://www.proxim.com` dove devi scaricare la documentazione e il driver per Linux e Win9x.

Una volta scaricato il driver:

1. usa tar per scompattare su una directory vuota
2. usa make come aiuto
3. usa make modules; make modules_install per installare il driver rlmod.o e la relativa utility rl2cfg.
4. Per lanciare il driver (after modified /etc/conf.modules: see Par.6.1.) basta usare il comando ifconfig

Utility rl2cfg (per l'aiuto digita man rl2cfg dopo il passo 3) ti permette di cambiare i settaggi data-link level:

1. "rl2cfg dev ethx sta" per settare la scheda come station (Slave)
2. "rl2cfg dev ethx msta" per settare la sceda come master station (Master)
3. "rl2cfg dev ethx alt" per settare la scheda in modo automatico

Questo é tutto ciò di cui hai bisogno per settare la scheda.

6.3 Webgear Aviator 2.4 and AviatorPro

Tipo di rete: FHSS, solo Adhoc per Aviator 2.4 e anche Infrastructure per AviatorPro.

Queste schede richiedono più lavoro perché devi compilare il supporto Pcmcia.

Web site: *http://www.webgear.com* `http://www.webgear.com` .

Per configurare:

1. Scarica i sorgenti pcmcia e scompattali su /usr/src/pcmcia (vedi Par 5.2)
2. Scarica il driver da *http://www.webgear.com* `http://www.webgear.com` and digita "tar zxvf driver.tgz" sulla directory /usr/src/pcmcia.
3. riconfigura il supporto pcmcia (vedi Par 5.2)
4. Seguendo le istruzioni del driver devi aggiungere in fondo al file /etc/pcmcia/config.opts questa entry: "source ./ray_cs.opts".
5. Nota che nel file /etc/pcmcia/ray_cs.opts c'é una linea tipo: "module "ray_cs" opts "...". Devi modificare i settaggi a livello data-link presenti nel campo "opts" .

`Argomenti:`

- pc_debug=x , dove x é il livello di log.
- net_type=x, x=0 per AdHoc, x=1 per Infrastructure.
- essid=x, x é l'ESSID

Finalmente, per verificare la configurazione (con pc_debug>0), vedrai dei messaggi sulla console tipo questi: "network started" nel caso in cui stai creando una nuova rete Wireless network created e "network joined" se ti stai collegando ad una già esistente.

Puoi controllare il file /proc/ray_cs per aiutarti: il campo BSSID ti dice a quale sotto rete appartieni, se é nullo c'é qualcosa che non va!

6.4 Lucent Wavelan I, II, Orinoco products and Cabletron

Tipo di rete: DSSS, Adhoc e Infrastructure.

I prodotti Lucent sono molto professionali.

Web site: *http://www.lucent.com* `http://www.lucent.com` e *http://www.orinoco.com* `http://www.orinoco.com` .

Il Setup é analogo a quello delle WebGear: i passi 1,2,3 sono simili.

Devi poi aggiungere al file /etc/pcmcia/config.opts l'entry: "module "wavelan_cs" opts " ..."" per i modelli Lucent Wavelan I e "module and "wavelan2_cs" opts " ..."" per i modelli Lucent Wavelan II e Orinoco.

Sotto la voce opts dovreai specificare:

1. port_type=x, dove x indica Adhoc(3) o Infrastructure(1)
2. channel=x, x=canale, opzione rilevanto solo in AdHoc mode.
3. transmit_rate=x, per impostare la velocità: attenzione a questo settaggio per la compatibiltà con le schede Cabletron.

Note: In teoria é possibile installare 2 schede Lucent su una macchina Linux, una in Adhoc mode e l'altra in Infrastructure mode. Il problema é che soltanto una funziona in quanto, durante il lancio dei servizi pcmcia, tutte e 2 le schedeverranno settate con gli stessi parametri data-link (quindi entrambe in Adhoc o in Infrastructure mode). Quello che si può fare é creare un modulo (o un programma user) che permetta di cambiare i settaggi data_link relativi all'access mode e al canale a run-time .

La funzione del parametro canale serve ad evitare interferenze con altre schede Wireless.

