Linux IPX-HOWTO

Kevin Thorpe, `kevin@pricetrak.com`

v2.3, 06 Maggio 1998

Scopo di questo documento è descrivere come ottenere, installare e configurare diversi strumenti disponibili per utilizzare il protocollo IPX per kernel Linux. Revisione e manutenzione della traduzione italiana: *Andrea Girotto (andrea.girotto@usa.net)* .

Contents

1 Introduzione a questo documento

Questo è il Linux **IPX-HOWTO**. Insieme a questo, è consigliabile leggere (o aver letto) anche il documento *NET-3-HOWTO* di Linux.

1.1 Modifiche dalla versione precedente

Cambio di autore Sentiti ringraziamenti a Terry Dawson per avermi ceduto questo documento e congratulazioni per essere diventato padre :-).

Aggiunte Aggiunta una breve spiegazione di IPX: un tentativo di risposta alle numerose confuse domande poste sulle liste di discussione.

Correzioni/ggiornamenti Nuova versione di ncpfs che supporta login NDS. Si tratta di beta test e potrebbe essere proibita nel proprio paese in seguito all'uso di una tecnologia coperta da registrazione.

Aggiunta del supporto dei permessi (trustee permission) del server in mars_nwe. Ancora in beta test.

1.2 Introduzione

Paragonata ad altri sistemi operativi di tipo Unix l'implementazione di rete del kernel di Linux è innovativa. La possibilità di seguire un nuovo approccio nello sviluppo del kernel, ha consentito il supporto (previa compilazione) di diversi protocolli non TCP/IP. Il protocollo IPX è uno di questi.

Il kernel Linux supporta solamente il protocollo IPX. Non è ancora in grado di gestire protocolli come IPX/RIP, SAP o NCP, che sono implementati da altro software come documentato in questo HOWTO.

Il supporto IPX fu sviluppato originariamente da *Alan Cox (alan@lxorguk.ukuu.org.uk)* ed è stato significativamente accresciuto da *Greg Page (greg@caldera.com)* .

2 Avvertenza

L'autore non conosce e non può conoscere tutto lo scibile sul software di rete per Linux. Si prega di accettare e considerare il fatto che questo documento probabilmente contiene errori. Per informazioni più dettagliate e accurate, è necessario leggere i file README inclusi con ogni software descritto. Si tenterà di mantenere questo documento il più possibile esatto e aggiornato. Le versioni del software corrispondono a quelle disponibili al momento in cui si scrive.

Né l'autore, né gli sviluppatori del software descritto sono in grado di offrire protezione nei confronti delle vostre azioni. Qualora il software, anche se configurato come descritto in questo documento, dovesse causare dei problemi alla rete, la responsabilità è solo vostra. Questo avvertimento è dovuto al fatto che il disegno e la configurazione di una rete IPX non sempre sono semplici e talvolta possono sorgere interazioni indesiderate con altri router o file server, causati da errori in fase di progettazione o configurazione. L'inserimento di questo avvertimento è stato richiesto da coloro che hanno imparato la lezione sulla propria pelle.

3 Documentazione

Questo documento presume che si sappia compilare un kernel Linux con le opzioni di rete appropriate e che si conoscano i tool di rete basilari come `ifconfig` e `route`. Per maggiori informazioni, si rimanda al documento *NET-3-HOWTO* , che descrive proprio questi argomenti.

Altri documenti HOWTO, che potrebbero essere utili sono i seguenti.

Ethernet-HOWTO

Descrive i dettagli di configurazione di un dispositivo Ethernet per Linux.

PPP-HOWTO

Il supporto IPX è disponibile a partire dalle versioni *2.2.0d* dell'implementazione PPP di Linux.

3.1 Nuove versioni di questo documento

Se questa copia del documento ha più di due mesi, si raccomanda caldamente di ottenere una nuova versione. Il supporto di rete per Linux è in continua evoluzione, pertanto anche questo documento viene modificato molto spesso.

