

Wireless on Linux

Una panoramica delle schede wireless piu' diffuse e dei moduli necessari per usarle

Al giorno d'oggi, le reti senza fili costituiscono una realtà molto diffusa nelle case (e non solo) di moltissimi utenti. La tecnologia dominante in ambito informatico è senz'altro rappresentata da quell'insieme di dispositivi e protocolli noti ufficialmente come **standard IEEE 802.11**, nelle sue numerose varianti, modifiche e integrazioni che si sono succedute dal 1999 ad oggi, ma conosciute ai più semplicemente come "wireless" oppure "wi-fi". In questo articolo analizzeremo lo stato (attuale, ma con un breve sguardo al passato) del

supporto a questa realtà da parte del più diffuso sistema operativo libero in ambito desktop: GNU/Linux.

Oramai infatti l'utilizzo di tali dispositivi è consentito dalla presenza di numerosi *drivers* (software che dialogano con l'hardware e ne pilotano le azioni), nonostante il loro stadio di sviluppo non sia omogeneo, e dalla nascita di varie applicazioni grafiche che semplificano le operazioni di connessione ai meno esperti.

Tutti questi drivers sono partiti in tempi diversi ed in modo pressochè indipendente, ma recentemente sono stati tutti raggruppati nel progetto **Linux Wireless**, sul cui sito web (<http://www.linuxwireless.org>) si possono trovare le ultime notizie sull'argomento.

Da premettere che le aziende produttrici di hardware di tale segmento (ma non solo), sebbene in grandissimo numero, utilizzano grosso modo sempre gli stessi tipi di *chipset*, che in realtà rappresentano il "cervello" della scheda, e i cui produttori sono in quantitativo molto minore, nell'ordine di un paio di decine. Accade quindi molto frequentemente che due adattatori, progettati e costruiti da due aziende completamente differenti, magari una di fama mondiale e l'altra di reputazione molto più modesta, siano agli occhi del

sistema operativo molto simili, se non addirittura identiche; tale situazione si presenta anche con adattatori che utilizzano differenti meccanismi e modalità per il collegamento con il resto della macchina: è il caso dei *bus PCI, PCI-Express* (e le loro varianti per notebook o dispositivi embedded: *MiniPCI* e *Mini PCI-Express*), *CardBus/PCMCIA, ExpressCard, PC-104* (quest'ultimo nato e diffuso principalmente in ambito industriale) et similia. Questo consente di utilizzare un unico driver per pilotare una grande varietà di hardware di produttori diversi, che tuttavia presentano lo stesso chipset o comunque appartenenti alla stessa famiglia, il tutto con minimi "aggiustamenti" nel codice sorgente. Per questo motivo, di seguito parleremo esclusivamente dei chipset, dei loro produttori e dei driver che ne consentono l'utilizzo.

Chiudendo questa parentesi "metodologica", veniamo al sodo: i chipset più diffusi sono sicuramente quelli prodotti dalle aziende **Atheros, Intel** e **Broadcom**, che possiamo goliardicamente definire "*il Buono, il Brutto e il Cattivo*", rifacendoci al titolo di un film di qualche decennio fa. Infatti, **Atheros** ha da sempre reso disponibile un driver per il proprio hardware, anche se purtroppo costituito da una parte proprietaria (l'*HAL, hardware abstraction layer*, che si occupava del "dialogo" reale con l'hardware) e per questo mai inclusi nel kernel tree, ma almeno funzionale, completo e versatile: stiamo parlando del progetto **Madwifi**. Esso consentiva l'uso non solo della consueta modalità *Managed* (quella utilizzata per collegarsi ad un *access point* preesistente, ad esempio per la normale navigazione su Internet) ma anche delle più "esclusive" modalità *Master*, necessaria per creare un *access point*, e della *Monitor*, in cui la scheda, sotto la guida di apposito software, "capta" il traffico circostante senza emettere