I driver Lucent possono anche essere usati per le schede Cabletron *http://www.cabletron.com* `http://www.cabletron.com`

6.5 YDI

Tipo di Rete: DSSS, Adhoc e Infrastructure.

YDI vende schede molto professionali insieme a relative antenne, amplificatori e altro ancora.

Web site: *http://www.ydi.com* `http://www.ydi.com`

`Per installare:`

1. usa tar per scompattare il driver in una directory vuota.
2. digita make per compilare.
3. usa make install per installare il driver am930_isa e la relativa utility wlanctl

Una volta fatto puoi scegliere se digitare i comandi a mano con "wlanctl" o lanciare uno degli scripts "scripts/wlan" or "scripts/rc.wlan" per configurare automaticamente la tua scheda.

Nella configurazione a mano i principali settaggi sono:

1. "wlanctl scan ..." per cercare un BSS già presente.
2. "wlanctl netlist" per mostrare i BSSs trovati con il comando "wlanctl scan ...".
3. "wlanctl bsscreate ... ssid" per creare una buona rete con il parametro ssid.
4. "wlanctl bssjoin bssid" per "raggiungere" la rete specificata dal relativo bssid.
5. "wlanctl authen" e "wlanctl assoc" a scopo di autenticazione.

7 FAQ - Frequently asked questions

D1: Qual'é la differenze tra il BSSID e l'ESSID e quando devo esare l'ESSID?

R1: Il BSSID é un numero da 48 bit usato per identificare l'area ristretta cui ci si trova corrispondente al BSS, dove tutti gli hosts dialogano direttamente tra di loro (eventualmente con un Access Point), mentre l'ESSID é una stringa a lunghezza variabile che permette a differenti reti BSS di dialogare fra loro impostando un identificativo comune noto come, appuntom ESSID. ESS stà per Extended Service Set (ESS). Esiste un solo Access Point per ogni BSS e tutti gli AccessPoint (relativi a sottoreti diverse) dialogeranno tra loro soltanto se avranno lo stesso ESSID. Quindi hai bisogno dell'ESSID se hai una rete già di un certo tipo, con almeno 2 Access Points.

D2: Quale Access Point devo comprare?

R2: Quello che costa meno (a parità di prestazioni!): ciò che é importante é che l'AccessPoint e le schede che stai usando utilizzino protocolli compatibili: vedi Par 3.3 Compatibilità.

D3: A che serve il settaggio del canale?

R3: Quando hai diversi BSS (e magari di diversi standard) potresti aver dei problemi di interferenza: cambiando il canale opportunamente sull'AccessPoint o sulla scheda (in caso di AdHoc mode) puoi attenuare tali disturbi.

D4: Perché non riesco ad impostare il canale su un host che funziona in Infrastructure mode?

R4: Perché in Infrastructure mode il canale viene sempre deciso dall'Access Point.

8 Links

8.1 Software Free

- *progetto Linux-wlan* `http://www.linux-wlan.org/`
- *Jean Tourrilhes Wireless Howto* `http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux`

8.2 Links Commerciali

- *Fatamorgana Computers* `http://www.fatamorgana.com`
- *sito web Lucent* `http://www.lucent.com`

- *sito web YDI* `http://www.ydi.com`
- *sito web Siemens* `http://www.siemens.com`

9 Appendix A - Netmask 255.255.255.255, proxy arp and bridging

Qui vediamo alcuni vantaggi nell'utilizzare Linux per le reti Wireless.

Linux permette di specificare la netmask 255.255.255.255 per un'interfaccia: questo può aiutare in ambito Wireless perché permette, successivamente, di assegnare un indirizzo IP verso un'interfaccia piuttosto che un'altra. Ad esempio puoi specificare la macchina A sull'interfaccia eth0, la macchina B sull'interfaccia eth1 e così via.

Tutto ciò non ha particolary effetti collaterali.

Inoltre puoi utilizzare le funzionalità di proxy arp tramite il file "/proc/sys/net/ipv4/conf/ethx/proxy_arp" dove ethx é l'interfaccia desiderata.

Se per esempio esegui il comando "echo 1 > proxy_arp/proc/sys/net/ipv4/conf/ethx/proxy_arp" abiliterai il proxy arp per l'interfaccia ethx, mentre con "echo 0 > proxy_arp/proc/sys/net/ipv4/conf/ethx/proxy_arp" lo disabiliterai.