3.1.1 Risorse internazionali

L'ultima versione rilasciata di questo documento può sempre essere ottenuta tramite ftp anonimo da:

`ftp:`/sunsite.unc.edu/pub/Linux/docs/HOWTO/IPX-HOWTO>/ o: `ftp://sunsite.unc.edu/pub/Linux/docs/HOWTO/ot`

Tramite il World Wide Web dal *Linux Documentation Project Web Server* `http://sunsite.unc.edu/LDP/linux.html` , alla pagina:

IPX-HOWTO `http://sunsite.unc.edu/LDP/HOWTO/IPX-HOWTO.html` o direttamente da me, `<kevin@pricetrak.com>`. In alternativa, possono essere utilizzati i newsgroups: `comp.os.linux.networking`, `comp.os.linux.answers` e `news.answers`

3.1.2 Risorse italiane

Questo documento e altri correlati sono disponibili presso i siti *Ftp (ftp.pluto.linux.it)*

e *Web* del *PLUTO (http://www.pluto.linux.it)* .

3.2 Feedback

Si prega di inviare direttamente all'autore ogni commento (in inglese, NdT), aggiornamento o suggerimento, all'indirizzo `<kevin@pricetrak.com>`. Prima verrà ricevuta una segnalazione, prima sarà possibile aggiornare e correggere questo documento. Se dovessero esserci problemi, per favore scrivere direttamente all'autore in quanto molto raramente legge i gruppi di discussione.

3.3 Mailing list

Esiste una mailing list per la discussione dei diversi pacchetti di software Linux IPX descritti in questo documento. È possibile iscriversi inviando un messaggio a: *listserv@sh.cvut.cz*

con le parole “`add linware`” nel corpo del messaggio. Per comunicare con la lista si deve scrivere a: *linware@sh.cvut.cz* . L'autore consulta regolarmente questa lista.

L'archivio della mailing list si trova all'indirizzo

www.kin.vslib.cz, /hypermail/linware/ `http://www.kin.vslib.cz/hypermail/linware/` .

4 Alcuni dei termini utilizzati in questo documento

Si useranno spesso i termini *client* e *server*. Si tratta di concetti molto specifici, tuttavia in questo documento si è leggermente generalizzata la loro definizione, assegnandogli il seguente significato:

client

Macchina o programma che inizia un'azione o una connessione allo scopo di ottenere l'utilizzo di servizi o di dati.

server

Macchina o programma che accetta connessioni in entrata da macchine remote ed è in grado di fornire a esse servizi o dati.

Queste definizioni non sono completamente esatte, tuttavia forniscono un modo per distinguere gli estremi dei sistemi peer to peer come *SLIP* o *PPP*, i quali, di fatto, non hanno veri e propri client e server. Altri termini che si troveranno sono i seguenti.

Bindery

Il **bindery** è un database specializzato in grado di memorizzare su un fileserver Novell informazioni riguardo la configurazione della rete. I client di rete possono interrogare il **bindery** per ottenere informazioni circa i servizi disponibili, l'instradamento (routing) e altre informazioni utente.

Tipo di trama (frame type)

Termine utilizzato per descrivere il protocollo utilizzato per trasportare i datagrammi IPX (e IP) lungo i segmenti di una rete di tipo ethernet. I quattro più comuni sono:

Ethernet_II

È una versione riveduta dello standard ethernet DIX originale. A Novell è stato assegnato un identificativo di protocollo formale e questo significa che IPX e IP possono coesistere in un ambiente Ethernet_II. Questo protocollo viene comunemente utilizzato in ambienti Novell e si tratta di una buona scelta.