alcun pacchetto, permettendo di capire ad esempio se la propria rete wireless fuoriesce dai confini domestici o aziendali (il che secondo la legge italiana è illegale, in mancanza di un'autorizzazione). Recentemente si è avuto un cambiamento radicale nella realtà di questi chipset, articolato in tre grandi filoni: nel progetto Madwifi si è giunti alla realizzazione di un HAL libero (**OpenHAL**) da parte di Reyk Floeter e parallelamente nel tardo 2007 Atheros ha proceduto al rilascio del codice sorgente dell'HAL "originale" di Sam Leffler con licenza ISC; grazie al lavoro degli sviluppatori del team di OpenWRT (<http://openwrt.org>) Madwifi viene tutt'ora sviluppato. Si è avuta anche la creazione di due ulteriori drivers, completamente liberi sin dall'origine (e inclusi nel kernel tree) e destinati nel lungo periodo a sostituire completamente Madwifi, avendone le stesse possibilità di utilizzo. Si tratta dei driver "ath5k" e "ath9k", utilizzati rispettivamente per i "vecchi" chipset della famiglia AR5xxx (che generalmente presentano almeno due delle modalità 802.11a/b/g) e per i più nuovi AR9xxx che forniscono il supporto anche alla futura modalità 802.11n (che promette un sensibile aumento di velocità, ma che, ricordiamo, non è ancora definito come uno standard). Addirittura, per ath9k, Atheros ha contribuito fornendo stralci di codice sorgente, scrivendone attivamente altre parti e assumendo nel proprio organico alcuni sviluppatori volontari del driver. Per ath5k ha comunque fornito le specifiche dei chipset, che sono molto preziose per realizzare questo genere di software. Attualmente ath5k supporta ufficialmente le modalità Managed, Ad-hoc e Monitor, e in via ancora sperimentale anche Master; le funzionalità non strettamente necessarie al collegamento sono ancora in via di sviluppo come anche le caratteristiche non-standard, come

Wireless on Linux

Una panoramica delle schede wireless piu' diffuse e dei moduli necessari per usarle

ad esempio *Super A/G* e *XR*; in *ath9k* invece sono presenti tutte le caratteristiche pianificate. Esiste anche un quarto driver, "ar9170", (derivato dal precedente "otus", oramai obsoleto e deprecato) libero e nato per supportare appunto i chipset AR9170 usati nei dispositivi USB con supporto 802.11n, che sarà incluso nel kernel a partire dalla versione 2.6.31 e completato nella 2.6.32, ma di cui non si hanno notizie precise circa la sua stabilità negli utilizzi quotidiani, data la bassa diffusione attuale di questi dispositivi. Bisogna in ogni caso precisare che per ar9170 è prevista la presenza di un *firmware* (un piccolo software che viene caricato all'interno della scheda all'avvio della stessa, e necessario alla sua inizializzazione) per consentirne l'uso; quello ufficiale è solo binario e dunque non libero, ma nelle ultime settimane è stato realizzato un firmware libero che di fatto rende libero l'intero driver. Tale firmware libero è reperibile sul sito <http://www.linuxwireless.org>. Tramite ar9170 si possono utilizzare le modalità Station, Ad-hoc e Monitor, mentre non è conosciuto lo stadio del supporto alla modalità Master. Parliamo ora di Intel: il colosso californiano ha prodotto dapprima il chipset **2100**, con supporto 802.11b. Successivamente è stata la volta dei chipset **PRO/Wireless 2200** e **2915**, entrambi con supporto 802.11b/g (il secondo anche 802.11a) per cui è stato creato il driver "ipw2200", libero e direttamente supportato da Intel. Entrambi però prevedono l'utilizzo di un firmware proprietario. Questi driver non consentono di usare le modalità Master e Monitor; successivamente si è vista concretizzarsi la possibilità di usare la modalità Master con i chipset ipw2200, ma solo con una versione pesantemente modificata del driver (non inclusa nel kernel tree) e con una ben precisa del firmware. Poi è arrivato il chipset **PRO/Wireless 3945** con supporto 802.11a/b/g. Esso era inizialmente guidato dal