Cos'é il proxy_arp? In maniera sintetica possiamo dire che il proxy arp é una funzionalità che ti aiuta quandi vuoi che un router linux risponda ad una richiesta ARP (che "vede passare" su un'interfaccia) nel caso in il destinatario é su un'altra interfaccia del router linux.

Esempio:

```
192.168.1.1 ---- 192.168.1.2 Linux router 192.168.2.2 ----192.168.2.1
```

Per far funzionare questa rete bisogna:

```
Senza proxy-arp
```

1. Settare su 192.168.1.1 192.168.1.2 come gateway
2. Settare su 192.168.2.1 192.168.2.2 come gateway
3. pingare da una estremità all'altra con successo.

```
Con proxy-arp
```

1. Su 192.168.1.1 settare 192.168.1.2 come gateway
2. NON settare il gateway su 192.168.2.1 ma abilitare il proxy_arp sull'interfaccia verso destra
3. pingare da una estremità all'altra con successo.

Il proxy arp permette al router linux di rispondere quando si vede arrivare un'ARP request dall'host 192.168.2.1: il router dice di "avere" lui stesso l'indirizzo di destinazione 192.168.1.1. Dopodiché quando 192.168.2.1 comincia a mandare i pacchetti ICMP (come qualunque altro pacchetto), il nostro router LInux sa che deve mandarli all'host 192.168.1.1.

Nelle reti Wireless il proxy arp può quindi aiutarti quando hai tanti router Linux e non vuoi perdere il tuoi tempo a settare per ogni host le routes statiche; permette cioé un certo grado di trasparenza.

Infine puoi sperimentare il bridging sotto Linux:

1. Installando un kernel recente
2. scaricando le ottime utilities per il bridging da *http link* `http://openrock.net/bridge` o *ftp link* `ftp://openrock.net/bridge`

Bridging dovrebbe essere ancora più comodo del proxy arp.

10 Appendix B - Siemens DECT Radio Modem

Cosa sono? Questi 2 componenti non sono delle vere schede per PC ma sono semplicamente dei modems che si attaccano alla seriale come dei normali devices esterni.

```
Host1-seriale - RadioModem1 - - - - - RadioModem2 - seriale-Host2
```

Come li connetto?

Possiamo modellare i 2 radiomodems com un cavo Null Modem:

```
Host1-seriale - - CAVO NULL MODEM - - seriale-Host2
```

In questo modo ci sono 2 modi possibili per sfruttarne le potenzialità:

1. Linux con Windows, Linux utilizza un demone che ascolta le chiamate ppp, mentre Windows utilizza una connessione di Accesso Remoto per chiamare Linux.
2. Linux con Linux, dove vengono simultaneamente lanciati 2 scripts che eseguono una connessione ppp.

Per la modalità 1 puoi usare questo script per Linux:

"/usr/sbin/pppd -detach lock idle 300 crtscts connect "/usr/sbin/chat -v TIMEOUT 5 AT OK AT OK AT OK AT OK" IPLINUX:IPWINDOWS /dev/ttySx 115200 disconnect "/usr/sbin/chat -v AT OK" ms-dns IPDNS"

dove:

- /dev/ttySx é la porta seriale,
- IPDNS é l'indirizzo IP del tuo Server DNS,
- IPLINUX é l'indirizzo IP di Linux e IPWINDOWS quello di Windows.

Tale scripts é necessario per "far credere" a Windows che sulla seriale sia collegato un modem reale!!

Su Windows devi creare una connessione in Accesso Remoto con crtscts abilitato, velocità a 115200 e con uno stupido numero da chiamarel (richiesto da Accesso Remoto ma assolutamente non usato).

Con 2 macchine Linux devi soltante lanciare, per ogni macchina uno script come questo:

"/usr/sbin/pppd passive local crtscts IPLINUX1:IPLINUX2 /dev/ttySx 115200 noauth persist"

dove dovrai invertire IPLINUX1 con IPLINUX2 sull'altra macchina.

Nota che puoi anche abilitare l'autenticazione, sia con la modalità Linux-Windows che con quella Linux-Linux.