

Dalla porta seriale a «Internet mobile»

«

35.1	Porte seriali	1536
35.1.1	Configurazione con il kernel Linux	1536
35.1.2	Connettori	1538
35.1.3	Controllo del flusso o handshaking	1538
35.1.4	Cavi RS-232C	1539
35.2	Modem	1540
35.2.1	Insieme esteso di comandi Hayes	1540
35.2.2	Indicatori luminosi dei modem esterni	1545
35.2.3	Codici di risposta	1545
35.2.4	Sequenze di escape	1547
35.3	File di dispositivo e collegamenti	1547
35.3.1	Gestione oculata dei permessi	1548
35.4	Programmi di comunicazione	1548
35.4.1	Accesso brutale al modem	1548
35.4.2	Utilizzo sommario di Minicom	1548
35.4.3	Utilizzo sommario di Seyon	1549
35.5	Configurazione del modem	1550
35.5.1	Profilo di configurazione del modem	1550
35.6	Rapidità di modulazione e velocità di trasmissione	1551
35.7	Impostazione della velocità	1552
35.8	Introduzione al PPP	1553
35.8.1	Funzionalità del kernel Linux	1553
35.9	Funzionamento generale del demone per il PPP	1554
35.9.1	Struttura del sistema di configurazione	1554
35.9.2	Struttura del sistema di autenticazione	1554
35.9.3	Interfacce PPP e funzioni privilegiate	1555
35.9.4	Indirizzi IP	1556
35.9.5	Script di contorno	1556
35.10	Avvio e opzioni	1556
35.10.1	Opzioni principali	1558
35.11	File per il sistema di autenticazione	1563
35.11.1	Configurazione PAP	1564
35.11.2	Configurazione CHAP	1565
35.12	Script	1566
35.12.1	Verifica dell'ambiente	1567
35.12.2	Gestione dinamica degli indirizzi DNS	1567
35.13	Impostazione della distribuzione GNU/Linux Debian	1568
35.14	Connessioni su porte seriali	1568
35.14.1	Programma di comunicazione	1568
35.15	Connessione PPP senza autenticazione	1570
35.15.1	Script di connessione	1570
35.15.2	Verifica della connessione	1571
35.15.3	Varianti	1572
35.16	Linea dedicata	1572
35.16.1	Ruolo dei modem	1573
35.16.2	Simulazione con l'aiuto di Minicom	1573
35.16.3	Connessione con pppd	1573
35.17	Autenticazione con il protocollo PPP	1574
35.17.1	Autenticazione tradizionale	1574

35.17.2	Autenticazione attraverso il PPP	1576
35.18	Cliente PPP che utilizza un sistema di identificazione tradizionale	1576
35.18.1	Chat	1576
35.18.2	Script di chat	1577
35.18.3	Demone per il PPP e Chat assieme	1579
35.19	Cliente PPP che fornisce esclusivamente un'identificazione PAP o CHAP	1580
35.20	WvDial	1581
35.20.1	Configurazione automatica di WvDial	1581
35.20.2	Configurazione automatica e trasparente di pppd	1582
35.20.3	Configurazione manuale	1583
35.20.4	Avvio e funzionamento	1584
35.21	Connessione mobile con «chiavetta»	1585
35.21.1	Problematiche con i sistemi GNU/Linux	1585
35.21.2	Modem USB	1586
35.21.3	WvDial e le «chiavette»	1586
35.22	Riferimenti	1587

35.1 Porte seriali

«

In un elaboratore x86 degli anni 1990 erano disponibili generalmente due porte seriali, prevedendo la possibilità di averne fino a quattro, denominate 'COM1:', 'COM2:',... La tabella 35.1 mostra la corrispondenza tra indirizzi e nomi dei file di dispositivo di un sistema GNU/Linux.

Tabella 35.1. Indirizzi delle porte seriali.

Porta su x86	IRQ	I/O	dispositivo
'COM1:'	4	3F8 ₁₆	'/dev/ttyS0'
'COM2:'	3	2F8 ₁₆	'/dev/ttyS1'
'COM3:'	4	3E8 ₁₆	'/dev/ttyS2'
'COM4:'	3	2E8 ₁₆	'/dev/ttyS3'

Nelle prime versioni del kernel Linux si distingueva tra dispositivi per le chiamate in uscita e dispositivi per le chiamate in ingresso: per le prime si utilizzavano i nomi '/dev/cua*' che sono ormai superati, ma attualmente, i dispositivi '/dev/ttyS*' svolgono entrambi i compiti.

Dal momento che la prima e la terza porta seriale, così come la seconda e la quarta, condividono lo stesso IRQ, per evitare conflitti era ed è meglio limitarsi all'utilizzo delle sole prime due porte seriali. Tuttavia, il kernel Linux potrebbe gestire delle schede seriali multiple speciali, in cui, con un solo IRQ si hanno a disposizione fino a un massimo di 32 porte seriali.

35.1.1 Configurazione con il kernel Linux

«

In presenza di porte seriali configurate in modo non standard, è indispensabile configurare il kernel Linux in modo da poterle gestire correttamente. A questo proposito, i sistemi GNU/Linux offrono Setserial,¹ un programma di servizio specifico per configurare le porte seriali in base alle loro caratteristiche reali:

```
setserial [opzioni] dispositivo [parametro [argomento]]...
```

```
setserial -g [-a] [-b] dispositivo...
```

Il programma 'setserial' permette di definire o verificare le informazioni sulla configurazione di una porta seriale particolare nell'ambito dei kernel Linux. Principalmente, si tratta dell'indicazione

dell'indirizzo di I/O e del numero di IRQ in cui il kernel si deve aspettare di trovare la porta seriale in questione.

In pratica, l'uso di 'setserial' è necessario quando si utilizzano porte seriali configurate in modo non standard, allo scopo di ottenerne l'identificazione e gestione corretta, secondo la loro configurazione particolare. Quando esiste questa esigenza, dal momento che il kernel dovrebbe essere configurato in tal modo a ogni avvio, è generalmente opportuno programmare l'utilizzo di 'setserial' all'interno della procedura di inizializzazione del sistema.

Per fare riferimento alla porta seriale da verificare o di cui si deve definire la configurazione, si utilizza il nome del file di dispositivo corrispondente, '/dev/ttyS*', subito dopo le opzioni eventuali.

Dopo il nome del dispositivo seriale, vengono indicati i «parametri», che a loro volta sono seguiti da un argomento eventuale. Se 'setserial' viene utilizzato senza parametri, oppure con l'opzione '-g', si ottiene semplicemente lo stato attuale della configurazione della porta seriale corrispondente.

Segue la descrizione di alcune opzioni della riga di comando.

Opzione	Descrizione
-g	Mostra le informazioni sui dispositivi seriali indicati come argomenti.
-a	Quando 'setserial' viene utilizzato per informare sullo stato della configurazione, con questa opzione si ottengono tutte le informazioni disponibili.
-b	Quando 'setserial' viene utilizzato per informare sullo stato della configurazione, con questa opzione si ottiene solo un riassunto delle informazioni disponibili.

Segue la descrizione di alcuni parametri da indicare nella riga di comando.

Parametro	Descrizione
port indirizzo_i/o	Permette di definire l'indirizzo di I/O della porta seriale.
irq indirizzo_irq	Permette di definire l'indirizzo IRQ della porta seriale.
uart {none↔ ↔ 8250 16450 16550↔ ↔ 16550A 16650↔ ↔ 16750↔ ↔ 16850 16950 16954}	Permette di definire in modo esplicito il tipo di UART utilizzato, salvo il caso di 'none' che disabilita la porta seriale. Può essere utile quando il sistema di autorilevamento non funziona per qualche ragione, oppure quando il tipo individuato non risulta veritiero. In generale, si distingue tra il tipo 16550A e gli altri; il primo ha una memoria FIFO che viene utilizzata, mentre per gli altri, anche se alcuni ne dispongono, non ne viene attivato l'utilizzo.
spd_hi	Fa in modo che venga utilizzata la velocità di 57600 bit/s (bps) quando l'applicazione ne richiede 38400.
spd_vhi	Fa in modo che venga utilizzata la velocità di 115200 bit/s quando l'applicazione ne richiede 38400.
spd_shi	Fa in modo che venga utilizzata la velocità di 230400 bit/s quando l'applicazione ne richiede 38400.
spd_warp	Fa in modo che venga utilizzata la velocità di 460800 bit/s quando l'applicazione ne richiede 38400.

Vengono descritti alcuni esempi.

```
• # setserial -g -a /dev/ttyS1 [Invio]
```

Visualizza tutte le informazioni disponibili sulla seconda porta seriale.

```
• # setserial /dev/ttyS2 port 0x2e8 [Invio]
```

Imposta la configurazione della terza porta seriale corrispondente al file di dispositivo '/dev/ttyS2', definendo che per questa viene utilizzato l'indirizzo di I/O 2E8₁₆.

```
• # setserial /dev/ttyS2 irq 5 [Invio]
```

Imposta la configurazione della terza porta seriale, definendo che per questa viene utilizzato il livello di IRQ 5.

```
• # setserial /dev/ttyS2 port 0x3e8 irq 5 spd_hi ↵
  ↵      ↵      ↵      ↵      ↵      ↵
    uart 16550 [Invio]
```

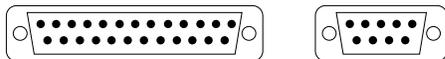
Imposta la configurazione della terza porta seriale, definendo che per questa viene utilizzato l'indirizzo di I/O 3E8₁₆ e l'IRQ numero 5. Inoltre si stabilisce che si tratta di un UART 16550 (senza FIFO, o non funzionante) e si fa in modo di utilizzare una velocità elevata (57600 bit/s) quando l'applicazione richiede 38400 bit/s.

Quando si ha la necessità di configurare una o più porte seriali attraverso 'setserial', è opportuno che questa operazione venga svolta ogni volta che si accende l'elaboratore, attraverso la procedura di inizializzazione del sistema. Generalmente si tratta di modificare o creare il file '/etc/init.d/setserial', o un altro file simile, in relazione all'organizzazione della propria distribuzione GNU/Linux.

35.1.2 Connettori

Il connettore di un porta seriale presente su un vecchio elaboratore x86 può essere di due tipi: maschio DB-25 o maschio DB-9. La porta seriale RS-232C originale utilizza il connettore DB-25, ma dal momento che in pratica si utilizzano solo nove dei 25 contatti, sugli elaboratori x86 sono apparse delle semplificazioni a nove contatti.

Figura 35.4. Connettori DB-25 e DB-9. Il terminale numero uno si trova a un'estremità della fila superiore di contatti.



La tabella seguente elenca i segnali associati ai contatti delle porte seriali:

Segnale	DB-25	DB-9
TD <i>Transmit data</i>	2	3
RD <i>Receive data</i>	3	2
RTS <i>Request to send</i>	4	7
CTS <i>Clear to send</i>	5	8
DSR <i>Data set ready</i>	6	6
Massa dei segnali	7	5
DCD <i>Data carrier detect</i>	8	1
DTR <i>Data terminal ready</i>	20	4
RI <i>Ring indicator</i>	22	9

Figura 35.6. La parte posteriore di un modem esterno tipico. Si può osservare il connettore seriale sulla parte sinistra.



35.1.3 Controllo del flusso o handshaking

Il controllo del flusso dei dati, tra la porta seriale e l'unità periferica a essa connessa, può essere di due tipi:

- hardware o RTS/CTS;
- software o XON/XOFF.

Il controllo di flusso hardware prevede l'utilizzo dei segnali RTS e CTS per la sincronizzazione tra la porta seriale e la periferica. Si tratta anche del metodo che garantisce la maggiore velocità. Il controllo di flusso software ignora i segnali hardware e utilizza invece i codici XON e XOFF.

35.1.4 Cavi RS-232C

Si tratta dei cavi utilizzati per connettere un'unità periferica a una porta seriale. A seconda dei componenti da connettere tra loro, si parla di DTE (*Data terminal equipment*) e DCE (*Data communications equipment*). L'elaboratore è sempre un DTE, il modem è un'unità DCE, mentre una stampante o un terminale può essere un DTE.

Quando si connettono due unità eterogenee, come un elaboratore con un modem, si utilizza un cavo seriale composto da un connettore DB-25 maschio, da collegare all'unità periferica DCE, e da un connettore DB-25 o DB-9 femmina, da collegare alla porta seriale dell'elaboratore (DTE). Con questo tipo di cavo, tutti i segnali di un capo sono connessi con gli stessi segnali dell'altro.

Tabella 35.7. Cavo seriale RS-232C standard (DTE-DCE)

Segnale	DTE (elaboratore)	DTE (elaboratore)	DCE (modem)
	DB-25	DB-9	DB-25
TD <i>Transmit data</i>	2	3	2
RD <i>Receive data</i>	3	2	3
RTS <i>Request to send</i>	4	7	4
CTS <i>Clear to send</i>	5	8	5
DSR <i>Data set ready</i>	6	6	6
Massa dei segnali	7	5	7
DCD <i>Data carrier detect</i>	8	1	8
DTR <i>Data terminal ready</i>	20	4	20
RI <i>Ring indicator</i>	22	9	22

Un cavo Null-modem, per la connessione tra due elaboratori (o comunque due unità DTE) attraverso la porta seriale, può essere realizzato utilizzando due connettori DB-25 femmina, oppure DB-9 femmina, oppure un DB-25 e un DB-9 femmina. Se si intende utilizzare un controllo di flusso software, ovvero XON/XOFF, sono sufficienti tre fili, mentre per un controllo di flusso hardware, ovvero RTS/CTS, sono necessari sette fili. Se il cavo ha una schermatura metallica, questa può essere connessa alla parte metallica di uno solo dei due connettori.

Tabella 35.8. Cavo seriale a tre fili, per collegamenti tra DTE e DTE.

DB-25 femmina	DB-25 femmina	DB-25 femmina	DB-9 femmina	DB-9 femmina	DB-9 femmina
2	3	2	2	2	3
3	2	3	3	3	2
7	7	7	5	5	5

Tabella 35.9. Cavo seriale a sette fili, per collegamenti tra DTE e DTE.

DB-25 femmina	DB-25 femmina	DB-25 femmina	DB-9 femmina	DB-9 femmina	DB-9 femmina
2	3	2	2	3	2
3	2	3	3	2	3
4	5	4	8	7	8
5	4	5	7	8	7
6+8	20	6+8	4	6+1	4
20	6+8	20	6+1	4	6+1
7	7	7	5	5	5

35.2 Modem

Il modem è l'apparecchio che consente di trasformare un flusso di dati seriale in un segnale analogico modulato in modo da contenere tali informazioni, e viceversa. Il nome rappresenta esattamente la fusione delle parole «modulatore» e «demodulatore». La tecnologia del modem riguarda quindi l'utilizzo di linee analogiche per la trasmissione di dati in forma digitale.

La tabella 35.10 elenca alcune sigle utilizzate per identificare le caratteristiche dei modem, in particolare quelle dell'ITU (*International telecommunications union*).

Tabella 35.10. Standard sulle caratteristiche dei modem.

Standard	Caratteristiche
V.21 (Bell 103)	300 bit/s
V.22 (Bell 212A)	1 200 bit/s
V.23	trasmissione/ricezione 1 200 / 75 bit/s
V.22 bis	2 400 bit/s
V.27	fax
V.29	fax
V.32	4 800 bit/s, 9 600 bit/s
V.32 bis	4 800 bit/s, 7 200 bit/s, 9 600 bit/s, 12 000 bit/s, 14 400 bit/s
V.34	28 800 bit/s
V.34+	33 600 bit/s
V.42	correzione errori (include LAP-M)
V.42 bis	compressione dati
MNP4	correzione errori
MNP5	compressione dati
V.90	trasmissione/ricezione 31 200 / 56 000 bit/s

Quando si utilizza il modem si distinguono due situazioni: la modalità di comando e la modalità dati. Quando si accende il modem, questo si trova nella modalità di comando, con la quale accetta una serie di comandi dall'elaboratore o dall'unità a cui è collegato, rispondendo di conseguenza. Quando si stabilisce una connessione, si passa alla modalità dati e il modem non accetta più comandi (tranne uno speciale), perché tutto il traffico viene considerato parte della comunicazione.

35.2.1 Insieme esteso di comandi Hayes

I comandi dei modem compatibili Hayes iniziano quasi sempre per «AT» seguito da una serie eventuale di codici di comando alfanumerici e quindi da un codice di ritorno a carrello (<CR>).

```
AT[comando...]
```

Per esempio:

- ATDP chiamata a impulsi (telefono decadico);
- ATDT chiamata a toni (telefono multifrequenza).

I comandi di base iniziano con una lettera alfabetica; a questi sono stati aggiunti nel tempo dei comandi estesi che possono iniziare con una e-commerciale ('&'), un simbolo di percentuale ('%'), una barra obliqua inversa ('\') e altri simboli ancora. Quando si fa riferimento a comandi estesi, è difficile stabilire quale sia lo standard; qui si vogliono elencare solo i comandi di base e quelli estesi più comuni e quindi più importanti.

Alcuni comandi speciali non fanno uso del solito prefisso di comando AT. Sono pochi e piuttosto importanti.

Tabella 35.11. Comandi senza il prefisso AT.

Comando	Descrizione
A/	Ripete l'ultimo comando (si usa da solo, senza il prefisso AT e senza <CR> alla fine).

Comando	Descrizione
<i>pausa+++pausa</i>	Sequenza di escape, preceduta e seguita da una pausa di almeno un secondo. Si può usare quando il modem è nella modalità dati e lo si vuole riportare a quella di comando. Generalmente, dopo la pausa finale, viene inviato al modem un comando AT nullo: pausa+++pausaAT<CR> . Dopo aver riportato il modem alla modalità di comando, è possibile rimetterlo subito nella modalità dati attraverso il comando ATO.

I comandi seguenti richiedono il prefisso AT e sono seguiti dal carattere di ritorno a carrello (<CR>). I comandi prefissati da AT possono essere più o meno complessi e lunghi di conseguenza; questa lunghezza ha un limite che varia da modem a modem. In generale, quando possibile, è opportuno suddividere questi comandi se sono troppo lunghi.²

La maggior parte dei casi, i comandi AT sono formati da una sigla iniziale che definisce il tipo di comando e sono seguiti da un parametro numerico. Per esempio, ATH0 serve a chiudere la linea telefonica. Questi comandi possono essere composti senza il parametro finale (cioè senza il numero), quando si vuole fare riferimento allo zero. Quindi, ATH è esattamente uguale a ATH0.

I comandi AT possono contenere spazi, per facilitare la lettura umana. Resta comunque valido il problema del limite massimo alla loro lunghezza, che in tal modo deve tenere conto anche degli spazi aggiuntivi (ammesso che il modem non ne tenga conto esplicitamente).

Tabella 35.12. Comandi AT.

Comando	Descrizione
A	<i>Answer</i> . Risposta senza attendere il segnale di chiamata.
D ⁿ	<i>Dial pulse</i> . Compone il numero di telefono <i>n</i> a impulsi.
DT ⁿ	<i>Dial tone</i> . Compone il numero di telefono <i>n</i> a toni. Se all'interno delle cifre del numero telefonico viene utilizzata una virgola (','), questa rappresenta una pausa nella composizione. Solitamente, questa pausa dura due secondi. Il comando ATD è speciale: dopo il numero telefonico da comporre non è possibile accodare altri comandi.
E0	<i>Echo</i> . Disattiva l'eco dei comandi.
E1	Attiva l'eco dei comandi. È il valore predefinito.
F0	Funzionamento in <i>Half duplex</i> .
F1	Funzionamento in <i>Full duplex</i> .
H0	<i>Hang</i> . Il modem chiude la connessione alla linea telefonica.
H1	Il modem apre la connessione alla linea telefonica.
H2	Il telefono e il modem sono entrambi connessi alla linea telefonica.
L0	<i>Loudness</i> . Il livello sonoro dell'altoparlante interno al modem viene posizionato al livello minimo.
L1	Il livello sonoro dell'altoparlante interno al modem viene posizionato a un livello basso.
L2	Il livello sonoro dell'altoparlante interno al modem viene posizionato a un livello medio. È il valore predefinito.
L3	Il livello sonoro dell'altoparlante interno al modem viene posizionato a un livello alto.
M0	<i>Mode</i> . Altoparlante spento.
M1	Altoparlante acceso durante la chiamata e spento non appena riceve il segnale di portante. È il valore predefinito.