802.3

È un protocollo I.E.E.E. che definisce un meccanismo CSMA/CD (*Carrier Sense Multiple Access with Collision Detection*). Si basa sullo standard Ethernet DIX originale, con una modifica rilevante: l'identificativo di protocollo (type field) è stato convertito in una lunghezza di campo (length field). Questo è il motivo per cui IPX non dovrebbe essere eseguito con questo protocollo. IEEE 802.3 è stato progettato per trasportare **solo** frame IEEE 802.2 ma esistono implementazioni che lo utilizzano per trasportare direttamente i frame e di fatto funzionano. Si consiglia di evitarlo a meno che non si stia tentando di interagire con una rete già configurata per esso.

802.2

È un protocollo I.E.E.E. che definisce un insieme di procedure di Controllo Logico dei Link (*Logical Link Control*). Fornisce un modo semplice per consentire la coesistenza di protocolli differenti, tuttavia è piuttosto limitato. Novell utilizza un Service Address Point non ufficiale (una sorta di identificativo di protocollo) ma dal momento che anche tutti gli altri lo utilizzano, questo non ha ancora rappresentato un problema.

SNAP

acronimo di *Sub Network Access Protocol*. Questo protocollo è progettato per funzionare su 802.3 e 802.2. Espande le capacità di 802.2 e fornisce una compatibilità con i tipi di frame esistenti Ethernet ed Ethernet_II IPX

IPX

Internet Packet eXchange: protocollo utilizzato da Novell Corporation per fornire un supporto internetworking per i loro prodotti NetWare(tm). IPX ha funzionalità simili al protocollo IP utilizzato in TCP/IP.

Indirizzo di rete IPX

Numero che identifica univocamente una particolare rete IPX. La notazione comune per questo indirizzo è in esadecimale. Un esempio può essere: `0x23a91002`.

Rete interna IPX

Rete IPX virtuale, poiché non corrisponde a una rete fisica. Viene utilizzata per fornire un'identificazione e un indirizzamento univoco per un host IPX. Solitamente, è utile per identificare gli host IPX che esistono in più di una rete fisica IPX come i fileserver. L'indirizzo è codificato nella stessa forma di una rete IPX fisica.

RIP

Routing Information Protocol: protocollo utilizzato per propagare automaticamente gli instradamenti (route) di rete in una rete IPX. È funzionalmente simile al RIP utilizzato in TCP/IP.

NCP

NetWare Core Protocol: protocollo di filesystem di rete (networked filesystem) progettato da Novell Corporation per il loro prodotto NetWare(tm). Funzionalmente equivale a NFS utilizzato in TCP/IP.

SAP

Service Advertisement Protocol: protocollo progettato da Novell Corporation per pubblicizzare i servizi di rete in un ambiente NetWare(tm).

Indirizzo hardware

Numero che identifica in modo univoco un host all'interno di una rete fisica. Ne sono esempio gli Indirizzi Ethernet. Un indirizzo Ethernet viene generalmente codificato con sei valori esadecimali separati da un carattere "due punti" es.: `00:60:8C:C3:3C:0F`.

instradamento (route)

È il percorso seguito dai pacchetti attraverso la rete per raggiungere la propria destinazione.

5 Breve discussione sulla topologia di rete IPX

Questa è una spiegazione estremamente semplice per persone che non conoscono IPX. Reti estese probabilmente violano molte delle regole indicate, In reti IPX complesse si dovrebbe sempre consultare l'amministratore di rete.

La rete IPX si risolve intorno a uno schema di *reti* numerate, diversamente da IP che pone maggior enfasi sulle *interfacce*. Una rete è una collezione di materiale connesso allo stesso segmento LAN e che *usa lo stesso tipo di trama*. Diffenti tipi di trama sullo stesso segmento LAN sono trattati come reti separate.

Ciascuna rete deve allocare un numero che sia univoco nell'intera interconnessione. Normalmente questo compito è assolto dal server Netware(tm), ma può essere facilmente svolto da Linux. Ai client IPX viene dato questo numero dal server all'atto di partire, a loro è solo richiesto di conoscere il corretto tipo di trama.