driver "ipw3945", non incluso nel kernel e bisognoso di un demone esterno per il suo funzionamento, oltre che di un firmware (chiamato microcode) non libero; attualmente si usa il driver "iwl3945" che elimina la necessità del demone esterno, ma che richiede ugualmente la presenza di un firmware (che ora Intel chiama *ucode*, e che ancora una volta non è stato rilasciato sotto una licenza libera). L'ultima generazione di chipset Intel è quella che offre il supporto 802.11n e guidata dal driver "iwlagn": parliamo dei chipset **WiFi Link AGN 4965, 5150 e 6050**. Anche questo driver vede necessario l'uso di un firmware proprietario (il solito *ucode*). Tutti questi drivers sono inclusi nel kernel tree, eccetto, lo ribadiamo, il vecchio ipw3945 e la versione modificata di ipw2200. Completamente opposto l'atteggiamento di **Broadcom**: tale azienda infatti non solo inizialmente non ha fornito un driver che permettesse di utilizzare i suoi prodotti con GNU/Linux, ma non ha nemmeno mai fornito le specifiche di tali dispositivi, precludendo così la realizzazione di un driver libero, almeno in tempi brevi. Per sopperire a questa mancanza infatti, alcuni sviluppatori volontari hanno realizzato, tramite *reverse engineering*, il driver (rilasciato con licenza GPL) "bcm43xx", successivamente evolutosi nei drivers "b43legacy", utilizzato per i primissimi dispositivi 802.11b, e "b43", che viene usato per quasi tutti i dispositivi più recenti fino a quelli della generazione 802.11a/b/g compresi, su bus PCI. Per la generazione 802.11n, infatti, tali sviluppatori non hanno ancora prodotto codice, causa la mancanza di specifiche, che tuttavia sono state da pochissimo completate dal team di *reverse engineering*. Nel frattempo però, per tale hardware è stato rilasciato da Broadcom un driver ufficiale, semi-proprietario (e quindi non incluso nel kernel tree), di prestazioni discutibili e capace

solo delle modalità Managed ed Ad-Hoc. Esso per giunta appare un semplicistico adattamento dei drivers realizzati da Broadcom per un diffuso sistema operativo proprietario. Per il funzionamento di b43 e b43legacy è necessario un firmware, che ufficialmente non è stato rilasciato, e che quindi è stato estratto (tramite un'utilità che gli stessi sviluppatori hanno realizzato: *b43-fwcutter*) prima dai driver ufficiali per altri sistemi operativi e successivamente dai drivers ufficiali per Linux di cui pocanzi. In ogni caso, ultimamente è stato realizzato da alcuni sviluppatori italiani un firmware libero, sebbene ancora sperimentale e limitato al funzionamento solo con alcuni chipset (<http://www.ing.unibs.it/openfw/wf/index.php>). I driver b43 e b43legacy hanno il supporto a parecchie funzionalità, tra cui le modalità Station, Monitor, Master, Ad-hoc e *Monitor while operating* (di fatto una congiunzione di monitor e station), il supporto *mesh*, *packet-injection*, crittografia hardware e *Bluetooth coexistence* (se l'adattatore bluetooth è montato sulla stessa scheda wireless); Tuttavia non sono utilizzabili le modalità 802.11a e 802.11n. Per i dispositivi USB si usa il driver "rndis_wlan"; in realtà, gli unici dispositivi USB ad oggi conosciuti e basati su prodotti broadcom sono quelli con chipset bcm4310 (che in realtà usa il bus PCI, risultando perciò incompatibile con rndis_wlan; il supporto dovrebbe arrivare un giorno da b43, ma purtroppo ancora questo hardware non è supportato) e bcm4320 (che appare l'unico ora compatibile con rndis_wlan). Problema conosciuto di tale driver è l'impossibilità di connessione con reti a SSID nascosto con crittografia WPA/WPA2.

Nella prossima puntata esamineremo lo stato dei drivers per i chipset meno diffusi.