Comando	Descrizione
M2	Altoparlante sempre acceso.
M3	Altoparlante spento durante la composizione, quindi acceso, poi spento non appena riceve il segnale di portante.
O0	<i>On-line</i> . Quando per qualche motivo il modem è tornato alla modalità di comando mentre si trovava in quella dati, per esempio perché è stato generato un escape (+++), con il comando O0 si fa in modo che il modem torni alla modalità dati.
O1	Riporta il modem alla modalità dati, forzando però una procedura di equalizzazione, in modo da riadattarsi alle caratteristiche della linea.
Q0	<i>Quiet</i> . Vengono inviati i codici di risultato.
Q1	Non vengono inviati i codici di risultato.
S <i>n</i> = <i>x</i>	<i>S-register</i> . Attribuisce al registro <i>n</i> il valore <i>x</i> .
S <i>n</i> ?	Visualizza il valore del registro <i>n</i> .
V0	<i>Verbose</i> . Non vengono tradotti i codici di risultato.
V1	Vengono tradotti i codici di risultato in forma verbale. È il valore predefinito.
X0	<i>Extensive</i> . Seleziona i codici di risultato a livello base (300 bit/s).
X1	Esteso senza rilevamento del tono di chiamata (<i>dialtone</i>) o del segnale di occupato (<i>busy</i>).
X2	Esteso con rilevamento del tono di chiamata (<i>dialtone</i>), ma non del segnale di occupato (<i>busy</i>).
X3	Esteso con rilevamento del segnale di occupato (<i>busy</i>), ma non del tono di chiamata (<i>dialtone</i>). ATX3 è la scelta migliore quando si utilizzano le linee telefoniche italiane. Se si tentano altre modalità si ottiene solo il tipico messaggio di errore: 'NO DIALTONE' .
X4	Esteso con rilevamento del tono di chiamata (<i>dialtone</i>) e del segnale di occupato (<i>busy</i>).
Y0	Disabilita la disconnessione dopo uno <i>space</i> lungo (ovvero dopo un <i>break</i>). È il valore predefinito.
Y1	Abilita la disconnessione dopo uno <i>space</i> lungo (ovvero dopo un <i>break</i>).
Z	Preleva il profilo di configurazione dalla memoria non volatile. Se il modem è provvisto di diverse memorie per la registrazione dei profili di configurazione, si possono utilizzare i comandi ATZ0, ATZ1, ATZ2,... per prelevare il primo profilo, il secondo, il terzo,... In generale, ATZ e ATZ0 sono la stessa cosa.
&C0	<i>Carrier</i> . Il modem mantiene sempre alto il DCD (<i>Data carrier detect</i>).
&C1	Il livello del DCD segue l'andamento della portante rilevata dal modem.
&D0	Il modem ignora il DTR.
&D1	Il modem passa allo stato di comando quando il DTR passa dal livello alto al livello basso.
&D2	Quando il DTR passa dal livello alto al livello basso, il modem interrompe la comunicazione (aggancia) e disabilita la risposta automatica (ammesso che questa sia stata abilitata). Infine, torna alla modalità di comando.
&D3	Quando il DTR passa dal livello alto al livello basso, il modem si reinizializza.
&F	<i>Firmware</i> . Preleva il profilo di configurazione preimpostato dal fabbricante della ROM (praticamente una reinizializzazione del modem).
&L0	<i>Line</i> . Linea commutata.
&L1	Linea dedicata.
&S0	<i>Set</i> . Il modem mantiene sempre alto il DSR (<i>Data set ready</i>).
&S1	Il DSR funziona in base alle specifiche EIA.

Comando	Descrizione
&V	<i>View</i> . Consente di visualizzare il profilo memorizzato nella memoria non volatile. Se il modem è provvisto di diverse memorie per la registrazione dei profili di configurazione, si possono utilizzare i comandi AT&V0, AT&V1, AT&V2,... per visualizzare il primo profilo, il secondo, il terzo,... In generale, AT&V e AT&V0 sono la stessa cosa.
&W	<i>Write</i> . Scrive nella memoria non volatile il profilo attivo di configurazione. Se il modem è provvisto di diverse memorie per la registrazione dei profili di configurazione, si possono utilizzare i comandi AT&W0, AT&W1, AT&W2,... per registrare nel primo profilo, nel secondo, nel terzo,... In generale, AT&W e AT&W0 sono la stessa cosa.

Tabella 35.13. Sintesi dei comandi AT.

Comando	Descrizione
A	Risposta.
DP	Composizione a impulsi.
DT	Composizione a toni.
E	Eco dei comandi.
F	Duplex.
H	Aggancio.
L	Livello sonoro.
M	Altoparlante.
Q	Codici di risultato.
S <i>n</i> = <i>x</i>	Attribuzione del valore <i>x</i> al registro <i>n</i> .
S <i>n</i> ?	Interrogazione del contenuto del registro <i>n</i> .
V	Traduzione dei codici di risultato numerici.
X	Estensione.
Y	Disconnessione automatica.
Z	Prelievo del profilo di configurazione dalla memoria non volatile.
&F	Prelievo del profilo di configurazione dalla ROM.
&L	Linea dedicata o commutata.
&V	Visualizza il profilo di configurazione della memoria non volatile.
&W	Registra il profilo di configurazione nella memoria non volatile.

I registri sono delle caselle di memoria che permettono di ridefinire determinati valori riferiti al comportamento del modem. Per modificare un registro si utilizza il comando AT*S**n*=*x*, dove *n* è il numero del registro e *x* è il valore che gli si vuole assegnare.

Tabella 35.14. Registri «S» principali.

Registro	Descrizione
S0	Numero di squilli prima della risposta. Zero equivale a inibire la risposta automatica ed è il valore predefinito.
S1	Contatore degli squilli. Il modem utilizza questo registro come variabile per il conteggio degli squilli: quando il valore di questo registro raggiunge quello di S0, il modem risponde.
S2	Il codice di escape. Il valore predefinito corrisponde a 43, ovvero al simbolo '+'. Per passare dalla modalità <i>on line</i> a quella dei comandi, si preme per tre volte di seguito in rapida successione questo tasto: [+][+][+].
S3	Il codice utilizzato come <i>carriage return</i> . Il valore predefinito è 13, corrispondente a <CR>.

Registro	Descrizione
s4	Il codice utilizzato come <i>linefeed</i> . Il valore predefinito è 10, corrispondente a <LF>.
s5	Il codice utilizzato come <i>backspace</i> . Il valore predefinito è 8, corrispondente a <BS>.
s6	Tempo di attesa per il segnale di centrale espresso in secondi. Si tratta del tempo che il modem attende prima di iniziare a comporre il numero telefonico. Il valore predefinito è due.
s7	Tempo di attesa per la portante espresso in secondi. Si tratta del tempo entro il quale il modem si aspetta di ricevere la portante. Se ciò non avviene, il modem restituisce il messaggio di errore 'NO CARRIER'. Solitamente, il valore predefinito è 30.
s8	Durata della pausa espressa in secondi. Quando all'interno del numero telefonico da comporre appare una virgola, questa viene interpretata come pausa di composizione. La durata predefinita della pausa è di due secondi.
s9	Tempo per il rilevamento della portante espresso in decimi di secondo. La quantità di tempo necessario, durante il quale la portante deve essere presente per poter essere rilevata dal modem. Il valore predefinito è sei, corrispondente a 0,6 secondi.
s10	Tempo massimo di perdita della portante espresso in decimi di secondo. La durata massima della perdita della portante. Se la portante viene a mancare per un tempo maggiore, il modem riaggancia, ovvero chiude la comunicazione. Solitamente il valore predefinito è sette, corrispondente a 0,7 secondi. In presenza di linee disturbate, può essere necessario aumentare questo valore.
s11	Intervallo di tono espresso in millisecondi. Quando si utilizza la composizione a toni o DTMF, i toni che rappresentano le cifre numeriche devono essere spazati l'uno dall'altro da una breve pausa. Questo registro esprime il valore della pausa. Il valore predefinito si aggira tra i 50 ms e i 100 ms (millisecondi). La scelta della durata della pausa dipende dalle capacità della propria centrale: un valore di 50 è considerato il minimo possibile in assoluto.
s12	Tempo morto della sequenza di escape espresso in cinquantiesimi di secondo. È il tempo che deve trascorrere prima e dopo una sequenza di escape (+++). Il valore predefinito è 50, corrispondente a un secondo.

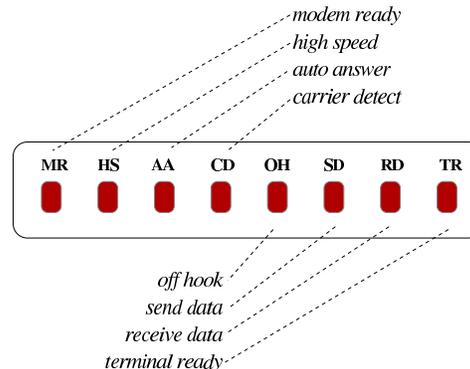
Tabella 35.15. Sintesi dei registri «S» principali.

Registro	Descrizione
s0	Numero di squilli prima della risposta automatica.
s1	Contatore degli squilli.
s2	Codice di escape.
s3	Codice di ritorno a carrello.
s4	Codice per l'avanzamento di riga.
s5	Codice per il <i>backspace</i> .
s6	Secondi di attesa per il segnale di centrale.
s7	Secondi di attesa per la portante.
s8	Secondi di durata della pausa (virgola).
s9	Decimi di secondo per il rilevamento della portante.
s10	Decimi di secondo consentiti per la perdita della portante.
s11	Millisecondi di spaziatura tra i toni di composizione.
s12	Cinquantiesimi di secondo per i tempi morti delle sequenze di escape.

35.2.2 Indicatori luminosi dei modem esterni

I modem esterni tradizionali (quelli usati per le linee telefoniche analogiche) hanno degli indicatori luminosi, più o meno standard, che danno un'indicazione istantanea sullo stato di questo. Queste indicazioni sono abbastanza importanti, ed è utile conoscerne il significato.

Figura 35.16. Indicatori luminosi dei modem esterni.



Segue la descrizione del significato di questi indicatori luminosi:

Sigla	Descrizione
MR, <i>Modem ready</i>	quando l'indicatore MR è acceso, il modem è alimentato elettricamente;
HS, <i>High speed</i>	quando l'indicatore HS è acceso, la comunicazione tra DTE e modem avviene a una velocità «elevata» (può trattarsi di un valore che supera i 2400 bit/s);
AA, <i>Auto answer</i>	quando l'indicatore AA è acceso, il modem è configurato per rispondere alle chiamate, oppure ha ricevuto uno o più squilli del telefono;
CD, <i>Carrier detect</i>	quando l'indicatore CD è acceso, il modem sta ricevendo, dal modem remoto, un segnale di portante valido;
OH, <i>Off hook</i>	quando l'indicatore OH è acceso, il modem sta utilizzando la linea telefonica;
SD, <i>Send data</i>	quando l'indicatore SD è acceso, il modem sta trasmettendo dati (ovvero sta ricevendo dati dall'elaboratore, o da altra unità, da trasmettere nella linea);
RD, <i>Receive data</i>	quando l'indicatore RD è acceso, il modem sta ricevendo dati (ovvero sta inviando i dati ricevuti dalla linea, verso l'elaboratore o altra unità);
TR, <i>Terminal ready</i>	di solito viene utilizzato per visualizzare la condizione del segnale DTR (<i>Data terminal ready</i>).

35.2.3 Codici di risposta

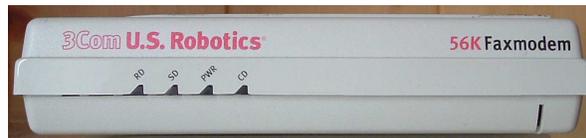
Quando il modem è configurato in modo da restituire i codici di risposta, questi vengono emessi in forma verbale o numerica: ATQ0 abilita l'emissione delle risposte, ATV1 visualizza i messaggi in inglese invece che in forma numerica.

Tabella 35.18. Codici di risposta standard dei modem.

codice numerico	codice verbale	descrizione
0	OK	Comando eseguito senza errori
1	CONNECT	Connessione stabilita (a 300 bit/s)
2	RING	Il telefono sta suonando
3	NO CARRIER	Perdita della portante o mancato rilevamento
4	ERROR	Errore nel comando o riga troppo lunga
5	CONNECT 1200	Connessione stabilita a 1200 bit/s
6	NO DIALTONE	Assenza del tono di chiamata

codice numerico	codice verbale	descrizione		
7	BUSY	Rilevamento del segnale di occupato		
8	NO ANSWER			
9/10	CONNECT 2400	Connessione 2400 bit/s	stabilita	a
13	CONNECT 9600	Connessione 9600 bit/s	stabilita	a
18	CONNECT 4800	Connessione 4800 bit/s	stabilita	a
20	CONNECT 7200	Connessione 7200 bit/s	stabilita	a
21	CONNECT 12000	Connessione 12000 bit/s	stabilita	a
25	CONNECT 14400	Connessione 14400 bit/s	stabilita	a
43	CONNECT 16800	Connessione 16800 bit/s	stabilita	a
85	CONNECT 19200	Connessione 19200 bit/s	stabilita	a
91	CONNECT 21600	Connessione 21600 bit/s	stabilita	a
99	CONNECT 24000	Connessione 24000 bit/s	stabilita	a
103	CONNECT 26400	Connessione 26400 bit/s	stabilita	a
107	CONNECT 28800	Connessione 28800 bit/s	stabilita	a
151	CONNECT 31200	Connessione 31200 bit/s	stabilita	a
155	CONNECT 33600	Connessione 33600 bit/s	stabilita	a
180	CONNECT 33333	Connessione 33333 bit/s	stabilita	a
184	CONNECT 37333	Connessione 37333 bit/s	stabilita	a
188	CONNECT 41333	Connessione 41333 bit/s	stabilita	a
192	CONNECT 42666	Connessione 42666 bit/s	stabilita	a
196	CONNECT 44000	Connessione 44000 bit/s	stabilita	a
200	CONNECT 45333	Connessione 45333 bit/s	stabilita	a
204	CONNECT 46666	Connessione 46666 bit/s	stabilita	a
208	CONNECT 48000	Connessione 48000 bit/s	stabilita	a
212	CONNECT 49333	Connessione 49333 bit/s	stabilita	a
216	CONNECT 50666	Connessione 50666 bit/s	stabilita	a
220	CONNECT 52000	Connessione 52000 bit/s	stabilita	a
224	CONNECT 53333	Connessione 53333 bit/s	stabilita	a
228	CONNECT 54666	Connessione 54666 bit/s	stabilita	a
232	CONNECT 56000	Connessione 56000 bit/s	stabilita	a
256	CONNECT 28000	Connessione 28000 bit/s	stabilita	a
260	CONNECT 29333	Connessione 29333 bit/s	stabilita	a
264	CONNECT 30666	Connessione 30666 bit/s	stabilita	a
268	CONNECT 32000	Connessione 32000 bit/s	stabilita	a
272	CONNECT 34666	Connessione 34666 bit/s	stabilita	a
276	CONNECT 36000	Connessione 36000 bit/s	stabilita	a
280	CONNECT 38666	Connessione 38666 bit/s	stabilita	a
284	CONNECT 40000	Connessione 40000 bit/s	stabilita	a

Figura 35.19. La parte anteriore di un modem esterno tipico. In questo caso sono visibili solo alcuni degli indicatori tradizionali.



35.2.4 Sequenze di escape

Quando si utilizza un programma per interagire con un modem e si devono indicare dei comandi AT di qualche tipo, capita la necessità di indicare dei simboli speciali, come il ritorno a carrello, o delle pause nel flusso di questi. Spesso sono validi i codici di escape che si vedono nella tabella 35.20.

Tabella 35.20. Codici di escape tipici per i programmi che interagiscono con il modem.

Codice	Significato
\d	Pausa di un secondo.
\p	Pausa di 0,1 s.
\n	<LF> (line feed).
\r	<CR> (carriage return).
\N	<NUL>.
\s	<SP> (spazio normale).
\t	<HT> (tabulazione).
\\	Una barra obliqua inversa singola.

35.3 File di dispositivo e collegamenti

I file di dispositivo relativi alle porte seriali di un sistema GNU/Linux hanno un nome del tipo `‘/dev/ttyS*’`. Dal momento che, almeno in teoria, è possibile gestire un massimo di 32 porte, i numeri utilizzati vanno da 0 a 31 (`‘/dev/ttyS0’`, `‘/dev/ttyS1’`, ..., `‘/dev/ttyS31’`).

Quando si utilizzano programmi che accedono alle porte seriali, occorre prendersi cura dei permessi associati a questi file di dispositivo, altrimenti sono utilizzabili solo dall'utente `‘root’`.

```
$ ls -l /dev/ttyS[0-3] [Invio]
crw-r--r-- 4 root root 4, 64 dic 16 17:30 /dev/ttyS0
crw-r--r-- 4 root root 4, 65 dic 16 17:37 /dev/ttyS1
crw-r--r-- 4 root root 4, 66 mag 5 1998 /dev/ttyS2
crw-r--r-- 4 root root 4, 67 mag 5 1998 /dev/ttyS3
```

Per esempio, se si vuole rendere disponibile l'utilizzo da parte di tutti gli utenti del modem connesso alla seconda porta seriale, occorre agire come segue:

```
# chmod a+rw /dev/ttyS1 [Invio]
$ ls -l /dev/ttyS[0-3] [Invio]
crw-r--r-- 4 root root 4, 64 dic 16 17:30 /dev/ttyS0
crw-rw-rw- 4 root root 4, 65 dic 16 17:37 /dev/ttyS1
crw-r--r-- 4 root root 4, 66 mag 5 1998 /dev/ttyS2
crw-r--r-- 4 root root 4, 67 mag 5 1998 /dev/ttyS3
```

Quando si ha a disposizione un modem soltanto, potrebbe essere opportuno predisporre un collegamento simbolico corrispondente a `‘/dev/modem’`, che punti al file di dispositivo corrispondente alla porta seriale a cui è connesso effettivamente il modem stesso. Così facendo, se i programmi che lo utilizzano fanno riferimento a questo collegamento, non occorre più cambiare la loro configurazione quando si sposta il modem: basta cambiare il collegamento.

```
lrwxrwxrwx 1 root root 65 dic 16 17:37 /dev/modem -> ttyS1
```

Ci sono pro e contro sull'utilità di questo collegamento. L'argomento più importante da tenere in considerazione contro la presenza di questo collegamento è il fatto che i programmi che lo utilizzano potrebbero creare dei file lucchetto (*lock file*) che segnalano il suo utilizzo, mentre può sembrare che il dispositivo che viene utilizzato effettivamente sia libero.

Per comodità, negli esempi che appaiono in questo e anche in altri capitoli, si utilizza la convenzione del collegamento `"/dev/modem"`, ma ciò non deve essere inteso come un invito a seguire questa strada in modo generalizzato.

35.3.1 Gestione oculata dei permessi

La gestione dei permessi per l'accesso al dispositivo della porta seriale cui è connesso il modem, può essere fatta in modo più proficuo assegnando a questi l'appartenenza a un gruppo diverso da `root`, per esempio `dialout`, abbinando poi questo gruppo agli utenti cui si vuole concedere l'accesso.

Supponendo di voler utilizzare il gruppo `dialout`, si potrebbe modificare il file `"/etc/group"` in modo che al gruppo `dialout` facciano parte anche gli utenti che devono accedere alle porte seriali in uscita. Per esempio, la riga seguente rappresenta il record del file `"/etc/group"` in cui si dichiara il gruppo `dialout`.

```
dialout::14:dialout,root,daniele,tizio,caio
```

Qui, oltre all'utente fittizio `dialout` (ammesso che esista) e all'amministratore `root`, viene concesso agli utenti `daniele`, `tizio` e `caio` di partecipare a questo gruppo.

35.4 Programmi di comunicazione

Un programma di emulazione di terminale è l'ideale per verificare il funzionamento del modem e soprattutto per poter memorizzare il profilo di configurazione preferito in modo che il comando ATZ lo imposti istantaneamente secondo la proprie necessità. Oltre a tali esigenze, attraverso questo tipo di programma si può effettuare una connessione fittizia al proprio fornitore di accesso a Internet in modo da conoscere precisamente la procedura di connessione e da poter realizzare uno script adeguato.

35.4.1 Accesso brutale al modem

Anche senza un programma di emulazione di terminale si può accedere al modem, utilizzando gli strumenti elementari offerti dal sistema operativo. È sufficiente il programma `cat` utilizzato nel modo seguente (si suppone che il collegamento `"/dev/modem"` corrisponda al dispositivo seriale abbinato al modem).

```
# cat < /dev/modem & [Invio]
# cat > /dev/modem [Invio]
```

Con questi due comandi, si ottiene di emettere quanto generato dal modem attraverso lo standard output e di dirigere lo standard input (ottenuto dalla tastiera) verso il modem.

```
AT [Invio]
```

```
AT
```

```
OK
```

In questo modo si può fare (quasi) tutto quello che si potrebbe con un programma di emulazione di terminale. Si può anche simulare la connessione con un ISP, ma forse qualche messaggio potrebbe non essere visualizzato nel momento giusto.