L'instradamento tra reti è normalmente effettuato ponendo due schede di rete sullo stesso server. Questo server mantiene tramite il protocollo RIP una tabella di instradamento per l'interconnessione. Periodicamente i server si scambiano questa tabella di instradamento. In breve tempo ciascun server 'scopre' la topologia dell'intera rete.

Se si desidera usare solo i servizi di un server Netware esistente, si può usare `ipx_configure` (sezione 7.1) per definire automaticamente le interfacce IPX tramite richieste broadcast cercando un server. Se fallisce, o se si desidera fornire servizi IPX, sarà necessario definire manualmente le interfacce usando `ipx_interface` o `mars_nwe`.

6 I file correlati a IPX nel filesystem `/proc`

All'interno del filesystem `/proc` sono presenti molti file correlati al supporto IPX di Linux:

/proc/net/ipx_interface

Contiene informazioni circa le interfacce IPX configurate sulla macchina. Tali interfacce potrebbero essere state configurate manualmente, oppure rilevate e configurate automaticamente.

/proc/net/ipx_route

Contiene una lista di percorsi esistenti nella tabella di instradamento IPX. Potrebbero essere stati aggiunti manualmente, oppure automaticamente tramite un demone di instradamento IPX.

/proc/net/ipx

Contiene una lista di socket IPX attualmente aperti per un utilizzo sulla macchina.

7 I tool IPX di Greg Page

Greg Page (greg@caldera.com)

della *Caldera Incorporated*, ha scritto molti strumenti per la configurazione IPX e ha fornito miglioramenti per il supporto IPX.

I miglioramenti nel kernel consentono di configurare Linux come un bridge o router pienamente funzionale. Il supporto IPX migliorato è già presente nella distribuzione del kernel pertanto, probabilmente, lo possedete già. I tool di configurazione forniscono la possibilità di impostare i dispositivi di rete per supportare IPX, l'instradamento IPX e altre facilitazioni sotto Linux. I tool di rete IPX per Linux sono disponibili presso: *sunsite.unc.edu, /pub/Linux/system/filesystems/ncpfs/ipx.tgz* .

7.1 I tool IPX in maggior dettaglio

ipx_interface

Viene utilizzato per aggiungere, cancellare o controllare manualmente la capacità di IPX verso un dispositivo di rete esistente. Normalmente si tratta di un dispositivo di rete Ethernet per esempio, `eth0`. Almeno un'interfaccia IPX deve essere definita come *interfaccia primaria*; si ottiene con l'opzione **-p** di questo comando. Ad esempio, per abilitare il dispositivo Ethernet `eth0` come interfaccia primaria IPX utilizzando il tipo di frame IEEE 802.2 e l'indirizzo di rete IPX `39ab0222`, si utilizza il comando:

```
# ipx_interface add -p eth0 802.2 0x39ab0222
```

Se il tipo di trama differisce da quello dei server NetWare(tm) sulla rete, essi di proposito ignoreranno il computer. Se il tipo di trama è corretto ma il numero di rete differisce, essi ignoreranno comunque il computer ma segnaleranno frequentemente il fatto sulla console NetWare del server. L'ultima situazione è certo che suscita rimproveri dal proprio amministratore NetWare e potrebbe disabilitare i client NetWare esistenti.

Se il sistema restituisce un messaggio di errore mentre si esegue questo programma e non è ancora stato configurato TCP/IP, allora sarà necessario avviare manualmente l'interfaccia `eth0` utilizzando il comando:

```
# ifconfig eth0 up
```

ipx_configure

Abilita o disabilita la configurazione automatica delle interfacce e l'impostazione dell'interfaccia primaria.

–auto_interface

consente di impostare se i dispositivi di rete debbano essere configurati automaticamente come dispositivi IPX oppure no.

–auto_primary

consente di impostare se il software IPX debba selezionare automaticamente un'interfaccia primaria oppure no. Si sono riscontrati problemi usando questa opzione con client Windows 95 connessi alla rete.