35.4.2 Utilizzo sommario di Minicom

Prima di poter utilizzare Minicom³ occorre che sia stato predisposto il file `"/etc/minirc.dfl"` attraverso la procedura di configurazione cui si accede attraverso Minicom quando viene avviato con l'opzione `-s`. Per gli scopi degli esempi riportati in queste sezioni, è

sufficiente salvare la configurazione predefinita, in pratica basta che il file `"/etc/minirc.dfl"` esista e sia vuoto.

Oltre al file di configurazione, occorre aggiungere all'interno del file `"/etc/minicom.users"` i nomi degli utenti abilitati al suo utilizzo.

Per avviare Minicom (l'eseguibile `minicom`) è sufficiente il nome senza argomenti.

```
$ minicom [Invio]
```

Segue un breve esempio nel quale in particolare si interroga il modem per conoscere il profilo di configurazione memorizzato nella memoria non volatile (AT&V).

```
Minicom 1.71 Copyright (c) Miquel van Smoorenburg

Press CTRL-A Z for help on special keys

AT S7=45 S0=0 L1 V1 X4 &c1 E1 Q0
OK

AT&V [Invio]

ACTIVE PROFILE:
B1 E1 L1 M1 Q0 V1 W0 X4 &B1 &C1 &D2 &G0 &L0 &P0 &Q0 &R0 &S0 &X0 &Y0
&A013 &C1 &G1 &A3 &C0 &G0 &J0 &K5 &N3 &Q3 &T000 &V0 &X0 -J1 *H3 *0032
S00:000 S01:000 S02:043 S03:013 S04:010 S05:008 S06:002 S07:045 S08:002
S09:006 S10:014 S11:095 S12:050 S18:000 S25:005 S26:001 S37:000 S72:000

STORED PROFILE 0:
B1 E1 L2 M1 Q0 V1 W0 X3 &B1 &C1 &D2 &G0 &L0 &P0 &Q0 &R0 &S0 &X0
&A013 &C1 &G1 &A3 &C0 &G0 &J0 &K5 &N3 &Q3 &T000 &V0 &X0 -J1 *H3 *0032
S00:000 S02:043 S03:013 S04:010 S05:008 S06:002 S07:060 S08:002
S09:006 S10:014 S11:095 S12:050 S18:000 S25:005 S26:001 S37:000 S72:000

TELEPHONE NUMBERS:
&Z0=
&Z1=
&Z2=
&Z3=

OK

[Ctrl a][x]
```

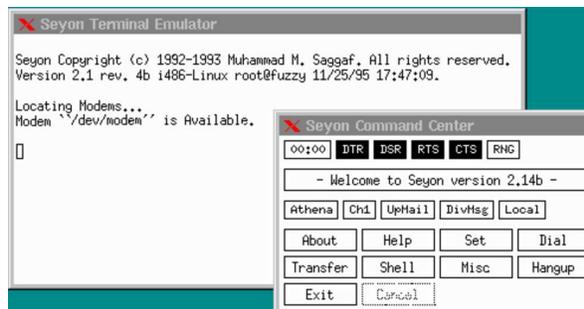
Nell'esempio, è stato trascurato il fatto che la configurazione predefinita non sia adatta alla situazione normale delle linee telefoniche italiane. Infatti, la stringa di inizializzazione inviata automaticamente da Minicom al modem contiene il comando ATX4 che in Italia non è appropriato.

35.4.3 Utilizzo sommario di Seyon

Seyon⁴ è un programma di emulazione di terminale che utilizza l'interfaccia grafica X. Se si utilizza il collegamento `"/dev/modem"` per riferirsi alla porta seriale alla quale è connesso il modem si può avviare l'eseguibile `seyon` nel modo seguente:

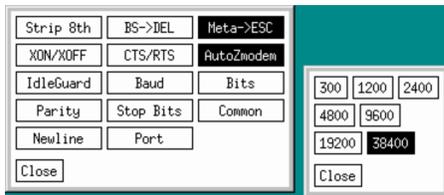
```
$ seyon -modems /dev/modem [Invio]
```

Figura 35.28. Avvio del programma di comunicazione Seyon.



La finestra *Seyon Command Center* permette di accedere alla configurazione dei parametri di comunicazione attraverso il pulsante `SET`.

Figura 35.29. Configurazione della velocità massima di comunicazione attraverso il pannello di comando di Seyon.



La figura 35.30 è un esempio di connessione attraverso comandi scritti direttamente senza l'aiuto del programma di comunicazione.

Figura 35.30. Esempio di connessione con Seyon.

```
Seyon Copyright (c) 1992-1993 Muhammad M. Saggaf, All rights reserved.
Version 2.1 rev. 4b i486-Linux root@fuzzy 11/25/95 17:47:09.

Locating Modems...
Modem "/dev/modem" is Available.

ATZ
OK
ATX3
OK
ATDT306371
CONNECT 9600

Welcome to Linux 1.2.3.

tvlllogin: daniela
Password:
Last login: Mon Mar 31 23:02:26 on ttyS17
Linux 1.2.3. (POSIX).
```

35.5 Configurazione del modem

« Nelle sezioni precedenti sono stati visti dei comandi e dei registri utili a definire il comportamento del modem. I programmi che utilizzano il modem, come quelli di comunicazione e i fax, hanno la necessità di predisporre il modem nel modo ottimale per ciò che da loro deve essere fatto.

I programmi più sofisticati guidano l'utente alla configurazione del modem senza la necessità di indicare esplicitamente alcun comando AT. Questi programmi trasformano poi la configurazione in una stringa di inizializzazione che viene inviata al modem prima di qualunque attività.

I programmi meno sofisticati prevedono la possibilità per l'utente di inserire una stringa di inizializzazione che vada a sommarsi alla configurazione già gestita dal programma.

Esiste tuttavia la possibilità di inserire una configurazione di massima già nel modem, come viene descritto nella prossima sezione.

35.5.1 Profilo di configurazione del modem

« I modem standard contengono una *configurazione di fabbrica* registrata su ROM e almeno un *profilo di configurazione* registrato in una memoria non volatile, modificabile da parte dell'utilizzatore.

La predisposizione di una buona configurazione in questa memoria non volatile, permette di utilizzare il comando ATZ per richiamare tutto ciò che in essa è stato definito, semplificando la configurazione attraverso i programmi che utilizzano il modem. La sequenza di operazioni seguente mostra il modo normale di predisporre una tale configurazione.

La prima cosa da fare è utilizzare un programma di comunicazione come Minicom per poter colloquiare con il modem.

```
$ minicom [Invio]
```

```
...
OK
```

Quasi tutti i programmi del genere, subito dopo l'avvio, inizializzano il modem in qualche modo. Prima di proseguire si carica il profilo

di configurazione memorizzato precedentemente nella memoria non volatile.

```
ATZ [Invio]
```

```
OK
```

Si procede quindi con dei comandi che servono a cambiare la modalità di funzionamento del modem. In questo caso si cambia il tipo di responso in modo che sia compatibile con il tipo di linee telefoniche utilizzate in Italia, quindi si modifica il registro S11 in modo che la pausa tra i toni di composizione sia di 100 ms.

```
ATX3 [Invio]
```

```
OK
```

```
ATS11=100 [Invio]
```

```
OK
```

Per verificare l'esito, basta utilizzare il comando AT&V.

```
AT&V [Invio]
```

```
ACTIVE PROFILE:
B1 E1 L2 M1 Q0 V1 W0 X3 &B1 &C1 &D2 &G0 &L0 &P0 &Q0 &R0 &S0 &X0
%A013 %C1 %G1 \A3 \C0 \G0 \J0 \K5 \N3 \Q3 \T000 \V0 \X0 -J1 *H3 *0032
S00:000 S02:043 S03:013 S04:010 S05:008 S06:002 S07:060 S08:002
S09:006 S10:014 S11:100 S12:050 S18:000 S25:005 S26:001 S37:000 S72:000

STORED PROFILE 0:
B1 E1 L2 M1 Q0 V1 W0 X4 &B1 &C1 &D2 &G0 &L0 &P0 &Q0 &R0 &S0 &X0
%A013 %C1 %G1 \A3 \C0 \G0 \J0 \K5 \N3 \Q3 \T000 \V0 \X0 -J1 *H3 *0032
S00:000 S02:043 S03:013 S04:010 S05:008 S06:002 S07:060 S08:002
S09:006 S10:014 S11:095 S12:050 S18:000 S25:005 S26:001 S37:000 S72:000

TELEPHONE NUMBERS:
&Z0=
&Z1=
&Z2=
&Z3=
```

```
OK
```

Si può osservare la differenza tra il profilo attivo (il primo) e quello contenuto nella memoria non volatile (il secondo). Evidentemente può trattarsi soltanto delle due cose che sono state modificate. Se si desidera modificare altro si continua, altrimenti si memorizza il nuovo profilo di configurazione.

```
AT&W [Invio]
```

```
OK
```

Se si utilizza nuovamente il comando AT&V si può verificare che il profilo attivo è stato copiato nella memoria non volatile.

```
AT&V [Invio]
```

```
ACTIVE PROFILE:
B1 E1 L2 M1 Q0 V1 W0 X3 &B1 &C1 &D2 &G0 &L0 &P0 &Q0 &R0 &S0 &X0
%A013 %C1 %G1 \A3 \C0 \G0 \J0 \K5 \N3 \Q3 \T000 \V0 \X0 -J1 *H3 *0032
S00:000 S02:043 S03:013 S04:010 S05:008 S06:002 S07:060 S08:002
S09:006 S10:014 S11:100 S12:050 S18:000 S25:005 S26:001 S37:000 S72:000

STORED PROFILE 0:
B1 E1 L2 M1 Q0 V1 W0 X3 &B1 &C1 &D2 &G0 &L0 &P0 &Q0 &R0 &S0 &X0
%A013 %C1 %G1 \A3 \C0 \G0 \J0 \K5 \N3 \Q3 \T000 \V0 \X0 -J1 *H3 *0032
S00:000 S02:043 S03:013 S04:010 S05:008 S06:002 S07:060 S08:002
S09:006 S10:014 S11:100 S12:050 S18:000 S25:005 S26:001 S37:000 S72:000

TELEPHONE NUMBERS:
&Z0=
&Z1=
&Z2=
&Z3=
```

```
OK
```

Al termine basta concludere il funzionamento del modem. In questo caso con la sequenza [Ctrl a][x].

35.6 Rapidità di modulazione e velocità di trasmissione

« Quando si utilizzano le porte seriali e i modem, è importante chiarire i concetti legati alla velocità di trasmissione. Per prima cosa è

bene distinguere due situazioni: la comunicazione attraverso porte seriali (per esempio quella che può avvenire tra la porta seriale di un elaboratore e la porta corrispondente di un modem) rispetto a quella tra due modem, tramite un doppino telefonico (una coppia di fili di rame isolati tra di loro). Nel primo caso (porte seriali), i dati sono trasmessi solo in forma di segnale elettrico, in base alla tensione che questo assume, cosa che implica anche una limitazione nella lunghezza del cavo. Nel secondo caso (doppino di rame) la distanza da raggiungere impone che le informazioni siano trasmesse attraverso una o più portanti di frequenza adatte al mezzo.

Quando si parla di velocità di trasmissione attraverso un cavo seriale, l'unica indicazione possibile si riferisce al numero di bit che possono transitare nell'intervallo di un secondo, cosa espressa dall'unità di misura *bit/s*, conosciuta volgarmente come *bps* (*Bit per second*).

Quando si pensa alla trasmissione attraverso una portante modulata, oltre al concetto di velocità espresso in bit per secondo, si può aggiungere un parametro aggiuntivo che rappresenta la rapidità di modulazione della portante. Si parla in questo caso di *baud*.

In origine, i tipi di modulazione utilizzati permettevano di trasmettere dati a una velocità massima pari allo stesso valore baud, contribuendo a confondere le due cose. Attualmente, i modem più recenti possono operare a un massimo di 2400 baud, mentre riescono a comunicare a una velocità in bit/s ben superiore (33600 bit/s sono diventati una cosa normale). Questo significa, evidentemente, che le tecniche di modulazione attuali permettono di trasmettere più bit per ogni baud.

In conclusione:

- quando si parla di velocità di trasmissione, si intende fare riferimento all'unità di misura bit/s (bps), mentre il termine baud è piuttosto un parametro legato alle caratteristiche del mezzo trasmissivo;
- un'affermazione in cui si utilizza l'unità di misura baud per esprimere una velocità di trasmissione è probabilmente scorretta, o impropria, soprattutto quando si fa riferimento a valori superiori a 2400;
- a volte, la tradizione impone l'utilizzo errato del termine baud, ma questo accade proprio quando i valori bit/s e baud coincidono, per esempio quando si parla di *autobauding*, concetto che riguarda prevalentemente modem molto vecchi che utilizzano velocità inferiori o uguali a 2400 bit/s.

35.7 Impostazione della velocità

La velocità di comunicazione della porta seriale deve essere scelta opportunamente, in funzione della velocità con cui il modem è in grado di ricevere e trasmettere dati. Generalmente, la velocità della porta deve essere quattro volte superiore a quella della comunicazione del modem, perché potrebbe intervenire l'effetto della compressione dati ad aumentare il volume effettivo di informazioni scambiate.

Il problema si pone particolarmente quando si utilizzano modem con velocità di trasmissione superiore a 9600 bit/s.

In pratica, quando si usano modem da 9600 bit/s in su, si configura il programma di comunicazione per una velocità di 57600 bit/s, o superiore (purché la porta seriale dell'elaboratore e quella del modem lo consentano); se però il programma di comunicazione non consente di impostare velocità superiori a 38400 bit/s, si deve richiedere questa velocità massima, utilizzando `'setserial'` per impostare le modalità `'spd_*'`.

Le tabelle seguenti riassumono le impostazioni necessarie in funzione della velocità del modem utilizzato:

Velocità del modem	Velocità del programma	Opzioni di setserial
300	300	'spd_normal'
1200	1200	'spd_normal'
2400	2400	'spd_normal'
9600	57600	'spd_normal'
14400	57600	'spd_normal'
28800	115200	'spd_normal'
33600	115200	'spd_normal'
56000	230400	'spd_normal'

Velocità del modem	Velocità del programma	Opzioni di setserial
300	300	'spd_normal'
1200	1200	'spd_normal'
2400	2400	'spd_normal'
9600	38400	'spd_normal'
14400	38400	'spd_hi'
28800	38400	'spd_vhi'
33600	38400	'spd_vhi'
56000	38400	'spd_shi'
56000	38400	'spd_shi'

35.8 Introduzione al PPP

PPP sta per *Point-to-point protocol*; si tratta di un protocollo adatto alle connessioni *punto-punto* (*point-to-point*) nel senso che è fatto per mettere in comunicazione solo due punti tra di loro (di solito due elaboratori).

Il PPP è un protocollo piuttosto complesso e ricco di possibilità. Consente la connessione attraverso linee seriali dirette o provviste di modem (ovvero di altri apparecchi simili, come nel caso delle linee ISDN). Può instaurare una connessione anche attraverso un collegamento preesistente, sfruttando il flusso di standard input e standard output.

Generalmente, il PPP viene utilizzato per trasportare altri protocolli, fondamentalmente IP, anche se non si tratta dell'unica possibilità. Questo, tra le altre cose, permette l'assegnazione (statica o dinamica) degli indirizzi IP, consentendo in pratica a una delle due parti di ignorare il proprio fino a che non viene instaurata la connessione.

Il PPP può gestire un sistema di autenticazione, attraverso il quale, una, o entrambe le parti, cercano di ottenere dall'altra delle informazioni necessarie a riconoscerla. A questo proposito possono essere usati due modi di autenticazione: PAP e CHAP. Nella connessione PPP non esiste un cliente e un server, tuttavia, per quanto riguarda il problema dell'autenticazione, si considera cliente quel nodo che si fa riconoscere, attraverso uno di questi protocolli PAP o CHAP, presso l'altro, che così è il server. Tuttavia, la richiesta di autenticazione è facoltativa, tanto che si può benissimo instaurare una connessione senza alcuna autenticazione, se nessuna delle due parti ne fa richiesta all'altra. Inoltre, la richiesta di identificazione può anche essere reciproca; in tal caso entrambi i nodi che si connettono sono sia cliente, sia server, a fasi alterne.

35.8.1 Funzionalità del kernel Linux

Per poter utilizzare il protocollo PPP, è necessario che il kernel Linux sia predisposto per farlo (sezione 8.3.7). Naturalmente, lo stesso kernel deve poter gestire la rete.

Se il supporto al PPP è stato inserito nella parte principale del kernel, cioè non è stato lasciato in un modulo, si può trovare tra i messaggi di avvio qualcosa come l'esempio mostrato di seguito.

```
$ dmesg | less [Invio]
```

```
PPP generic driver version 2.4.1
PPP Deflate Compression module registered
PPP BSD Compression module registered
```

Se invece si tratta di una funzionalità gestita attraverso un modulo, questa dovrebbe attivarsi automaticamente al momento del bisogno.

35.9 Funzionamento generale del demone per il PPP

I sistemi GNU dispongono generalmente del demone `'pppd'`⁵ per la gestione del protocollo PPP. Si è accennato al fatto che il PPP non prevede un cliente e un servente, anche se questi termini si usano per distinguere le parti nella fase di autenticazione. In tal senso, questo programma serve sia per attendere una connessione che per iniziarla.

Il demone `'pppd'` deve amministrare un sistema piuttosto complesso di file di configurazione e di possibili script di contorno. La maggior parte di questi dovrebbe trovarsi nella directory `'/etc/ppp/'` e, tra tutti, il file più importante è `'/etc/ppp/options'`, all'interno del quale vanno indicate le opzioni di funzionamento che si vogliono attivare in generale.

35.9.1 Struttura del sistema di configurazione

Il demone `'pppd'` può essere configurato completamente attraverso le opzioni della riga di comando. Quanto definito in questo modo prevale su qualunque altro tipo di configurazione, pertanto si utilizza tale metodo solo per variare le impostazioni definite altrimenti.

Il file di configurazione principale è `'/etc/ppp/options'`; è il primo a essere letto e, teoricamente, tutti i file di configurazione successivi possono modificare quanto definito al suo interno.

Successivamente, se esiste, viene letto il file `'~/ .ppprc'`, che potrebbe essere contenuto nella directory personale dell'utente che avvia il processo. In generale, dato il ruolo che ha il programma `'pppd'`, non si usano configurazioni personalizzate degli utenti, per cui questo file non dovrebbe esistere.

Per ultimo viene letto un file di configurazione il cui nome dipende dal tipo di dispositivo utilizzato per instaurare la connessione. Data la natura del protocollo PPP, il dispositivo in questione corrisponde generalmente a una porta seriale (`'/dev/ttyS*'`); così, questo file di configurazione specifico deve avere un nome che corrisponde al modello `'/etc/ppp/options.ttyS*'` e il suo scopo è quello di definire dei dettagli che riguardano la connessione attraverso la linea a cui si riferisce.

A titolo di esempio viene anticipato come potrebbe apparire un file di configurazione di questo tipo. Si osservi il fatto che le righe bianche e quelle vuote vengono ignorate, inoltre, il simbolo `'#'` indica l'inizio di un commento che si conclude alla fine della riga.

```
# /etc/ppp/options
# Attiva il controllo di flusso hardware (RTS/CTS).
crtscts
# Vengono utilizzati i "file lucchetto" in stile UUCP.
lock
# Utilizza un modem.
modem
```

35.9.2 Struttura del sistema di autenticazione

Si è accennato al fatto che il PPP può gestire un sistema autonomo di autenticazione. Il demone `'pppd'` è in grado di utilizzare due tecniche: PAP (*Password authentication protocol*) e CHAP (*Challenge handshake authentication protocol*).

Questi sistemi si basano sulla conoscenza da parte di entrambi i nodi di alcune informazioni «segrete» (si parla precisamente di *secret*), che vengono scambiate in qualche modo e verificate prima di attuare la connessione.

È il caso di ribadire che si tratta di procedure opzionali, pertanto dipende da ognuno dei due nodi stabilire se si pretende che l'altra parte si identifichi prima di consentire la connessione.

Per utilizzare queste forme di autenticazione, occorre stabilire un nome e un *segreto* (in pratica una parola d'ordine) per il nodo che deve potersi identificare. L'altra parte deve disporre di questa informazione per poterla confrontare quando gli viene fornita.

Il protocollo PAP prevede che una parte invii all'altra il proprio nome e il segreto (cioè la parola d'ordine) che viene utilizzato per con-

sentire o meno la connessione. Il protocollo CHAP prevede invece che una parte, mentre chiede all'altra di identificarsi invii prima il proprio nome, attendendo come risposta il nome dell'altra parte e il segreto relativo da verificare. La differenza fondamentale sta nel fatto che con il PAP, una parte inizia a identificarsi anche senza sapere chi sia la controparte, mentre nel caso del CHAP, l'identificazione viene generata in funzione del nome della controparte.

Questi segreti sono conservati nel file `'/etc/ppp/pap-secrets'` per il protocollo PAP e nel file `'/etc/ppp/chap-secrets'` per il protocollo CHAP. Le informazioni contenute in questi file possono servire per identificare se stessi nei confronti dell'altra parte, oppure per verificare l'identità della controparte.

A titolo di esempio, si potrebbe osservare il testo seguente che rappresenta il contenuto del file `'/etc/ppp/chap-secrets'` del nodo `'dinkel'`.

```
# Segreti per l'autenticazione CHAP dalla parte del nodo
# «dinkel»
# cliente   servente   segreto   indirizzi IP ammissibili
dinkel     roggen      ciao     *
```

In tal caso, se il nodo remoto inizia una richiesta CHAP identificandosi con il nome `'roggen'`, gli si risponde con il nome `'dinkel'` abbinato alla parola d'ordine `'ciao'`. Dall'altra parte, il file dei segreti CHAP corrispondente dovrebbe avere lo stesso contenuto.

```
# Segreti per l'autenticazione CHAP dalla parte del nodo
# «roggen»
# cliente   servente   segreto   indirizzi IP ammissibili
dinkel     roggen      ciao     *
```

In questi termini, nell'ambito delle forme di autenticazione usate da `'pppd'`, si parla di cliente per indicare il nodo che deve identificarsi di fronte alla controparte e di servente per indicare la parte che richiede all'altra di identificarsi. In questa logica, le voci dei file `'/etc/ppp/*-secrets'` restano uguali quando si passa da una parte all'altra.

C'è da aggiungere che l'identità di un nodo non è definita dai file `'/etc/ppp/*-secrets'`, ma dalle opzioni che vengono date a `'pppd'`, per cui, se il nodo `'roggen'` vuole potersi identificare di fronte a `'dinkel'`, si può aggiungere la voce relativa nei file rispettivi.

```
# Segreti per l'autenticazione CHAP dalla parte del nodo
# «dinkel»
# cliente   servente   segreto   indirizzi IP ammissibili
dinkel     roggen      ciao     *
roggen     dinkel      medusa   *
```

```
# Segreti per l'autenticazione CHAP dalla parte del nodo
# «roggen»
# cliente   servente   segreto   indirizzi IP ammissibili
dinkel     roggen      ciao     *
roggen     dinkel      medusa   *
```

Da quello che si legge in questo ultimo esempio: `'dinkel'` utilizza il segreto `'ciao'` per identificarsi nei confronti di `'roggen'`; `'roggen'` utilizza il segreto `'medusa'` per identificarsi nei confronti di `'dinkel'`.

La sintassi del file `'/etc/ppp/pap-secrets'` è la stessa, con la differenza che sono ammissibili delle semplificazioni descritte in seguito.

35.9.3 Interfacce PPP e funzioni privilegiate

Il demone `'pppd'`, quando riesce a instaurare una connessione, definisce dinamicamente un'interfaccia di rete `'pppn'`, dove *n* è un numero che inizia da zero. Per questo e altri motivi, `'pppd'` deve funzionare con i privilegi dell'utente `'root'`. In tal senso, la collocazione normale di questo programma è la directory `'/usr/sbin/'`.

Può darsi che si voglia concedere l'utilizzo di `'pppd'` a utenti comuni; in tal caso si può attivare il bit SUID, tenendo conto dei pericoli potenziali che questa scelta può causare.

```
# chown root /usr/sbin/pppd [Invio]
```

```
# chmod u+s /usr/sbin/pppd [Invio]
```

Tuttavia, `'pppd'` riesce ugualmente a distinguere se l'utente che lo ha avviato è `'root'` (nella documentazione originale si parla di utente privilegiato), oppure se si tratta solo di un utente comune. Ciò serve per impedire l'utilizzo di opzioni delicate agli utenti comuni.

Di solito, questa distinzione si realizza nell'impossibilità da parte degli utenti comuni di utilizzare talune opzioni che annullino l'effetto di altre stabilite nella configurazione generale del file `'/etc/ppp/options'`. Questo vincolo non è generalizzato, ma riguarda solo alcune situazioni che vengono descritte nel contesto appropriato.

35.9.4 Indirizzi IP

«

Quando il protocollo PPP viene usato per trasportare comunicazioni IP, esiste la possibilità di definire in qualche modo quali indirizzi assegnare alle due parti della comunicazione. In particolare, con IPv4 gli indirizzi possono stati fissati in anticipo, oppure ottenuti dalla controparte; con IPv6, invece, gli indirizzi sono di tipo *link-local*, dove la parte finale degli ultimi 64 bit può essere determinata in modo casuale, o da indirizzi IPv4 preesistenti, oppure fissata in modo manuale.

35.9.5 Script di contorno

«

`'pppd'` può avviare degli script di contorno, in presenza di circostanze determinate. Questi possono essere diversi, ma in particolare, quando si gestiscono connessioni IPv4, sono importanti `'/etc/ppp/ip-up'` e `'/etc/ppp/ip-down'`, a cui corrispondono IPv6 gli script `'/etc/ppp/ipv6-up'` e `'/etc/ppp/ipv6-down'`. Il primo di questi (`'/etc/ppp/ip[v6]-up'`) viene avviato subito dopo una connessione e l'instaurazione di un collegamento IP tra le due parti; il secondo (`'/etc/ppp/ip[v6]-down'`) viene eseguito quando questo collegamento viene interrotto. Questi due script ricevono gli argomenti seguenti.

```
interfaccia dispositivo_linea velocità_bps indirizzo_ip_locale indirizzo_ip_remoto opzione_ipparam
```

Nel caso particolare di IPv6, la coppia di indirizzi locale e remoto, sono di tipo *link-local*.

Ogni distribuzione GNU potrebbe adattare questi script alle proprie esigenze particolari, in modo da rendere uniforme la gestione della rete. In generale, questi file potrebbero essere vuoti del tutto; il loro contenuto generico è quello seguente:

```
#!/bin/sh
# This script is called with the following arguments:
#   Arg  Name           Example
#   $1   Interface name  ppp0
#   $2   The tty         ttyS1
#   $3   The link speed   38400
#   $4   Local IP number  12.34.56.78
#   $5   Peer IP number   12.34.56.99
#
# The environment is cleared before executing this script
# so the path must be reset
#
PATH=/usr/sbin:/sbin:/usr/bin:/bin
export PATH
# last line
```

Il sesto argomento, deriva eventualmente dall'uso dell'opzione `'ipparam'` di `'pppd'`.

35.10 Avvio e opzioni

«

La sintassi per l'avvio del demone `'pppd'` è apparentemente molto semplice.

```
pppd [opzioni]
```

Queste opzioni possono apparire indifferentemente nella riga di comando, come si vede dalla sintassi, oppure nei vari file di configurazione, tenendo conto che quelle indicate sulla riga di comando prevalgono su tutto (ammesso che ciò sia consentito all'utente che avvia `'pppd'`).

Le opzioni sono di vario tipo e a seconda di questo possono essere usate in certi modi determinati.

Tipo di opzione	Descrizione
<i>dispositivo_di_comunicazione</i>	Tra gli argomenti della riga di comando o tra le opzioni di un file di configurazione, può apparire il percorso assoluto del file di dispositivo corrispondente alla linea utilizzata. Dato l'uso che si fa solitamente di <code>'pppd'</code> , si tratta normalmente di qualcosa che rispetta il modello <code>'/dev/ttyS*'</code> . Se manca l'indicazione di tale dispositivo, <code>'pppd'</code> utilizza direttamente quello del terminale attraverso il quale è stato avviato.
<i>velocità</i>	Tra gli argomenti della riga di comando o tra le opzioni di un file di configurazione, può apparire un numero puro e semplice, che rappresenta la velocità di comunicazione in bit per secondo (simbolo «bit/s», espresso volgarmente anche come «bps»). I valori utilizzabili dipendono molto anche dal sistema operativo utilizzato; per quanto riguarda GNU/Linux si tratta di quelli che si possono indicare nella configurazione delle porte seriali.
<i>ind_ipv4_locale : ind_ipv4_remoto</i> <i>ind_ipv4_locale :</i> <i>: ind_ipv4_remoto</i>	Due numeri IPv4, separati da due punti verticali (':'), come si vede dai modelli, rappresentano rispettivamente l'indirizzo del nodo locale e quello del nodo remoto. Gli indirizzi possono essere forniti in notazione decimale puntata o in forma di nome. In condizioni normali, il valore predefinito di quello locale è il primo indirizzo IPv4 del sistema. Il valore predefinito dell'indirizzo dell'elaboratore remoto si ottiene dallo stesso nodo remoto se non viene indicato esplicitamente in alcuna opzione. L'indirizzo 0.0.0.0 equivale a fare riferimento espressamente a quello predefinito, sia per la parte locale che per quella remota.
<i>opzione_argomento</i>	Un buon numero di opzioni di <code>'pppd'</code> prevede l'indicazione di un argomento successivo. Il loro uso dovrebbe essere intuitivo; in particolare, l'argomento potrebbe essere composto da più informazioni, ma si deve trattare sempre di un corpo unico.

Tipo di opzione	Descrizione
<i>opzione_booleana</i>	<p>Le opzioni rimanenti hanno significato solo in modo binario, ovvero in modo booleano. L'indicazione di queste parole chiave manifesta l'attivazione della modalità che rappresentano.</p> <p>Nel passato, l'uso di queste opzioni è stato un po' contorto. Occorre tenere conto di alcune cose: se la parola chiave inizia con 'no', dovrebbe intendersi che si tratti della disattivazione di qualcosa, secondo il senso che avrebbe leggendo in inglese; inoltre, per un problema di compatibilità con il passato, si può invertire il senso di alcune opzioni booleane facendo precedere la parola chiave relativa dal segno '-'. Per complicare ulteriormente le cose, alcune opzioni booleane (che non sono necessariamente le stesse appena descritte) possono avere l'aggiunta del segno '+' anteriormente, per confermare il senso verbale della parola chiave relativa.</p> <p>Per esempio, 'crtsets' rappresenta la gestione del controllo di flusso hardware e 'nocrtsets' indica l'opposto; mentre in origine '-crtsets' è stato il modo corretto per indicare l'inversione di 'crtsets'.</p> <p>Nella documentazione originale non si trova una spiegazione del modo con cui si possano utilizzare questi segni aggiuntivi, che sono superati e non più documentati. Purtroppo, però, molti esempi di utilizzo di 'pppd' che si trovano ancora in circolazione, fanno riferimento al vecchio modo di utilizzare le sue opzioni.</p>

35.10.1 Opzioni principali

È già stato introdotto l'uso delle opzioni di 'pppd', che possono apparire indifferentemente nella riga di comando o nei file di configurazione. Si è già accennato anche al problema dell'uso dei simboli '-' e '+' nel caso di opzioni booleane.

Opzione booleana	Descrizione
ipcp-accept-local ipcp-accept-remote	Queste due opzioni servono ad accettare le indicazioni sugli indirizzi IPv4 provenienti dal nodo remoto. Per la precisione, 'ipcp-accept-local' fa sì che venga accettato l'indirizzo locale proposto dal nodo remoto stesso, anche se questo è stato stabilito con la configurazione; 'ipcp-accept-remote' fa sì che venga accettato l'indirizzo remoto proposto dal nodo remoto anche se questo è già stato stabilito altrimenti.
auth noauth	Con l'opzione 'auth' si richiede espressamente che il nodo remoto si identifichi per consentire la connessione; al contrario, 'noauth' annulla tale necessità. Se l'opzione 'auth' appare nella configurazione generale, cioè nel file '/etc/ppp/options', l'uso dell'opzione 'noauth' per annullare tale disposizione, diviene una facoltà privilegiata, cioè concessa solo all'utente 'root'.
crtsets xonxoff nocrtsets	Con l'opzione 'crtsets' si richiede espressamente di utilizzare un controllo di flusso hardware, ovvero RTS/CTS; con l'opzione 'xonxoff' si richiede l'opposto, cioè di utilizzare un controllo di flusso software, ovvero XON/XOFF. L'opzione 'nocrtsets' indica semplicemente di disabilitare il controllo di flusso hardware.

Opzione booleana	Descrizione
defaultroute nodefaultroute	L'opzione 'defaultroute' fa sì che 'pppd', quando la connessione tra i due nodi del collegamento è avvenuta, aggiunga un percorso di instradamento predefinito (<i>default route</i>) utilizzando il nodo remoto come router. Questo percorso di instradamento viene poi rimosso dalla tabella di instradamento di sistema quando la connessione PPP si interrompe. L'opzione 'nodefaultroute' serve a evitare che questo instradamento predefinito abbia luogo. Per la precisione, se viene utilizzato nella configurazione generale del file '/etc/ppp/options', fa sì che l'uso successivo di 'defaultroute' divenga privilegiato, cioè riservato all'utente 'root'.
modem local	L'opzione 'modem' fa sì che 'pppd' utilizzi le linee di controllo del modem. Al contrario, 'local' dice a 'pppd' di ignorarle.
login	Con l'opzione 'login' si istruisce 'pppd' di utilizzare le informazioni di autenticazione gestite dal sistema operativo per gli accessi normali (il <i>login</i> appunto), cioè quelle sugli utenti con le parole d'ordine relative, per verificare l'identità del nodo remoto che si presenta utilizzando il protocollo PAP. In pratica, in questo modo, invece di dover accedere al file '/etc/ppp/pap-secrets', la verifica dell'abbinamento nome-segredo, avviene in base al sistema locale utente-parola d'ordine. Questo meccanismo si usa frequentemente quando la connessione PPP avviene attraverso linea telefonica commutata e i nodi che possono accedere corrispondono agli utenti previsti nel sistema locale (nel file '/etc/passwd'). Perché i nodi remoti possano accedere identificandosi come gli utenti del sistema, è comunque necessario che esista una voce nel file '/etc/ppp/pap-secrets' che consenta loro di essere accettati. Di solito si usa: '* * * *', che rappresenta qualunque nome per il cliente, qualunque nome per il server, qualunque segreto (o parola d'ordine) e qualunque indirizzo IP.
lock	Fa sì che 'pppd' crei un file lucchetto (<i>lock file</i>) riferito al dispositivo utilizzato per la comunicazione, secondo lo stile UUCP. In pratica, si crea un file secondo il modello '/var/lock/LCK..ttyS*'. Ciò è utile per segnalare agli altri processi che aderiscono a questa convenzione il fatto che il tale dispositivo è impegnato. In generale, è utile attivare questa opzione.
passive silent	L'opzione 'passive' fa sì che 'pppd' tenti inizialmente di connettersi al nodo remoto e, se non ne riceve alcuna risposta, resti in attesa passiva di una richiesta di connessione dalla controparte. Normalmente questa modalità non è attiva e di conseguenza 'pppd' termina la sua esecuzione quando non riceve risposta. L'opzione 'silent', invece, indica a 'pppd' di restare semplicemente in attesa passiva di una richiesta di connessione dalla controparte, senza tentare prima di iniziarla per conto proprio.

Opzione booleana	Descrizione
debug	Abilita l'annotazione di informazioni diagnostiche sullo svolgimento della connessione all'interno del registro del sistema. Per la precisione genera messaggi di tipo 'daemon' e di livello 'debug' (si veda eventualmente la sezione 16.1).
usepeerdns	Consente di ottenere dalla controparte l'indicazione di un massimo di due server DNS, i cui indirizzi vengono poi inseriti nelle variabili di ambiente 'DNS1' e 'DNS2' (utilizzabili nello script 'ip-up'), creando anche il file '/etc/ppp/resolv.conf' , compatibile con il file '/etc/resolv.conf' normale.
nodetach	In condizioni normali, quando 'pppd' deve utilizzare un dispositivo seriale che non corrisponde anche al terminale da cui è stato avviato, questo si mette da solo sullo sfondo. Per evitarlo si può usare l'opzione 'nodetach' .
persist nopersist	Con l'opzione 'persist' si richiede a 'pppd' di ristabilire la connessione quando questa termina; al contrario, 'nopersist' indica espressamente di non ritentare la connessione. In generale, il comportamento predefinito di 'pppd' è quello per cui la connessione non viene ristabilita dopo la sua conclusione.
proxyarp noproxyarp	Con l'opzione 'proxyarp' si fa in modo di inserire nella tabella ARP di sistema (<i>Address resolution protocol</i>) una voce con cui l'indirizzo IPv4 del nodo remoto viene abbinato all'indirizzo Ethernet della prima interfaccia di questo tipo utilizzata nell'elaboratore locale. Questo trucco ha il risultato di fare apparire il nodo remoto della connessione PPP come appartenente alla rete locale dell'interfaccia Ethernet. Al contrario, 'noproxyarp' impedisce questo e se utilizzato nella configurazione generale del file '/etc/ppp/options' , fa in modo che 'proxyarp' divenga un'opzione privilegiata e quindi riservata all'utente 'root' .
require-pap refuse-pap	Con l'opzione 'require-pap' si fa in modo che 'pppd' accetti la connessione solo se riceve un'identificazione PAP valida dal nodo remoto; al contrario, l'opzione 'refuse-pap' fa sì che 'pppd' si rifiuti di fornire un'identificazione PAP alla controparte.
require-chap refuse-chap	Con l'opzione 'require-chap' si fa in modo che 'pppd' richieda alla controparte l'identificazione CHAP e, di conseguenza, che accetti la connessione solo se ciò che riceve è valido secondo il file '/etc/ppp/chap-secrets' . L'opzione 'refuse-chap' fa sì che 'pppd' si rifiuti di fornire un'identificazione CHAP alla controparte.
ipv6cp-use-ipaddr	Fa in modo di usare l'indirizzo IPv4 per ottenere l'identificatore di interfaccia per l'indirizzo <i>link-local</i> locale.

Opzione con argomento	Descrizione
connect <i>comando</i>	Permette di utilizzare il comando, che eventualmente può essere delimitato tra apici (in base alle regole stabilite dalla shell utilizzata), per attivare la comunicazione attraverso la linea seriale. Di solito serve per avviare 'chat' che si occupa della connessione attraverso il modem su una linea commutata.
disconnect <i>comando</i>	Esegue il comando o lo script indicato, subito dopo la fine della connessione. Ciò può essere utile per esempio per inviare al modem un comando di aggancio (<i>hung up</i>) se la connessione fisica con il modem non consente di inviare i segnali di controllo necessari.
mru <i>n</i>	Fissa il valore dell'MRU (<i>Maximum receive unit</i>) a <i>n</i> . 'pppd' richiede così al nodo remoto di utilizzare pacchetti di dimensione non superiore a questo valore. Il valore minimo teorico per poter usare IPv6 è 1280, il valore predefinito è 1500.
mtu <i>n</i>	Fissa il valore dell'MTU (<i>Maximum transmit unit</i>) a <i>n</i> , cioè stabilisce la dimensione massima dei pacchetti trasmessi per quanto riguarda le esigenze del nodo locale (il valore minimo teorico per poter usare IPv6 è 1280). Il nodo remoto potrebbe richiedere una dimensione inferiore.
idle <i>n_secondi</i> maxconnect <i>n_secondi</i>	L'opzione 'idle' permette di stabilire il tempo di inattività oltre il quale la connessione deve essere interrotta. Il collegamento è inattivo quando non transitano pacchetti di dati. In generale, questa opzione non è conveniente assieme a 'persist' . L'opzione 'maxconnect' permette di fissare un tempo massimo per la connessione.

Opzione con argomento	Descrizione
<code>netmask</code> <i>maschera_di_rete_ipv4</i>	Fissa il valore della maschera di rete per la comunicazione con il nodo remoto attraverso IPv4. Il valore viene indicato secondo la notazione decimale puntata. Generalmente, la maschera di rete per una connessione punto-punto, dovrebbe essere 255.255.255.255, tuttavia, se si utilizza l'opzione <code>'proxyarp'</code> per fare figurare il nodo remoto come appartenente alla rete locale Ethernet, la maschera di rete deve seguire le particolarità di quella rete.
<code>ms-dns</code> <i>indirizzo</i>	Se <code>'pppd'</code> viene utilizzato per consentire la connessione da parte di sistemi MS-Windows, questa opzione permette di comunicare loro l'indirizzo IP di un server DNS. Questa opzione può apparire due volte, per fornire un massimo di due indirizzi riferiti a server DNS.
<code>ms-wins</code> <i>indirizzo</i>	Se <code>'pppd'</code> viene utilizzato per consentire la connessione da parte di sistemi MS-Windows, o in generale SMB, questa opzione permette di comunicare loro l'indirizzo IP di un server WINS (<i>Windows Internet name service</i>). Questa opzione può apparire due volte, per fornire un massimo di due indirizzi riferiti a server WINS.
<code>kdebug</code> <i>nlivello</i>	Abilita l'emissione di messaggi diagnostici da parte della gestione del PPP interna al kernel, cosa che si traduce generalmente nell'inserimento di tali messaggi nel registro del sistema. Il valore uno permette la generazione di messaggi di tipo generale; il valore due fa sì che venga emesso il contenuto dei pacchetti ricevuti; il valore quattro fa sì che venga emesso il contenuto dei pacchetti trasmessi. Per ottenere una combinazione di queste cose, basta sommare i numeri relativi.

Opzione con argomento	Descrizione
<code>ipv6</code> <i>identificatore_di_interfaccia_locale</i> , <i>↔</i> <i>↔</i> <code>identificatore_di_interfaccia_remota</code> <code>ipv6</code> <i>identificatore_di_interfaccia_locale</i> <code>ipv6</code> , <i>identificatore_di_interfaccia_remota</i>	Permette di definire esplicitamente l'identificatore di interfaccia locale, remota, o entrambe. Si tratta di un numero di 64 bit da esprimere in forma di stringa come se fosse un indirizzo IPv6; per esempio <code>'::0001:0002'</code> .

Opzione di identificazione	Descrizione
<code>name</code> <i>nome</i>	Si tratta di un'opzione privilegiata, cioè riservata all'utente <code>'root'</code> , che permette di stabilire il nome locale utilizzato sia per la propria identificazione che per il riconoscimento di un altro nodo. In pratica, se <code>'pppd'</code> deve identificarsi nei confronti di un nodo remoto, utilizza un segreto in cui il primo campo (cliente) corrisponde a tale nome; se invece si deve riconoscere un nodo remoto che si identifica, <code>'pppd'</code> utilizza un segreto in cui il secondo campo (server) corrisponde a questo. È importante tenere presente l'ambiguità di questa opzione. Per identificare il nodo locale nei confronti del nodo remoto, sarebbe meglio utilizzare l'opzione <code>'user'</code> .
<code>remotename</code> <i>nome</i>	Definisce il nome prestabilito del nodo remoto. Questa opzione è ambigua quanto <code>'name'</code> e va utilizzata con la stessa prudenza. Potrebbe essere utile quando il nodo locale si vuole identificare presso il nodo remoto utilizzando la procedura PAP; in tal caso, dato che il nome del nodo remoto non viene rivelato in anticipo, si ha la possibilità di selezionare una voce particolare dall'elenco contenuto nel file <code>'/etc/ppp/pap-secrets'</code> , facendo riferimento al secondo campo (server). In generale, l'uso delle opzioni <code>'name'</code> e <code>'remotename'</code> dovrebbe essere sensato solo quando l'unico nodo che deve identificarsi è quello locale nei confronti di quello remoto, cioè quando non si pretende anche l'identificazione inversa. Tuttavia, se è possibile risolvere la cosa con l'uso dell'opzione <code>'user'</code> , tutto diventa più semplice.
<code>usehostname</code>	Si tratta di un'opzione con la quale si stabilisce che il nome locale corrisponda a quello del nodo. Questa opzione prevale e si sostituisce a <code>'name'</code> .
<code>domain</code> <i>dominio</i>	Nel caso sia attivata l'opzione <code>'usehostname'</code> , fa sì che il nome locale comprenda anche il dominio indicato. Questo dominio non viene aggiunto a quanto stabilito con l'opzione <code>'name'</code> .
<code>user</code> <i>nome</i>	Permette di stabilire il nome locale da utilizzare per la propria identificazione nei confronti del nodo remoto. A differenza di <code>'name'</code> , questa opzione entra in gioco solo quando il nodo locale deve identificarsi, per cui, serve a selezionare una voce dai file dei segreti, facendo riferimento al primo campo, quello del cliente. Questa opzione prevale su <code>'name'</code> , per ciò che riguarda questa situazione particolare.

35.11 File per il sistema di autenticazione

Si è già accennato all'uso dei file con cui si configurano i sistemi di autenticazione PAP e CHAP. Il loro formato è identico, anche se

le diverse caratteristiche di PAP e CHAP consentono la presenza di voci sostanzialmente differenti.

Questi file di configurazione introducono il concetto di cliente e servernte nel momento dell'autenticazione: chi chiede all'altro di identificarsi è il servernte, mentre l'altro è il cliente. Teoricamente, la richiesta di autenticazione può essere reciproca, per cui, a fasi alterne, entrambi i nodi sono sia cliente che servernte nell'ambito del sistema di autenticazione. Quando si legge un file `/etc/ppp/*-secrets` occorre sempre fare mente locale a chi sia il nodo che si identifica nei confronti dell'altro, per determinare se il nodo locale è un cliente o un servernte in quel momento.

Per quanto riguarda la sintassi di questi file, come succede spesso, le righe vuote e quelle bianche vengono ignorate; nello stesso modo viene ignorato il contenuto dei commenti introdotti dal simbolo `#` e conclusi dalla fine della riga. Le altre righe, che contengono delle voci significative, sono trattate come record suddivisi in campi attraverso degli spazi lineari (spazi veri e propri o tabulazioni), secondo la sintassi seguente:

```
cliente servernte segreto indirizzo_ip_accettabile_del_cliente..
```

Ogni voce dovrebbe avere l'indicazione dei primi quattro campi.

Dal momento che la separazione tra i campi avviene per mezzo di spazi lineari, se un campo deve contenere spazi, questi devono essere protetti in qualche modo: si possono usare gli apici doppi per delimitare una stringa, oppure si può utilizzare la barra obliqua inversa (`\`) davanti a un carattere che si vuole sia trattato semplicemente per il suo valore letterale (vale anche per gli spazi).

Possono essere utilizzati anche dei simboli jolly (dei metacaratteri), che hanno valore diverso a seconda del campo in cui appaiono. In generale però, ci si limita all'uso dell'asterisco (`*`) nel campo del cliente, in quello del servernte, o in quello del primo indirizzo IP ammissibile. L'asterisco corrisponde a qualunque nome o a qualunque indirizzo e si può usare solo se il tipo di autenticazione utilizzato lo consente.

Meritano un po' di attenzione il quarto campo e quelli successivi. Questi, eventualmente, servono a elencare una serie di indirizzi IP che possono essere utilizzati dal nodo corrispondente al cliente con quella connessione particolare; si può utilizzare anche la forma *indirizzo/maschera* per rappresentare un gruppo di indirizzi in modo più chiaro. Se non si vogliono porre limitazioni agli indirizzi IP, si deve utilizzare un asterisco (`*`).

Come ultima considerazione, occorre tenere presente che quando `pppd` cerca una corrispondenza nei file dei segreti, se c'è la possibilità di farlo, seleziona la voce più specifica, cioè quella che contiene meno simboli jolly.

35.11.1 Configurazione PAP

L'autenticazione PAP prevede che un nodo si identifichi prima di conoscere l'identità della sua controparte. In questo senso, l'indicazione del nome del servernte può essere utile solo per distinguere la coppia nome-segreto da inviare. Si osservi l'esempio seguente:

```
# Segreti per l'autenticazione PAP
# cliente servernte segreto indirizzi IP ammissibili
tizio uno tazza *
caio due capperi *
sempronio tre serpenti *
```

Concentrando l'attenzione al caso in cui sia il nodo locale a doversi identificare presso altri nodi remoti, questo potrebbe essere conosciuto con nomi differenti, a seconda del collegamento che si vuole instaurare. Osservando la prima voce dell'esempio, il nodo locale cliente è conosciuto presso il nodo `uno` (servernte) con il nome `tizio` e per quella connessione deve utilizzare il segreto `tazza`.

Dal momento che il protocollo PAP non prevede di ottenere l'informazione sul nome remoto prima di fornire la propria identità, è necessario istruire `pppd` su quale voce utilizzare. Se i nomi locali so-

no tutti diversi, è sufficiente specificare questo dato attraverso l'opzione `'name'`, ma forse sarebbe meglio l'opzione `'user'`, essendo più specifica; se invece questi nomi possono essere uguali (in alcuni o in tutti i casi), occorre specificare anche l'opzione `'remotename'`.

A questo punto, però, dal momento che il nome del servernte non viene ottenuto attraverso il protocollo PAP, quello indicato nel secondo campo delle voci del file `/etc/ppp/pap-secrets` può essere un nome di fantasia, scelto solo per comodità.⁶

Per lo stesso motivo, se i nomi dal lato cliente sono tutti diversi, ovvero si utilizza una sola voce, il nome del nodo remoto (servernte) può essere semplicemente sostituito con un asterisco, come nell'esempio seguente:

```
# Segreti per l'autenticazione PAP
# cliente servernte segreto indirizzi IP ammissibili
tizio * tazza *
caio * capperi *
sempronio * serpenti *
```

La funzione del file `/etc/ppp/pap-secrets` non si esaurisce solo nel compito di fornire l'identità del nodo locale (in qualità di cliente) quando il nodo remoto lo richiede, perché può essere usato anche per verificare l'identità del nodo remoto, quando è questo ultimo a presentarsi come cliente.

Dal file `/etc/ppp/pap-secrets` non si riesce a distinguere quando il nodo locale è un cliente e quando è un servernte. Ciò dipende dalle opzioni. Se si richiede espressamente un'autenticazione PAP attraverso l'opzione `'require-pap'`, vuol dire che il nodo remoto deve identificarsi e il suo nome deve apparire nel primo campo di una voce del file `/etc/ppp/pap-secrets` locale. In pratica, le cose non cambiano quando si legge il contenuto di questo file; sono le circostanze (ovvero le opzioni) che danno significato alle sue voci: ogni volta bisogna mettersi nei panni giusti e pensare che il nodo locale sia un cliente o un servernte a seconda della situazione.

È bene ricordare che quando si utilizza l'autenticazione PAP, dal lato del nodo che deve verificare l'identità di altri nodi, cioè dal lato del servernte, si preferisce spesso fare riferimento agli utenti registrati nel sistema, piuttosto che al contenuto del file `/etc/ppp/pap-secrets`. Per questo si utilizza l'opzione `'login'`, assieme a `'require-pap'`, ma si deve comunque aggiungere una voce particolare nel file `/etc/ppp/pap-secrets`, come mostrato nell'esempio seguente:

```
# Segreti per l'autenticazione PAP
# cliente servernte segreto indirizzi IP ammissibili
* * * *
```

È difficile spiegare le ragioni di questo, ma è così. Diversamente, occorrerebbe ripetere l'indicazione delle utenze nel file `/etc/ppp/pap-secrets`, dove nel primo campo (cliente) andrebbero i nomi degli utenti e nel terzo le parole d'ordine. In particolare, come si può intuire, la stringa nulla delimitata con gli apici doppi nella posizione del segreto, rappresenta qualunque parola d'ordine.

L'amministratore del nodo remoto che deve identificarsi, deve inserire una voce nel proprio file `/etc/ppp/pap-secrets`, dove nel primo campo (cliente) deve mettere il nominativo-utente necessario per accedere presso la controparte e, di conseguenza, nel terzo campo deve mettere la parola d'ordine di questo utente.

35.11.2 Configurazione CHAP

L'autenticazione CHAP prevede che un nodo si identifichi dopo aver conosciuto il nome della controparte. La compilazione del file `/etc/ppp/chap-secrets` segue le stesse regole del file utilizzato per l'autenticazione PAP, ma in tal caso, diventa meno probabile l'uso del jolly `*`.

L'autenticazione CHAP viene usata meno frequentemente perché con questa non è possibile fare riferimento agli utenti registrati nel sistema attraverso l'opzione `'login'`.

35.12 Script

Si è già accennato alla possibilità di affiancare a `'pppd'` alcuni script o programmi che possano essere avviati da questo in momenti determinati della fase di connessione e di disconnessione. Quando si utilizza il protocollo PPP per trasportare quello IP, sono particolarmente importanti `'ip-up'` e `'ip-down'`, oppure `'ipv6-up'` e `'ipv6-down'`, che dovrebbero essere contenuti nella directory `'/etc/ppp/'`.

Tutti gli script che `'pppd'` può gestire (non solo quelli descritti qui) sono avviati senza che `'pppd'` debba attendere la loro conclusione; inoltre ottengono tutti i privilegi dell'utente `'root'`, in modo da permettere loro di eseguire qualunque operazione, soprattutto per ciò che riguarda la configurazione della rete. Tutti i flussi standard (standard input, standard output e standard error) sono ridiretti verso `'/dev/null'`. Infine, questi dispongono solo di un numero limitato di variabili di ambiente che vengono descritte di seguito.

Variabile	Descrizione
<code>'DEVICE'</code>	Contiene il nome del dispositivo seriale utilizzato.
<code>'IFNAME'</code>	Contiene il nome dell'interfaccia di rete abbinata alla connessione PPP.
<code>'IPLOCAL'</code> , <code>'IPREMOTE'</code>	Queste due variabili contengono rispettivamente l'indirizzo IP locale e quello remoto della connessione.
<code>'PEERNAME'</code>	Contiene il nome del nodo remoto, ottenuto a seguito di un'autenticazione.
<code>'SPEED'</code>	Contiene la velocità espressa in bit/s (bps) della linea seriale.
<code>'UID'</code>	Contiene il numero UID reale dell'utente che ha avviato <code>'pppd'</code> .
<code>'USEPEERDNS'</code>	Questa variabile di ambiente viene creata se viene usata l'opzione <code>'usepeerdns'</code> .
<code>'DNS1'</code> , <code>'DNS2'</code>	Contengono rispettivamente il primo e il secondo indirizzo IP dei server DNS forniti dalla controparte, quando si utilizza l'opzione <code>'usepeerdns'</code> .

La variabile di ambiente `'USEPEERDNS'` può essere sfruttata per verificare l'utilizzo o meno di questa funzionalità, per esempio nel modo seguente:

```
#!/bin/sh
...
if [ "$USEPEERDNS" ] && [ "$DNS1" ]
then
...
fi
...
```

Come si può intuire dai nomi di questi script, `'ip[v6]-up'` viene avviato da `'pppd'` quando la connessione è attiva, mentre `'ip[v6]-down'` viene avviato quando questa connessione non è più disponibile.

Oltre alle variabili di ambiente descritte in precedenza, questi ricevono degli argomenti che potrebbero anche essere superflui:

- nome_interfaccia**
è l'equivalente del contenuto della variabile `'IFNAME'`;
- dispositivo_della_linea**
è l'equivalente del contenuto della variabile `'DEVICE'`;
- velocità_bps**
è l'equivalente del contenuto della variabile `'SPEED'`;
- indirizzo_ip_locale**
è l'equivalente del contenuto della variabile `'IPLOCAL'`;
- indirizzo_ip_remoto**
è l'equivalente del contenuto della variabile `'IPREMOTE'`;
- opzione_ipparam**
è il valore dell'opzione `'ipparam'` se questa viene utilizzata con `'pppd'`.

L'esempio seguente riguarda uno script `'ip-up'` (connessioni IPv4) con il quale si vuole fare in modo che i messaggi in coda nel sistema

locale di posta elettronica vengano inviati non appena la connessione PPP viene instaurata.

```
#!/bin/sh
# /etc/ppp/ip-up
#
# Per facilitare le cose, viene definita la variabile di
# ambiente PATH, così da poter avviare i programmi più
# facilmente.
PATH=/usr/sbin:/sbin:/usr/bin:/bin
export PATH
#
# Se l'indirizzo IP remoto corrisponde a quello che consente
# l'accesso a Internet, si invia la posta elettronica
# rimasta in coda.
case "$5" in
  111.112.113.114)
    sendmail -q
    ;;
  *)
esac
```

35.12.1 Verifica dell'ambiente

Alle volte, sembra che le cose non vadano come dovrebbero, in base a quanto si trova nella documentazione. Per esempio, nella descrizione di queste funzionalità all'interno di `pppd(8)` è specificato che questi script ricevono soltanto le variabili che sono state presentate in queste sezioni. Eppure, ci sono degli esempi di utilizzo di `'pppd'` che fanno affidamento su altre risorse. In generale, sarebbe bene fare affidamento soltanto su quanto indicato nei documenti originali, tuttavia, alle volte potrebbe essere utile sapere esattamente qual è l'ambiente che ricevono questi script e quali sono precisamente gli argomenti che gli vengono passati.

```
#!/bin/sh
/bin/echo $@ >> /tmp/ambiente-ppp
set >> /tmp/ambiente-ppp
exit 0
```

L'esempio mostra una soluzione semplicissima per ottenere tali informazioni. Può trattarsi di uno qualunque degli script che è in grado di comandare `'pppd'`, non solo quelli riferiti alle connessioni IP che sono già stati presentati. Viene accodato al file `'/tmp/ambiente-ppp'` il contenuto di tutti gli argomenti ricevuti; quindi, attraverso il comando `'set'`, viene aggiunto anche lo stato di tutto l'ambiente.

35.12.2 Gestione dinamica degli indirizzi DNS

Si è accennato all'utilizzo dell'opzione `'usepeerdns'` per ottenere automaticamente l'indicazione dei server DNS remoti, offerti dal fornitore di accesso a Internet. Per sfruttare questa possibilità, si può intervenire in due modi differenti, a seconda che si gestisca un server DNS locale o meno.

Il demone `'pppd'` crea automaticamente il file `'/etc/ppp/resolv.conf'`, contenente una o due direttive del tipo:

```
nameserver 111.112.113.1
nameserver 111.112.113.2
```

Se non si dispone di un DNS locale, è sufficiente sostituire il file `'/etc/resolv.conf'` con un collegamento simbolico che punti al file `'/etc/ppp/resolv.conf'`.

Diversamente, se si dispone anche di un server DNS locale, oppure ci sono altre direttive che si vogliono preservare, le cose si complicano, perché occorre costruire un file `'/etc/resolv.conf'` ogni volta e bisogna poi ripristinarlo alla fine del collegamento PPP. Si può intuire che per questo vadano usati opportunamente gli script `'ip[v6]-up'` e `'ip[v6]-down'`.

Semplificando molto le cose, `'/etc/resolv.conf'` potrebbe sempre essere un collegamento simbolico, che viene modificato al volo, in modo da utilizzare la configurazione normale, oppure il file `'/etc/ppp/resolv.conf'`. A titolo di esempio, nello script `'ip[v6]-up'` potrebbero essere aggiunte le istruzioni seguenti:

```
if [ "$USEPEERDNS" ] && [ "$DNS1" ]
then
  rm -f /etc/resolv.conf
  ln -s /etc/ppp/resolv.conf /etc/resolv.conf
fi
```

Supponendo che il file `/etc/resolv.conf.standard` contenga le direttive che servono quando non è più disponibile la connessione PPP, lo script `'ip[v6]-down'` potrebbero contenere anche le istruzioni seguenti:

```
rm -f /etc/resolv.conf
ln -s /etc/resolv.conf.standard /etc/resolv.conf
```

35.13 Impostazione della distribuzione GNU/Linux Debian

La distribuzione GNU/Linux Debian organizza la gestione del PPP in modo particolare, allo scopo di non dover modificare direttamente gli script `'ip-up'` e `'ip-down'`, oltre a fornire una soluzione già pronta per l'attribuzione dinamica degli indirizzi IP dei serveri DNS remoti.

Lo script `'ip-up'` esegue in sequenza tutti gli script che trova nella directory `'/etc/ppp/ip-up.d/'`, mentre lo script `'ip-down'` esegue in sequenza tutti gli script che trova nella directory `'/etc/ppp/ip-down.d/'`. Si può intendere che queste due directory non siano standard; tuttavia, con tale meccanismo, si evita che i pacchetti applicativi che devono intervenire in qualche modo nella connessione PPP, possano limitarsi a collocare i loro script in queste directory, senza modificare direttamente `'ip-up'` o `'ip-down'`.

All'interno di questo meccanismo, si inserisce anche la gestione dinamica degli indirizzi dei serveri DNS remoti. Precisamente ciò avviene per mezzo degli script `'/etc/ppp/ip-up.d/0dns-up'` e `'/etc/ppp/ip-down.d/0dns-down'` (il nome degli script inizia con uno zero, per garantire che vengano eseguiti prima degli altri, dal momento che si rispetta l'ordine alfabetico). Lo script `'0dns-up'` si limita a controllare che ci siano i presupposti necessari e che sia stato ottenuto almeno un indirizzo IP di un serveri DNS remoto; se le cose stanno così, sostituisce il file `'/etc/resolv.conf'` in modo appropriato; al termine, lo script `'0dns-down'` ripristina le cose come stavano prima della connessione PPP.

35.14 Connessioni su porte seriali

Per connettere due porte seriali di due elaboratori (cioè due unità DTE), occorre realizzare un cavo apposito, detto Null-modem. Se ne possono usare due tipi: a tre o a sette fili. Il primo permette solo una connessione con controllo di flusso software, detto anche XON/XOFF, mentre il secondo consente un controllo di flusso hardware, o RTS/CTS. La sezione 35.1.4 ne mostra lo schema di collegamento.

Dopo aver realizzato il cavo seriale, è sufficiente anche quello a soli tre fili, si può controllare il suo funzionamento collegando con questo due elaboratori. Su entrambi viene utilizzato un programma di comunicazione per tentare una trasmissione elementare.

Prima di utilizzare i programmi di comunicazione, occorre accertarsi di disporre dei file di dispositivo corretti, `'/dev/ttySn'`, ed eventualmente di un collegamento simbolico denominato `'/dev/modem'` che punti al dispositivo corrispondente alla porta seriale utilizzata per la connessione.⁷

Supponendo di utilizzare la seconda porta seriale, si potrebbe creare il collegamento nel modo seguente:

```
# ln -s -i /dev/ttyS1 /dev/modem [Invio]8
```

35.14.1 Programma di comunicazione

Una volta sistemati i collegamenti simbolici in entrambi gli elaboratori, è il momento di avviare un programma di terminale di comunicazione. Il programma di comunicazione più comune nelle distribuzioni GNU è Minicom, che viene mostrato negli esempi seguenti.

Se non si vuole intervenire sui permessi del file di dispositivo di comunicazione, occorre agire come utente `'root'`. Per questo motivo è importante fare attenzione a non salvare alcuna configurazione di Minicom, perché questa diventerebbe quella predefinita per tutti gli utenti.

Si avvia Minicom (l'eseguibile `'minicom'`) su entrambi gli elaboratori.

```
# minicom [Invio]
```

```
Welcome to minicom 1.75
```

```
Press CTRL-A Z for help on special keys
```

Attraverso i due programmi occorre configurare entrambe le porte seriali nello stesso modo. In particolare, se si utilizza un cavo seriale a tre fili, si deve specificare che la comunicazione avviene attraverso un controllo di flusso software.

```
[Ctrl a][z]
```

Con questa combinazione si ottiene il menù di Minicom.

```
Commands can be called by CTRL-A <key>
```

Main Functions	Other Functions	
Dialing directory..D	run script (Go)....G	Clear Screen.....C
Send files.....S	Receive files.....R	cOnfigure Minicom..O
comm Parameters....P	Add linefeed.....A	Suspend minicom....J
Capture on/off....L	Hangup.....H	eXit and reset....X
send break.....F	initialize Modem..M	Quit with no reset.Q
Terminal settings..T	run Kermit.....K	Cursor key mode...I
lineWrap on/off...W	local Echo on/off..E	Help screen.....Z
		scroll Back.....B

Select function or press Enter for none.

È necessario configurare la porta seriale, per quanto riguarda la velocità di comunicazione, la parità, la dimensione del *data bit* e il tipo di controllo di flusso.

```
[o]
```

Si presenta un menù di diverse scelte possibili.

```
Filenames and paths
File transfer protocols
**Serial port setup**
Modem and dialing
Screen and keyboard
Save setup as dfl
Save setup as..
Exit
```

Si deve selezionare la voce *Serial port setup*, spostando il cursore con i tasti freccia e premendo `[Invio]` alla fine.

```
A - Serial Device      : /dev/modem
B - Lockfile Location  : /var/lock
C - Callin Program    :
D - Callout Program   :
E - Baud/Par/Bits     : 38400 8N1
F - Hardware Flow Control : Yes
G - Software Flow Control : No
```

Si seleziona la voce `'E'` per modificare la velocità di comunicazione.

```
[e]
```

```
Current: 38400 8N1
```

Speed	Parity	Data
A: 300	J: None	Q: 5
B: 1200	K: Even	R: 6
C: 2400	L: Odd	S: 7
D: 9600	M: Mark	T: 8
E: 19200	N: Space	
F: 38400		
G: 57600		
H: 115200	O: 8-N-1	
	P: 7-E-1	

È il caso di utilizzare sempre blocchetti di 8 bit dati senza parità, con un bit di stop, corrispondente alla sigla convenzionale 8N1. La

velocità può essere spinta al massimo.

[h]

```
Current: 115200 8N1
```

Al termine si conferma con la semplice pressione del tasto [Invio].

[Invio]

```
A - Serial Device      : /dev/modem
B - Lockfile Location  : /var/lock
C - Callin Program    :
D - Callout Program   :
E - Baud/Par/Bits     : 115200 8N1
F - Hardware Flow Control : Yes
G - Software Flow Control : No
```

Si passa quindi a configurare il controllo di flusso. Si suppone di dovere utilizzare il controllo di flusso software perché si dispone di un cavo seriale a soli tre fili. In caso contrario si può utilizzare la configurazione opposta.

[f]

```
F - Hardware Flow Control : No
G - Software Flow Control : No
```

[g]

```
F - Hardware Flow Control : No
G - Software Flow Control : Yes
```

Si esce da questo menù con la semplice pressione del tasto [Invio].

[Invio]

Quindi si esce dal menù precedente selezionando la voce 'Exit'.

```
Filenames and paths
File transfer protocols
Serial port setup
Modem and dialing
Screen and keyboard
Save setup as df1
Save setup as..
**Exit**
```

Da questo momento, tutto quello che si digita da una parte deve apparire sullo schermo dell'altra. Questo serve a provare che la connessione è corretta.

Per terminare la connessione si può utilizzare semplicemente il comando seguente, da entrambe le parti.

[Ctrl a][q]

35.15 Connessione PPP senza autenticazione

Quando si è certi che il cavo seriale è funzionante, si può passare alla realizzazione di una connessione punto-punto con l'aiuto di 'pppd'.

La connessione PPP si presta a tanti tipi di situazione. Qui si intende mostrare il caso più semplice, in cui si utilizza solo una connessione seriale senza modem e nessuna delle due parti richiede all'altra di identificarsi.

Per poter comprendere gli esempi che vengono mostrati nelle sezioni seguenti, è necessario leggere il capitolo ??capitolo ppp??, tenendo presente che il kernel deve essere stato predisposto per il PPP.

Si considera che gli script '/etc/ppp/ip-up' e '/etc/ppp/ip-down' non siano stati predisposti.

35.15.1 Script di connessione

La cosa più semplice è la realizzazione di uno script su entrambi gli elaboratori da collegare, con l'indicazione invertita degli indirizzi IP da utilizzare. In particolare, con questo esempio, non si fa affidamento sulla configurazione generale del file '/etc/ppp/options', che si suppone assente, oppure vuoto.

Si suppone di disporre dell'indirizzo 192.168.100.1 per l'elaboratore A e 192.168.200.1 per l'elaboratore B. Si vuole utilizzare un controllo di flusso software perché si dispone di un cavo seriale a tre fili. Entrambi gli elaboratori utilizzano la seconda porta seriale.

```
#!/bin/sh

# Elaboratore A

IP_REMOTO="192.168.200.1"
IP_LOCALE="192.168.100.1"
PERIFERICA="/dev/ttyS1"
VELOCITA="115200"
C_FLUSSO="noctrlsets"

/usr/sbin/pppd \
  mru 576 \
  mtu 576 \
  lock \
  passive \
  local \
  $C_FLUSSO \
  $IP_LOCALE:$IP_REMOTO \
  $PERIFERICA \
  $VELOCITA \
  noauth \
  refuse-chap \
  refuse-pap \
  persist
```

Nello script dell'elaboratore B, basta scambiare gli indirizzi.

```
#!/bin/sh

# Elaboratore B

IP_REMOTO="192.168.100.1"
IP_LOCALE="192.168.200.1"
...
```

Una volta avviati i due script, ognuno nel proprio elaboratore, quando la connessione si instaura si può controllare con 'ifconfig' e 'route' che tutto sia in ordine.

35.15.2 Verifica della connessione

L'esecuzione dei due script porta alla definizione di una nuova interfaccia di rete, 'ppp0', con l'aggiunta di una nuova voce nella tabella di instradamento.

A# ifconfig [Invio]

```
...
ppp0  Link encap:Point-to-Point Protocol
      inet addr:192.168.100.1  P-t-P:192.168.200.1  Mask:255.255.255.0
      UP POINTOPOINT RUNNING MTU:576  Metric:1
      RX packets:5 errors:0 dropped:0 overruns:0
      TX packets:10 errors:0 dropped:0 overruns:0
```

B# ifconfig [Invio]

```
...
ppp0  Link encap:Point-to-Point Protocol
      inet addr:192.168.200.1  P-t-P:192.168.100.1  Mask:255.255.255.0
      UP POINTOPOINT RUNNING MTU:576  Metric:1
      RX packets:5 errors:0 dropped:0 overruns:0
      TX packets:10 errors:0 dropped:0 overruns:0
```

A# route -n [Invio]

```
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
192.168.200.1 0.0.0.0 255.255.255.255 UH 0 0 0 ppp0
127.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 4 lo
```

B# route -n [Invio]

```
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
192.168.100.1 0.0.0.0 255.255.255.255 UH 0 0 0 ppp0
127.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 4 lo
```

Se non ci sono altri instradamenti che creano conflitti, anche 'ping' dovrebbe funzionare.

35.15.3 Varianti

Una volta verificato che la connessione funziona, si può provare ad aumentare il valore di MTU e MRU,⁹ eventualmente si può fare anche in modo che il collegamento diventi il nuovo instradamento predefinito.

```
...
/usr/sbin/pppd \
  mru 1500 \
  mtu 1500 \
  lock \
  passive \
  local \
  $C_FLUSSO \
  $IP_LOCALE:$IP_REMOTO \
  $PERIFERICA \
  $VELOCITA \
  noauth \
  refuse-chap \
  refuse-pap \
  defaultroute \
  persist
```

Se si vuole utilizzare il controllo di flusso hardware, basta cambiare il valore della variabile 'C_FLUSSO', indicando l'opzione 'crtscts'.

```
...
C_FLUSSO="crtscts"

/usr/sbin/pppd \
...
```

Infine, si può fare in modo che ognuna delle due parti lasci che l'altra definisca il proprio indirizzo IP. Per ottenere questo è sufficiente indicare l'indirizzo relativo come 0.0.0.0.

```
...
# Elaboratore A

IP_REMOTO="0.0.0.0"
IP_LOCALE="192.168.100.1"
...
```

```
...
# Elaboratore B

IP_REMOTO="0.0.0.0"
IP_LOCALE="192.168.200.1"
...
```

35.16 Linea dedicata

Una linea dedicata, o *leased line*, è generalmente un cavetto a due fili indipendente dalla rete telefonica commutata. Il termine *leased line*, linea affittata, deriva dal fatto che in origine le leggi della maggior parte dei paesi impedivano l'utilizzo di una rete di cavi per comunicazione privati, per cui questi si potevano solo affittare.

Per quanto ci riguarda, nelle sezioni seguenti, la linea dedicata è un doppio telefonico che collega due modem, ognuno connesso al proprio elaboratore.

Per fare sì che una linea dedicata di questo tipo funzioni, occorre disporre di modem **esterni** adatti a questo, in grado di essere configurati (anche attraverso microinterruttori) in modo da essere autonomi. In pratica, questi modem devono essere capaci di ricaricare la configurazione e rimettersi automaticamente in comunicazione, senza interventi software, sia in presenza di interruzioni temporanee della linea, sia quando si interrompe e poi riprende l'erogazione dell'energia elettrica.

Nelle sezioni seguenti si mostrano alcuni esempi che possono essere provati anche senza disporre di modem particolari, allo scopo di comprendere il problema.

35.16.1 Ruolo dei modem

Quando si utilizzano i modem in questo modo, senza accedere alla rete telefonica normale, non è più necessario comporre un numero telefonico e non esiste più il segnale di libero o di occupato.

Uno dei due modem deve essere configurato in modo da ricevere una chiamata su linea dedicata; l'altro deve essere configurato per chiamare. Giusto per ricordarlo, servono i comandi AT seguenti:

Comando	Descrizione
AT&L1	è il codice necessario a informare il modem che si tratta di una connessione autonoma su linea dedicata; alcuni modem potrebbero richiedere un numero diverso, come L2 per esempio;
ATX1	è il codice necessario a fare ignorare al modem chiamante il tono di chiamata e il segnale di occupato;
ATA	è il codice necessario ad attivare il modem in ricezione; ciò comporta l'emissione da parte di quel modem della portante di ricezione;
ATD	è il codice necessario ad attivare il modem in chiamata; ciò comporta l'emissione da parte di quel modem della portante di chiamata.

In pratica, a parte le possibili esigenze particolari di un modem rispetto a un altro, il comando da dare per mettere un modem in ascolto potrebbe essere AT&L1A, mentre, per mettere l'altro modem in chiamata, si potrebbe usare il comando ATX1&L1D.

Ci sono poi altre considerazioni da fare sui modem, ma per questo è meglio leggere il *Leased line mini HOWTO* di Rob van der Putten.

Quando i due modem hanno stabilito la comunicazione, tutto funziona come se le porte seriali rispettive fossero connesse attraverso un cavo seriale Null-modem; cosa già descritta nella prima parte di questo capitolo.

35.16.2 Simulazione con l'aiuto di Minicom

Con l'aiuto di Minicom si possono inviare i comandi necessari ai due modem, in modo da poter sperimentare l'uso della linea dedicata, anche se non si dispone di modem sofisticati con tutte le caratteristiche necessarie.

Si avvia Minicom in entrambi gli elaboratori, come già visto in precedenza per la connessione seriale pura e semplice. Si configura la comunicazione se ciò è necessario, tenendo presente che utilizzando il modem è meglio che il controllo di flusso sia di tipo hardware. Quindi, da una parte si digita il comando necessario ad attivare la ricezione, dall'altra il comando per iniziare la chiamata.

AT&L1A [Invio]

ATX1&L1D [Invio]

Se tutto va bene, i due modem iniziano la negoziazione e si stabilisce la connessione. Su entrambi i programmi Minicom dovrebbe apparire la risposta 'CONNECT' seguita dalla velocità. A questo punto, scrivendo da una parte si dovrebbe vedere il risultato dall'altra parte.

Se si vuole provare a utilizzare questa comunicazione, occorre concludere il funzionamento di Minicom senza reinizializzare i modem. Questo si ottiene con la combinazione [Ctrl a][q].

35.16.3 Connessione con pppd

Quando il collegamento tra i due modem è attivo, indipendentemente dal fatto che ciò sia stato ottenuto con l'aiuto di Minicom o che i modem si siano connessi in modo autonomo in base alla loro configurazione prememorizzata, si può stabilire una connessione PPP come già visto in precedenza.

Segue lo script già visto nella prima parte di questo capitolo, ritoccato in funzione dell'uso del modem.

```
#!/bin/sh

# Elaboratore A
```

```

IP_REMOTO="192.168.200.1"
IP_LOCALE="192.168.100.1"
PERIFERICA="/dev/ttyS1"
VELOCITA="38400"
C_FLUSSO="crtscts"

/usr/sbin/pppd \
  mru 576 \
  mtu 576 \
  passive \
  modem \
  $C_FLUSSO \
  $IP_LOCALE:$IP_REMOTO \
  $PERIFERICA \
  $VELOCITA \
  noauth \
  refuse-chap \
  refuse-pap \
  persist

```

Come prima, nel secondo elaboratore gli indirizzi IP devono essere invertiti.

```

IP_REMOTO="192.168.100.1"
IP_LOCALE="192.168.200.1"

```

Riquadro 35.85. Il protocollo SLIP.

All'inizio degli anni 1990, nei sistemi GNU/Linux è stato utilizzato il programma **'slattach'** per realizzare una connessione SLIP tra due elaboratori attraverso le porte seriali. Attualmente, questo programma sembra scomparso dalle distribuzioni GNU/Linux, al suo posto, per le connessioni SLIP si trova eventualmente **'dip'** che richiede un po' di configurazione.

Tuttavia, in generale le connessioni di tipo SLIP sono superate, soprattutto in considerazione del fatto che diventa impossibile trasportare in questo modo il protocollo IPv6, salvo l'inserimento in un tunnel IPv4.

35.17 Autenticazione con il protocollo PPP

« Fino a questo punto, è stato introdotto l'uso di **'pppd'** in generale e in particolare per le connessioni senza autenticazione. Quando è necessario riconoscere una delle due parti si può distinguere tra un'autenticazione tradizionale, dove si interviene come se si fosse davanti a un terminale a digitare il nominativo-utente e la parola d'ordine, oppure attraverso il PPP stesso, con i protocolli PAP o CHAP.

« 35.17.1 Autenticazione tradizionale

L'autenticazione di tipo tradizionale prevede che il protocollo PPP sia attivato dopo il riconoscimento dell'utente che richiede l'accesso. In pratica, si tratta di una connessione remota attraverso un terminale (o meglio, attraverso un programma di emulazione come Minicom o altro); si ottiene la classica richiesta **'login:'** e **'password:'**, alla quale si risponde e al termine si ottiene l'attivazione del PPP dalla parte remota.

L'attivazione del protocollo PPP potrebbe avvenire subito dopo il riconoscimento, oppure potrebbe essere necessario inviare un ritorno a carrello aggiuntivo, o avviare un comando apposito (indicato dal fornitore di accesso).

In questa situazione, quando ci si accorge che il nodo remoto ha attivato il PPP (si vedono apparire una serie di caratteri senza senso sullo schermo del terminale), si deve chiudere il programma con cui è stata fatta la connessione, senza reinizializzare il modem, quindi si deve attivare la gestione locale del PPP, in modo da utilizzare quella linea particolare.

Volendo provare quanto descritto, si potrebbe utilizzare Minicom, come è già stato mostrato altre volte in altri capitoli. Per questo bisogna ricordare di fare riferimento al dispositivo seriale giusto, cioè quello a cui è connesso il modem, quindi si deve verificare che le impostazioni della linea seriale siano quelle desiderate. Supponendo che il modem disponga di una configurazione di fabbrica sufficientemente corretta, la si può richiamare con il comando AT&F.

Dalla porta seriale a «internet mobile»

AT&F [Invio]

OK

Dovendo utilizzare le linee italiane si impartisce il comando ATX3, in modo che venga ignorata l'assenza del tono di chiamata.

ATX3 [Invio]

OK

Infine si può passare alla composizione (il numero di telefono indicato è di pura fantasia).

ATDT0987654321 [Invio]

In tal modo dovrebbe avvenire la composizione del numero e il modem remoto dovrebbe rispondere.

```
CONNECT 9600
```

In presenza di un sistema di autenticazione tradizionale, potrebbe apparire un messaggio di benvenuto e quindi la richiesta di introdurre il proprio nominativo.

Se non dovesse apparire nulla, potrebbe essere necessario inviare un carattere qualunque, o un semplice ritorno a carrello. È necessario provare per stabilire cosa bisogna fare per iniziare il colloquio con il nodo remoto.

```
Benvenuto presso il servizio della Società ...
```

```
login:
```

In tal caso si introduce il proprio nominativo-utente (in altri termini si esegue il *login*) e si conferma con [Invio].

```
login: tizio [Invio]
```

```
password:
```

Subito dopo si ottiene la richiesta di inserimento della parola d'ordine, alla quale si risponde nel modo solito, come di fronte a un terminale Unix classico.

```
password: tazza [Invio]
```

AmMESSO che il sistema remoto riconosca l'utente, cioè la coppia utente-parola d'ordine, questo potrebbe attivare immediatamente il PPP, oppure potrebbe attendere che l'utente faccia qualcosa di specifico prima di iniziare.

Nel caso peggiore si ottiene l'invito di una shell, attraverso la quale si può interagire e fare qualcosa con il proprio accesso remoto, per esempio attivare il programma **'pppd'** personalmente. In alternativa potrebbe essere necessario fare una scelta in base a un menù di opzioni che viene proposto, oppure potrebbe essere necessario premere un [Invio] in più. In pratica, bisogna provare. Quando si vedono apparire dei simboli strani, come quanto mostrato sotto, significa che il PPP è stato attivato dalla parte remota.

```
~y}#Å!;!}!} }.%}&k'q1}'*)({}*ò>~y}#Å!;!}!} }.%}&k'q1}'*)}
```

A questo punto, basterebbe concludere il funzionamento di Minicom, ma senza reinizializzare il modem (si usa il comando [Ctrl a][q]), avviando subito dopo **'pppd'** con le opzioni opportune, in modo da sfruttare il collegamento seriale corrispondente alla connessione instaurata.

Comunque, lo scopo di utilizzare Minicom è solo quello di scoprire la procedura corretta per instaurare una connessione PPP con il nodo remoto. Quando le operazioni da farsi diventano più chiare, si può predisporre un sistema automatico, attraverso **'chat'**.

È importante osservare che, quando la connessione PPP è preceduta da un'autenticazione tradizionale, il PPP **non dovrebbe** richiedere a sua volta altre forme di autenticazione, ma ciò non può essere escluso. In pratica, questo significa che potrebbe essere necessario predisporre i file **'/etc/ppp/pap-secrets'** e **'/etc/ppp/chap-secrets'**.

35.17.2 Autenticazione attraverso il PPP

L'autenticazione attraverso il PPP salta qualunque fase introduttiva, lasciando al protocollo PAP o a quello CHAP di verificare l'identità di chi accede. Per accertarsene si può usare lo stesso sistema già visto nella sezione precedente: si utilizza Minicom per iniziare la connessione, anche attraverso la composizione del numero telefonico, quindi, senza fare nulla, oppure provando a premere qualche tasto, si ottengono solo i caratteri tipici di un protocollo PPP.

```
~y#Ã!;!} }.)%}&k'q1' }"({*0>~y#Ã!;!} }.)%}&k'q1' }"
```

In tal caso, si è costretti a predisporre i file `/etc/ppp/pap-secrets` e `/etc/ppp/chap-secrets`. Eventualmente, per questo ultimo file potrebbe essere necessario conoscere il nome con cui si presenta il nodo remoto.

35.18 Cliente PPP che utilizza un sistema di identificazione tradizionale

È stato mostrato il procedimento di accesso a un sistema che utilizza un metodo di identificazione degli utenti di tipo tradizionale. Attraverso Minicom o un altro programma simile si possono dare i comandi necessari al modem, comporre il numero ed eseguire l'accesso. Al termine, una volta avviato il PPP dalla parte remota, si può chiudere il funzionamento del programma senza reinizializzare il modem (con Minicom si usa la sequenza `[Ctrl a][q]`).

A questo punto bisognerebbe avviare la gestione locale del PPP, in modo rapido, altrimenti il nodo remoto chiude la connessione. Per farlo si potrebbe realizzare uno script che avvii `'pppd'` indicando tutte le opzioni necessarie (si vuole ignorare volutamente il file `/etc/ppp/options` per non confondere il lettore con troppe cose).

```
#!/bin/sh

/usr/sbin/pppd \
  crtscts \
  modem \
  defaultroute \
  0.0.0.0:0.0.0.0 \
  /dev/ttyS1 \
  57600
```

L'esempio mostra l'utilizzo della seconda porta seriale, `/dev/ttyS1`, specificando esplicitamente che si attende dalla parte remota l'indicazione del numero IP locale e di quello remoto.

Se il nodo remoto dovesse pretendere anche un'autenticazione PAP, o CHAP, allora si devono predisporre i file `/etc/ppp/pap-secrets` e `/etc/ppp/chap-secrets`.

Naturalmente, non è molto pratico questo sistema di connessione attraverso l'uso di Minicom. Per automatizzare il procedimento di identificazione si può inserire un programma specifico: `'chat'`.

Prima di proseguire, si tenga presente che per chiudere il funzionamento di `'pppd'`, è sufficiente inviargli un segnale di interruzione (`'SIGINT'`).

35.18.1 Chat

Il programma Chat¹⁰ costituito in pratica dall'eseguibile `'chat'`, permette di definire una comunicazione tra l'elaboratore e il modem. Il suo scopo principale è quello di stabilire una connessione tra il demone `'pppd'` locale e quello di un elaboratore remoto, quando prima è necessario procedere a un'autenticazione di tipo tradizionale.

```
chat [opzioni] [script]
```

Segue la descrizione di alcune opzioni di questo programma.

Opzione	Descrizione
<code>-f chat_file</code>	Con questa indicazione, <code>'chat'</code> legge lo script di colloquio (<i>chat script</i>) dal file indicato. L'uso di questa opzione esclude l'indicazione dei comandi di script dalla riga di comando. Il file può contenere più righe, le stringhe possono essere separate utilizzando spazi o caratteri di tabulazione.
<code>-t timeout</code>	Fissa il valore del <i>timeout</i> , cioè del tempo massimo di attesa per la ricezione di una stringa.
<code>-r report_file</code>	Definisce il nome del file per contenere il rapporto quando viene utilizzata la parola chiave <code>'REPORT'</code> . Se non si specifica questo file viene utilizzato lo <i>standard error</i> .
<code>-v</code>	Attiva la modalità dettagliata per cui viene utilizzato il registro del sistema per annotare i messaggi di <code>'chat'</code> .
<code>-V</code>	Attiva la modalità dettagliata utilizzando lo <i>standard error</i> . In tal modo possono essere visualizzati immediatamente i messaggi che intercorrono tra <code>'chat'</code> e il modem. Questa opzione non può funzionare come previsto se lo <i>standard error</i> è ridiretto altrove, per esempio quando <code>'chat'</code> viene eseguito da <code>'pppd'</code> in modalità <code>'detached'</code> .
<i>script</i>	Se non viene specificato un file di script attraverso l'opzione <code>'-f'</code> questo deve essere fornito nella riga di comando, molto probabilmente racchiudendolo tra virgolette per permettere l'inserimento di spazi.

Quando il programma termina, il codice di uscita può dare delle informazioni importanti:

Codice di uscita	Significato
0	Conclusione normale: lo script è stato eseguito senza problemi.
1	Almeno uno dei parametri non è valido.
2	Errore durante l'esecuzione: potrebbe trattarsi di un errore di lettura di un file, o la ricezione di un segnale di <code>'SIGINT'</code> .
3	Errore di <i>timeout</i> .
4	È stata ricevuta la prima delle stringhe indicata come condizione di interruzione (ABORT).
5	È stata ricevuta la seconda delle stringhe indicata come condizione di interruzione (ABORT).
6	È stata ricevuta la terza delle stringhe indicata come condizione di interruzione (ABORT).
7	È stata ricevuta la quarta delle stringhe indicata come condizione di interruzione (ABORT).
3+n	È stata ricevuta la <i>n</i> -esima delle stringhe indicata come condizione di interruzione (ABORT).

35.18.2 Script di chat

Lo script di colloquio, ovvero lo script di `'chat'`, definisce la comunicazione. Lo script consiste di una o più coppie di stringhe di *attesa e invio* separate da spazi, con una coppia opzionale di stringhe di *subattesa-subinvio*, separate da un trattino. Per esempio:

```
ogin:-BREAK-ogin: tizio ssword: tazza
```

indica che `'chat'` si aspetta di ricevere la stringa `'ogin:'`. Se ciò non avviene entro il tempo massimo stabilito (*timeout*), invia un *break* al sistema remoto e quindi attende di nuovo la stringa `'ogin:'`. Se la stringa `'ogin:'` viene ricevuta già la prima volta, la sequenza di interruzione non viene generata. Se fallisce anche la seconda volta l'attesa, `'chat'` termina l'esecuzione. Quando `'chat'` ha ricevuto la stringa `'ogin:'` invia la stringa `'tizio'` e quindi si mette in attesa di ricevere la stringa `'ssword:'`. Quando la riceve invia la stringa `'tazza'`. Alla fine di ogni stringa trasmessa da `'chat'` viene ag-

giunto un ritorno a carrello (<CR>). Al contrario, per indicare che si attende un codice di ritorno a carrello, si utilizza la sequenza '\r'. Il motivo per il quale si indica solo la parte finale delle stringhe di identificazione è che in questo modo si possono ignorare le parti di stringa superflue che potrebbero anche essere giunte alterate. Un esempio molto simile al precedente potrebbe essere:

```
ogin:--ogin: tizio ssword: tazza
```

In questo caso, se non si riceve la stringa 'ogin:' al primo tentativo, 'chat' invia un semplice ritorno a carrello e quindi attende ancora una volta.

Il programma 'chat' è in grado di riconoscere una serie di stringhe speciali che vengono descritte di seguito.

Stringa	Descrizione
'ABORT'	<p>stringhe di interruzione</p> <p>Le stringhe di interruzione permettono di interrompere la comunicazione quando il modem restituisce una parola chiave particolare. Nell'esempio seguente la sequenza non attende nulla (i due apostrofi delimitano una stringa nulla ed è quel «nulla» che si attende) e quindi invia la stringa 'ATZ':</p> <pre>ABORT BUSY ABORT 'NO CARRIER' '' ATZ OK ATDT123456 CONNECT</pre> <p>La risposta attesa è la stringa 'OK'. Quindi invia 'ATDT123456' e attende 'CONNECT'. Quando viene ricevuta anche questa ultima stringa, lo script prosegue. Se però, in qualunque momento, il modem restituisce una delle stringhe 'BUSY' o 'NO CARRIER', l'esecuzione dello script viene interrotta.</p>
'REPORT'	<p>stringhe di rapporto</p> <p>Le stringhe di rapporto permettono di registrare nel file di rapporto (si veda l'opzione '-r') gli eventi specificati. Si osservi l'esempio:</p> <pre>REPORT CONNECT ABORT BUSY '' ATZ OK ATDT123456 CONNECT</pre> <p>Questa sequenza non attende nulla (i due apostrofi delimitano una stringa nulla ed è quel «nulla» che si attende) e quindi invia la stringa 'ATZ'. La risposta attesa è la stringa 'OK'. Quindi invia 'ATDT123456' e attende 'CONNECT'. Quando viene ricevuta anche questa ultima stringa, lo script prosegue e in più viene scritto all'interno del file di rapporto la parola 'CONNECT', seguita da tutto quello che il modem ha inviato insieme fino al raggiungimento del carattere di ritorno a carrello.</p>
'TIMEOUT'	<p>tempo massimo</p> <p>Il tempo massimo iniziale è di 45 s (secondi) e può essere cambiato utilizzando il parametro '-t' oppure durante l'esecuzione dello script. Si osservi l'esempio seguente, tenendo conto che è diviso in due per motivi tipografici:</p> <pre>'' ATZ OK ATDT123456 CONNECT TIMEOUT 10 \r ->ogin:--ogin: tizio ssword: tazza</pre> <p>Prima di attendere l'invito a inserire il nominativo-utente viene cambiato il tempo massimo di attesa (il <i>timeout</i>) a 10 secondi e, prima di attendere l'invito a inserire la parola d'ordine, viene cambiato a cinque secondi. Quando viene cambiato il valore del <i>timeout</i>, questo resta così fino al cambiamento successivo.</p>
'EOT'	<p>invio del codice di fine testo</p> <p>Il simbolo di EOT può essere rappresentato con '^D'. Quando si invia questo carattere non viene aggiunto il ritorno a carrello, al contrario del solito.</p>
'BREAK'	<p>interruzione</p> <p>La stringa speciale 'BREAK' rappresenta un segnale speciale nella trasmissione. L'azione normale del modem ricevente questo segnale è quello di cambiare la velocità di trasmissione. La sequenza di interruzione può essere incorporata all'interno di una stringa utilizzando la sequenza '\K'.</p>

All'interno di uno script di colloquio, si possono inserire dei simboli speciali, rappresentati prevalentemente attraverso delle sequenze di escape del tipo '\x'. Segue l'elenco di quelle più importanti per 'chat'.

Codice	Descrizione
''	Una coppia di apici singoli o di apici doppi rappresenta la stringa nulla. Se viene inviata una stringa nulla, in pratica si invia solo un ritorno a carrello.
\b	<i>Backspace</i> .
\c	Elimina il carattere di ritorno a carrello alla fine di una riga da trasmettere. È l'unico modo per riuscire a trasmettere una stringa che non termini con il solito ritorno a carrello. Si utilizza alla fine della stringa da trasmettere e non vale per le stringhe da ricevere.
\d	Attende per un secondo. Vale solo per le stringhe da trasmettere.
\k	Inserisce un carattere <i>break</i> . Vale solo per le stringhe da trasmettere.
\n	Rappresenta un carattere <i>line feed</i> o <LF>.
\N	Invia un carattere <NUL>. Vale solo per le stringhe da trasmettere.
\p	Esegue una pausa di 0,1 s. Vale solo per le stringhe da trasmettere.
\q	Sopprime la scrittura della stringa nel registro del sistema. Al suo posto appaiono alcuni punti interrogativi. Vale solo per le stringhe da trasmettere.
\r	Invia o attende un ritorno a carrello.
\s	Rappresenta uno spazio e può essere usato quando non si vuole usare la tecnica delle virgolette per racchiudere una stringa che contiene spazi.
\t	Invia o attende un carattere di tabulazione.
\\	Invia o attende un carattere '\'
\ooo	Rappresenta un carattere in notazione ottale. Alcuni simboli non possono essere ricevuti (attesi).
^x	Rappresenta una sequenza del tipo '^A', '^B', '^C',... Per esempio, '^Q' rappresenta il codice <DC1> pari a 17 ₁₀ . Alcuni simboli non possono essere ricevuti (attesi).

35.18.3 Demone per il PPP e Chat assieme

Per automatizzare la creazione di un collegamento PPP attraverso la linea telefonica, quando il nodo remoto utilizza un sistema di autenticazione tradizionale, si può combinare l'uso di 'pppd' e di 'chat'. Per la precisione, si utilizza 'pppd' con l'opzione 'connect', attraverso la quale si avvia 'chat' allo scopo di inizializzare il modem, comporre il numero ed eseguire il procedimento di autenticazione.

La prima cosa da fare è quella di creare uno script per 'chat', adatto alle esigenze del proprio modem, ma soprattutto, in grado di eseguire l'accesso presso la macchina remota. Si osservi l'esempio seguente, che fa riferimento al file '/etc/ppp/chatscript'.¹¹

```
TIMEOUT      3
ABORT        BUSY
ABORT        'NO CARRIER'
''           \dat&F
OK           \dat
OK           \datX3
OK           \dat
OK           '\datDT 0987654321'
TIMEOUT      30
CONNECT      ''
ogin:--ogin: tizio
word:       tazza
''          ''
```

Se si osserva l'esempio, si può notare che se la stringa 'ogin:' non viene ricevuta entro 30 s, viene inviato un ritorno a carrello e quindi la si attende nuovamente. Inoltre, alla fine, anche se non è detto che sia strettamente necessario, viene inviato un ritorno a carrello senza attendere nulla.

In questa situazione, si potrebbe predisporre un altro script (questa volta uno script di shell), per avviare 'pppd' con tutte le opzioni necessarie, ma soprattutto con l'uso di 'connect' per incorporare

'chat'.

```
#!/bin/sh

/usr/sbin/pppd \
  connect "/usr/sbin/chat -v -f /etc/ppp/chatscript" \
  crtscts \
  modem \
  defaultroute \
  0.0.0.0:0.0.0.0 \
  /dev/ttyS1 \
  57600
```

Come in altri esempi, viene utilizzata la seconda porta seriale e si lascia che sia la controparte a definire gli indirizzi IP di entrambi i nodi.

Ricapitolando, in questo modo: 'pppd' apre la linea seriale; avvia 'chat' che si occupa di inizializzare il modem, di comporre il numero telefonico e di eseguire l'accesso, fino a fare partire il PPP dall'altra parte; quindi 'pppd' riprende il controllo ed è pronto per comunicare con l'altro lato della comunicazione.

Volendo, si può incorporare tutto lo script di colloquio nello script di shell che serve ad avviare 'pppd'. Così facendo, diventa tutto un po' confuso da leggere, ma può essere un modo per tenere le informazioni sul proprio accesso remoto lontane da occhi indiscreti.

Nel file *allegati/ppp/ppp-connetti.txt* si trova uno script completo che prima di avviare 'pppd' verifica che non ci sia già un'interfaccia di rete denominata 'ppp0'.

Per semplificare la chiusura del PPP, si può preparare anche uno script come il file *allegati/ppp/ppp-chiudi.txt*.

Prima di poter eseguire uno script è importante ricordare di attribuirgli i permessi di esecuzione necessari.

```
chmod +x nome_del_file
```

Come già accennato nel capitolo introduttivo all'uso di 'pppd', se si vuole permettere anche agli utenti comuni di effettuare la connessione, occorre fare in modo che 'pppd' sia SUID-root. In pratica, si verifica e se necessario si modificano i permessi di 'pppd'.

```
# ls -l /usr/sbin/pppd [Invio]
```

```
-rwxr-xr-x 1 root root 69084 Mar 25 1997 /usr/bin/pppd
```

Dal momento che manca la modalità SUID, occorre attribuirgliela.

```
# chmod u+s /usr/sbin/pppd [Invio]
```

Si verifica nuovamente per sicurezza.

```
# ls -l /usr/sbin/pppd [Invio]
```

```
-rwsr-xr-x 1 root root 69084 Mar 25 1997 /usr/bin/pppd
```

La lettera 's' minuscola segnala l'attivazione della modalità SUID e del permesso di esecuzione per l'utente proprietario.

35.19 Cliente PPP che fornisce esclusivamente un'identificazione PAP o CHAP

Se si usa esclusivamente il protocollo PPP per ottenere l'autenticazione di chi accede, la configurazione del cliente diventa più semplice. La differenza rispetto a quanto mostrato nel caso di autenticazione tradizionale, sta nel fatto che non occorre più accedere in quel modo; tuttavia resta il problema di dover inizializzare il modem e di comporre il numero telefonico.

In pratica, il procedimento è simile a quanto è già stato mostrato, nel senso che 'pppd' viene usato ancora assieme a 'chat', solo che lo script di colloquio si limita a comandare il modem.

```
TIMEOUT 3
ABORT BUSY
ABORT 'NO CARRIER'
'' \dAT&F
OK \dAT
OK \dATX3
OK \dAT
OK '\dATDT 0987654321'
```

Quello che si vede potrebbe essere il nuovo script di colloquio di 'chat'. Per il resto, l'uso di 'pppd' non cambia, a parte il fatto di dover intervenire sui file '/etc/ppp/pap-secrets' e '/etc/ppp/chat-secrets'. Quello che segue è l'esempio di '/etc/ppp/pap-secrets'; nel caso di '/etc/ppp/chat-secrets' potrebbe essere necessario indicare espressamente il nome del server, ovvero del nodo remoto.

```
# /etc/ppp/pap-secrets
# Segreti per l'autenticazione PAP
# cliente  server  segreto  indirizzi IP ammissibili
tizio      *          tazza    *
```

A questo punto, specialmente nel caso che il nodo remoto richieda l'autenticazione PAP, è necessario aggiungere al comando 'pppd' l'opzione 'user', in modo da selezionare la voce corretta nel file '/etc/ppp/pap-secrets'.

```
#!/bin/sh

/usr/sbin/pppd \
  connect "/usr/sbin/chat -v -f /etc/ppp/chatscript" \
  user tizio \
  crtscts \
  modem \
  defaultroute \
  0.0.0.0:0.0.0.0 \
  /dev/ttyS1 \
  57600
```

35.20 WvDial

WvDial¹² è un programma frontale, per sistemi GNU/Linux, per l'uso e la gestione facilitata di 'pppd' allo scopo di realizzare delle connessioni su linea commutata attraverso il modem. WvDial si prende cura di attivare la connessione, sia in presenza di un sistema di autenticazione tradizionale, sia attraverso i protocolli PAP e CHAP, senza bisogno di intervenire nella configurazione dei file '/etc/ppp/pap-secrets' e '/etc/ppp/chap-secrets'.

In condizioni normali, WvDial è in grado di configurare quasi completamente il modem, lasciando all'utente l'onere di inserire i propri dati relativi all'utenza remota presso cui si vuole connettere.

35.20.1 Configurazione automatica di WvDial

Una volta installato WvDial, se non è già il sistema di gestione dei pacchetti della propria distribuzione a provvedervi, bisogna avviare il programma 'wvdialconf' allo scopo di generare il file di configurazione iniziale: '/etc/wvdial.conf'. Ci si comporta così (servono i privilegi dell'utente 'root'):

```
# wvdialconf /etc/wvdial.conf [Invio]
```

In quel momento non si deve muovere il mouse, o comunque non si deve interagire con alcuna unità che utilizzi una porta seriale. La prima volta, si potrebbe ottenere un rapporto simile a quello seguente, dove si vede che viene individuato un modem nella seconda porta seriale:

```
Scanning your serial ports for a modem.

Port Scan<1>: Ignoring ttyS0 because /dev/mouse is a link to it.
ttyS1<1>: ATQ0 V1 E1 -- OK
ttyS1<1>: ATQ0 V1 E1 Z -- OK
ttyS1<1>: ATQ0 V1 E1 S0=0 -- OK
ttyS1<1>: ATQ0 V1 E1 S0=0 &C1 -- OK
ttyS1<1>: ATQ0 V1 E1 S0=0 &C1 &D2 -- OK
ttyS1<1>: ATQ0 V1 E1 S0=0 &C1 &D2 S11=55 -- OK
ttyS1<1>: ATQ0 V1 E1 S0=0 &C1 &D2 S11=55 +FCLASS=0 -- OK
ttyS1<1>: Modem Identifier: ATI -- 5601
ttyS1<1>: Speed 2400: AT -- OK
ttyS1<1>: Speed 4800: AT -- OK
ttyS1<1>: Speed 9600: AT -- OK
```

```

ttyS1<+1>: Speed 19200: AT -- OK
ttyS1<+1>: Speed 38400: AT -- OK
ttyS1<+1>: Speed 57600: AT -- OK
ttyS1<+1>: Speed 115200: AT -- OK
ttyS1<+1>: Max speed is 115200; that should be safe.
ttyS1<+1>: ATQ0 V1 E1 S0=0 &C1 &D2 S11=55 +FCLASS=0 -- OK
Port Scan<+1>: S3

Found a modem on /dev/ttyS1.
/etc/wvdial.conf<Warn>: Can't read config file /etc/wvdial.conf:
  No such file or directory
ttyS1<Info>: Speed 115200; init 'ATQ0 V1 E1 S0=0 &C1 &D2 S11=55 +FCLASS=0'

```

In condizioni normali, il programma è in grado di individuare il modem e di determinare le sue capacità. Da questo si ottiene un file di configurazione iniziale abbastanza completo, simile a quello seguente:

```

[Dialer Defaults]
Modem = /dev/ttyS1
Baud = 115200
Init1 = ATZ
Init2 = ATQ0 V1 E1 S0=0 &C1 &D2 S11=55 +FCLASS=0
; Phone = <Target Phone Number>
; Username = <Your Login Name>
; Password = <Your Password>

```

Ammesso di utilizzare effettivamente un modem per linea telefonica, data la caratteristica delle linee italiane, per cui non esiste il tono di chiamata, è necessario aggiungere il comando ATX3; inoltre, come si intuisce, vanno definite le ultime tre direttive che appaiono opportunamente commentate. In altri termini, il file va modificato più o meno come si vede nell'esempio seguente, dove i dati relativi all'utenza sono ovviamente inventati:

```

[Dialer Defaults]
Modem = /dev/ttyS1
Baud = 115200
Init1 = ATZ
Init2 = ATQ0 V1 E1 S0=0 &C1 &D2 S11=55 +FCLASS=0
Init3 = ATX3
Phone = 0987 654321
Username = tizio
Password = supersegretissimo

```

In condizioni normali, è sufficiente avviare l'eseguibile `wvdial` con i privilegi dell'utente `root` e la connessione dovrebbe instaurarsi senza altri problemi. Eventualmente, con l'opzione `-c` è possibile indicare un file di configurazione differente:

```
# wvdial -c /etc/wvdial.1.conf [Invio]
```

C'è da considerare che se il file di configurazione di `wvdial` contiene dati delicati come la parola d'ordine per accedere al servizio remoto, il file deve essere reso inaccessibile agli utenti estranei; inoltre, si può valutare la possibilità di impostare l'eseguibile `wvdial` come SUID-root.

35.20.2 Configurazione automatica e trasparente di pppd

È importante sapere cosa fa WvDial con la configurazione di `pppd`, anche se può essere comodo lasciare fare tutto a lui. Ciò consente di capire in che modo va usato e quali possono essere eventualmente le limitazioni.

In condizioni normali, WvDial fa affidamento sul fatto che `pppd` riconosca l'opzione `call`, con la quale si seleziona un file di configurazione specifico nella directory `/etc/ppp/peers/`. Per la precisione, WvDial fa in modo che venga letto il file `/etc/ppp/peers/wvdial` che si solito dovrebbe trovarsi già lì a seguito della sua installazione.

Oltre a questo, l'eseguibile `wvdial` crea o modifica autonomamente i file `/etc/ppp/pap-secrets` e `/etc/ppp/chap-secrets`, in base alle informazioni sull'utenza che appaiono nel file di configurazione. Per questo, quando viene eseguito, ha bisogno di avere i privilegi dell'utente `root`, che fortunatamente rimangono inaccessibili agli utenti comuni.

In condizioni normali, precisamente quando è previsto l'uso di una sola utenza remota, sarebbe sufficiente utilizzare l'eseguibile `wvdial` con i privilegi dell'utente `root` solo la prima volta, dal momento che le modifiche apportate a questi file non avrebbero bisogno successivamente di essere aggiornate.

Seguendo l'esempio già visto in precedenza, in entrambi i file `/etc/ppp/pap-secrets` e `/etc/ppp/chap-secrets` apparirebbe in coda la riga seguente:

```
tizio * supersegretissimo
```

35.20.3 Configurazione manuale

La configurazione automatica, con gli aggiustamenti necessari che sono stati mostrati, può essere molto conveniente per un principiante; tuttavia, la configurazione manuale di WvDial consente di aggiungere delle indicazioni molto utili; in particolare permette di definire utenze differenti, da selezionare attraverso argomenti della riga di comando di `wvdial`.

Il file in questione può contenere righe bianche e vuote, che vengono ignorate, così come sono ignorate le righe che iniziano con un punto e virgola. Per il resto si tratta di direttive, nella forma

```
attributo = valore_assegnato
```

che possono essere raggruppate in sezioni precedute dalla dichiarazione

```
[Dialer nome_della_sezione]
```

In particolare, come è già stato visto nell'esempio introduttivo, tutte le direttive che non ricadono in sezioni particolari, fanno parte della sezione predefinita, denominata `Defaults`:

```
[Dialer Defaults]
...
...
```

Altre sezioni possono essere dichiarate per definire delle varianti nella configurazione, che poi vengono selezionate semplicemente nominandole nella riga di comando di `wvdial`. Per la precisione, tutte le sezioni aggiunte ereditano la configurazione della sezione predefinita, aggiungendo o sostituendo delle dichiarazioni particolari. Si osservi l'esempio seguente:

```

[Dialer Defaults]
Modem = /dev/ttyS1
Baud = 115200
Init1 = ATZ
Init2 = ATQ0 V1 E1 S0=0 &C1 &D2 S11=55 +FCLASS=0
Init3 = ATX3
Phone = 0987 654321
Username = tizio
Password = supersegretissimo

[Dialer treviso]
Phone = 0422 654321

[Dialer venezia]
Phone = 041 654321

[Dialer rimini]
Phone = 0541 654321

```

In questo caso, come si può intuire, ogni sezione aggiunta serve a definire un numero telefonico differente, lasciando tutti gli altri dati come fissato nella sezione predefinita.

Naturalmente, la possibilità di gestire sezioni aggiuntive permette anche di intervenire su altre variabili, come la configurazione del modem e la modalità di composizione del numero telefonico:

```
[Dialer silenzioso]
Init4 = ATM0

[Dialer impulsivi]
Dial Command = ATDP
```

Nell'esempio si vede la definizione di due sezioni: la prima permette di aggiungere un'istruzione al modem, in modo che l'altoparlante risulti disattivato completamente; la seconda permette di richiedere espressamente la composizione a impulsi (il vecchio sistema «decadico» dei telefoni a disco).

Segue la descrizione di alcune direttive di configurazione.

Direttiva	Descrizione
Inherits = <i>sezione</i>	Consente di ereditare le direttive di un'altra sezione, tenendo conto che la sezione predefinita viene ereditata automaticamente.
Modem = <i>file</i>	Definisce il file di dispositivo relativo alla porta seriale cui è connesso il modem.
Baud = <i>velocità_porta_seriale</i>	Definisce la velocità di comunicazione con il modem attraverso la porta seriale; in altri termini, si tratta della velocità della porta seriale, espressa in bit/s.
Init <i>n</i> = <i>comando_at_per_il_modem</i>	Le direttive da 'Init1' a 'Init9' permettono di definire diverse stringhe di inizializzazione del modem, in sequenza. La prima a essere eseguita è la direttiva 'Init1'; di seguito vengono eseguite le altre, fino a un massimo di nove.
Phone = <i>numero_telefonico_da_chiamare</i>	Definisce il numero telefonico per raggiungere il fornitore del servizio.
Dial Command = <i>comando_at_per_il_modem</i>	Il comando AT necessario per iniziare la composizione telefonica. Il comando predefinito è ATDT, per la composizione a toni (multifrequenza).
Login = <i>nominativo_utenza_remota</i>	Il nominativo utente da usare per la connessione remota.
Password = <i>parola_d'ordine</i>	La parola d'ordine da usare per l'autenticazione remota.
PPPD Path = <i>percorso_di_avvio_di_pppd</i>	In caso di necessità, permette di definire il percorso assoluto di 'pppd'. In modo predefinito, viene usato il percorso '/usr/sbin/pppd'.
Force Address = <i>ip_statico_locale</i>	Se ciò può essere utile, permette di definire l'indirizzo IP statico locale.
Auto Reconnect = {on off}	Questa opzione, attiva in modo predefinito, serve a ottenere il ripristino della connessione se questa cade per qualche motivo.

35.20.4 Avvio e funzionamento

WvDial si avvia attraverso l'eseguibile 'wvdial', il quale funziona in primo piano in modo predefinito:

```
wvdial [opzioni] {sezione...}
```

Per concludere la connessione e il funzionamento del programma, si utilizza il segnale di interruzione, che si ottiene normalmente con la combinazione [Ctrl c]. Segue la descrizione di alcuni esempi.

- # wvdial [Invio]
Avvia il programma in primo piano, in base alla configurazione della sezione predefinita.
- # wvdial -C /etc/wvdial.1.conf [Invio]
wvdial --config=/etc/wvdial.1.conf [Invio]
Avvia il programma in primo piano, in base alla configurazione della sezione predefinita, del file '/etc/wvdial.1.conf'.
- # wvdial > /var/log/wvdial.log 2>&1 & [Invio]
Avvia il programma sullo sfondo, ridirigendo i flussi di standard output e standard error nel file '/var/log/wvdial.log'.
- # wvdial treviso silenzioso [Invio]
Avvia il programma richiedendo espressamente l'utilizzo delle sezioni 'treviso' e 'silenzioso' dal file di configurazione.

35.21 Connessione mobile con «chiavetta»

Le connessioni con «chiavetta» USB, dotata di scheda telefonica per il collegamento alla rete GSM/UMTS, avvengono attraverso il protocollo PPP, come si farebbe per un modem tradizionale, con la differenza che non è necessario fornire dati per l'autenticazione, perché questi sono contenuti implicitamente nella scheda telefonica stessa.

Figura 35.115. Dispositivo HUAWEI E1800.



All'avvio del sistema operativo, qualunque esso sia, tali unità devono risultare staccate: vanno inserite solo durante il funzionamento. Inoltre, prima di procedere con una connessione, è necessario attendere che queste unità abbiano già negoziato automaticamente il protocollo utilizzabile con il ponte radio locale: di solito lo si determina attraverso un led lampeggiante, il cui colore varia in funzione del tipo di collegamento disponibile.

35.21.1 Problematiche con i sistemi GNU/Linux

Tralasciando il caso delle unità di memorizzazione, le unità USB richiedono quasi sempre la disponibilità nel kernel Linux di codice non libero, benché disponibile gratuitamente. Si tratta in particolare di microcodice collocato generalmente nella directory '/lib/firmware/' che il kernel invia all'unità, una volta individuata. Ciò significa che per poter accedere a queste unità è necessario disporre di un kernel completo, eventualmente realizzato a partire dai sorgenti originali.

Un altro aspetto importante da considerare consiste nel fatto che le unità USB che non sono rivolte specificatamente alla memorizzazione dei dati, tendono a essere realizzate con due modalità di funzionamento: una «normale» e l'altra in qualità di memoria solita o di disco ottico (in sola lettura). In pratica, il funzionamento in veste di unità di memorizzazione consente di allegare al dispositivo tutto il software e la documentazione necessari per l'uso con questo o quel sistema operativo.

Per poter controllare le due modalità di funzionamento è necessario disporre di USB_ModeSwitch (http://www.draisberghof.de/usb_modeswitch/) che nelle distribuzioni GNU/Linux Debian comporta l'installazione dei pacchetti 'usb_modeswitch' e 'usb_modeswitch_data'. In generale è importante che i pacchetti di USB_ModeSwitch siano aggiornati, in particolare 'usb_modeswitch_data', per garantire il riconoscimento corretto dei dispositivi (diversamente diventa necessario aggiungere dei file nella directory '/etc/usb_modeswitch.d/', con le informazioni sui dispositivi particolari gestiti).

Va osservato che il funzionamento di USB_ModeSwitch dipende da uDev, senza il quale non potrebbe avvenire un riconoscimento dei dispositivi contestualmente con il loro inserimento nelle porte USB.

35.21.2 Modem USB

«

Dato per assunto che il proprio dispositivo sia gestito correttamente da uDev e da USB_ModeSwitch, l'inserimento di questo tipo di dispositivo comporta normalmente la creazione di alcuni file di dispositivo, con nomi del tipo '/dev/ttyUSB0', '/dev/ttyUSB1',.... Pertanto, per interagire con tali unità si deve accedere a questi file di dispositivo (generalmente solo il primo), come se si trattasse di modem tradizionali.

35.21.3 WvDial e le «chiavette»

«

Per instaurare una connessione attraverso una «chiavetta» Internet, se questa viene gestita correttamente da uDev e USB_ModeSwitch, è sufficiente avvalersi di WvDial, il quale poi gestisce automaticamente il demone 'pppd'. Eventualmente è facoltà dell'amministratore di sistema decidere se WvDial debba poter essere avviato da qualunque utente, nel qual caso gli si può attribuire il permesso SUID-root.

Tutto quello che serve per la connessione è la preparazione del file di configurazione di WvDial. Si osservi l'esempio seguente, in cui le righe sono numerate per motivi tipografici:

```

1 [Dialer Defaults]
2 Modem = /dev/ttyUSB0
3 Baud = 115200
4 Init = AT&F Q0 V1 E1 S0=0 &C1 &D2 +FCLASS=0
5 Init2 = at+cgdcont=1,"IP","ibox.tim.it"
6 Phone = +99#
7 Username = tim
8 Password = tim
9 Stupid Mode = true

```

Riga n. 2. Si può osservare che il dispositivo da usare per il collegamento con il modem della chiavetta è il primo: '/dev/ttyUSB0'.

Riga n. 3. La velocità di comunicazione deve essere verificata, partendo dal valore che si vede nell'esempio, provando eventualmente a usare anche velocità maggiori: 230400, 460800 e 720000.

Riga n. 5. Nella seconda stringa di inizializzazione del modem, si vede l'indicazione del nome APN (*Access point name*) che in questo caso corrisponde a 'ibox.tim.it'. Il nome APN è molto importante per individuare il servizio a cui ci si connette e uno stesso gestore potrebbe distinguere nomi differenti in base alle tariffe e condizioni del servizio.

Riga n. 6. Il numero di telefono rimane generalmente quello che si vede nell'esempio (nel caso di Vodafone potrebbe essere invece '*99***1#', ma ciò deve essere verificato).

Righe 7 e 8. I dati identificativi dell'utente non servono, perché la chiavetta di identifica attraverso la scheda telefonica (SIM) che vi viene installata al suo interno. Tuttavia, WvDial richiede l'indicazione di questi valori che possono essere annotati con dati di fantasia.

Supponendo che il file di configurazione dell'esempio sia '/etc/wvdial/tim.conf', si potrebbe avviare WvDial nel modo seguente:

```
# wvdial -C /etc/wvdial/tim.conf [Invio]
```

```

--> WvDial: Internet dialer version 1.61
--> Cannot get information for serial port.
--> Initializing modem.
--> Sending: AT&F Q0 V1 E1 S0=0 &C1 &D2 +FCLASS=0
OK
--> Sending: at+cgdcont=1,"IP","ibox.tim.it"
at+cgdcont=1,"IP","ibox.tim.it"
OK
--> Modem initialized.
--> Sending: ATDT+99#
--> Waiting for carrier.
ATDT+99#
CONNECT
--> Carrier detected. Waiting for prompt.

```

Siccome la controparte non offre alcun invito tradizionale, WvDial passa dopo un po' all'avvio di 'pppd' e al conseguente aggiornamento del file '/etc/resolv.conf' e dell'instradamento predefinito:

```

--> Don't know what to do! Starting pppd and hoping for ↵
↵ the best.
--> Starting pppd at Tue May 3 21:00:14 2011
--> Pid of pppd: 9327
--> Using interface ppp0
--> pppd: @"rs[06][08]o[06][08][01]
--> local IP address 109.52.166.119
--> pppd: @"rs[06][08]o[06][08][01]
--> remote IP address 10.64.64.64
--> pppd: @"rs[06][08]o[06][08][01]
--> primary DNS address 217.200.200.42
--> pppd: @"rs[06][08]o[06][08][01]
--> secondary DNS address 213.230.129.10
--> pppd: @"rs[06][08]o[06][08][01]

```

A questo punto WvDial deve essere lasciato in funzione, altrimenti la combinazione di tasti [Ctrl c] ne concluderebbe il funzionamento, con la conclusione corretta della connessione. Pertanto, da un altro terminale si può verificare l'aggiornamento di '/etc/resolv.conf' e dell'instradamento predefinito:

```
# cat /etc/resolv.conf [Invio]
```

```

nameserver 217.200.200.42
nameserver 213.230.129.10

```

```
# route -n [Invio]
```

```

Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
10.64.64.64 0.0.0.0 255.255.255.255 UH 0 0 0 ppp0
127.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 lo
224.0.0.0 0.0.0.0 240.0.0.0 U 0 0 0 eth0
0.0.0.0 0.0.0.0 0.0.0.0 U 0 0 0 ppp0

```

Va osservato che se prima della connessione esisteva un instradamento predefinito, questo non viene modificato. In quel caso, però, tale instradamento va rimosso prima di avviare WvDial, per esempio con il comando seguente:

```

# route del -net default ;↵
↵ wvdial -C /etc/wvdial/tim.conf [Invio]

```

35.22 Riferimenti

«

- Robert Hart, *PPP HOWTO*, <http://www.google.it/search?q=Robert+Hart%2C+PPP+HOWTO>
- pppd(8)
- Rob van der Putten, *Leased line mini HOWTO*, http://tldp.org/HOWTO/html_single/Leased-Line/
- Egil Kvaleberg, *ISP-Hookup HOWTO*, <http://tldp.org/HOWTO/ISP-Hookup-HOWTO.html>
- Wikipedia, *Access Point Name*, http://en.wikipedia.org/wiki/Access_Point_Name

- Josua Dietze, *USB_ModeSwitch - Handling Mode-Switching USB Devices on Linux*, http://www.draisberghof.de/usb_modeswitch/

¹ **Setserial** GNU GPL

² AT sta per *Attention*.

³ **Minicom** GNU GPL

⁴ **Seyon** GNU GPL

⁵ **PPPd** software libero con licenza speciale

⁶ Per qualche motivo, se si utilizza il protocollo di autenticazione PAP per la propria identificazione e si vuole usare l'opzione '**remotename**', è necessario anche aggiungere l'opzione '**user**', o '**name**', per specificare il nome locale del cliente.

⁷ In generale, l'uso di un collegamento al file di dispositivo della porta seriale corrispondente al modem è sconsigliabile. Negli esempi viene fatto sempre riferimento al file '/dev/modem', ma ognuno può sostituire questo nome con quello più appropriato per il proprio sistema.

⁸ L'opzione '**-i**' fa sì che il collegamento '/dev/modem' possa essere sostituito se già esistente, chiedendo prima una conferma.

⁹ Se si intende instaurare un collegamento per trasportare direttamente IPv6, diventa indispensabile aumentare questi valori.

¹⁰ **Chat** dominio pubblico

¹¹ La scelta della collocazione e del nome di questo script è personale. In questo caso è stato messo nella directory '/etc/ppp/', anche se ciò potrebbe essere discutibile. Dal momento che contiene informazioni riservate, precisamente ciò che è necessario per accedere presso il server remoto a cui ci si connette, può darsi che sia meglio «nascondere» in qualche modo.

¹² **WvDial** GNU LGPL