Un esempio tipico consiste nell'abilitare sia la configurazione automatica delle interfacce che l'impostazione automatica di quella primaria, tramite:

```
# ipx_configure -auto_interface=on -auto_primary=on
```

ipx_internal_net

Consente di configurare o "deconfigurare" un indirizzo di rete interna. Tale indirizzo è opzionale, ma se viene configurato rappresenterà sempre l'interfaccia primaria. Per impostare l'indirizzo di rete IPX `ab000000` sul nodo IPX `1`, si utilizzerà il comando:

```
# ipx_internal_net add 0xab000000 1
```

ipx_route

Consente di modificare manualmente la tabella di instradamento IPX. Ad esempio, per aggiungere un percorso alla rete IPX `39ab0222` tramite un router il cui numero di nodo è `00608CC33C0F` sulla rete IPX `39ab0108`, si può utilizzare il comando:

```
# ipx_route add 0x39ab0222 0x39ab0108 0x00608CC33C0F
```

8 Configurazione della macchina Linux come router IPX

Se ci si trova nella situazione di dover interconnettere diversi segmenti IPX, è necessario utilizzare un router. In ambiente Novell ci sono due tipi di informazioni che devono essere diffuse all'interno della rete. Si tratta di "*network routing information*" propagate utilizzando Novell RIP, e "*service advertisement information*" propagate utilizzando Novell SAP. Affinché un router sia utile nella maggior parte delle situazioni, è necessario che supporti entrambi i protocolli.

Linux fornisce supporto per entrambi i protocolli e si può facilmente fare in modo che funzioni come un router completamente conforme a Novell.

Il supporto IPX per il kernel gestisce il percorso dei pacchetti IPX attraverso le interfacce, secondo le regole codificate all'interno della tabella di instradamento IPX. Linux ha bisogno di un programma per implementare RIP e SAP di Novell per assicurare che la tabella di instradamento IPX sia compilata correttamente e aggiornata periodicamente al fine di riflettere eventuali cambiamenti dello stato della rete.

Volker Lendecke (lendecke@namu01.gwdg.de) ha sviluppato un demone di instradamento per questo scopo: *ipxripd*. Il pacchetto menzionato successivamente *mars_nwe* include un demone di instradamento alternativo.

È possibile trovare `ipxripd` presso *sunsite.unc.edu, /pub/Linux/system/filesystems/ncpfs/ipxripd-0.7.tgz*

o presso il sito privato di Volker *ftp.gwdg.de, /pub/linux/misc/ncpfs/ipxripd-0.7.tgz* .

I passi da seguire per configurare una macchina Linux affinché si comporti come router sono:

1. Compilare il kernel con il supporto **IPX**, **Ethernet** e **/proc**.
2. Ottenere, compilare e installare il programma demone **ipxd**.
3. Avviare il nuovo kernel e assicurarsi che ognuna delle schede Ethernet sia stata correttamente rilevata e che non esistano conflitti di hardware.
4. Abilitare il protocollo IPX su ciascuna delle interfacce utilizzando il comando `ipx_interface` precedentemente descritto (vedere 7.1 (ipx_interface)).
5. Eseguire il programma demone **ipxd**.

Si consideri la semplice rete illustrata di seguito:

```
 IPX Addr: 0x01000000  802.2
|--------------------------|
              |
              \_________________________
                                        \          Linux Router
 IPX Addr: 0x02000000  802.2             \
|--------------------------|              \   eth0/-----------\
              |                            \--====|           |
              \_________________________          | IPX route |
                                        \     eth1|  Table    |
 IPX Addr: 0x03000000  etherII           \----====|    ^      |
|--------------------------|                      |    |      |
              |                               eth2|  IPXd     |
              \______________________________/====|           |
                                                  |  SAPd     |
 IPX Addr: 0x04000000  etherII                eth3|           |
|--------------------------|                 /====|           |
              |                              |    \___________/
              \______________________________/
```

La configurazione per la rete sopra riportata corrisponde a:

```
# ipx_interface add eth0 802.2 0x0100000000
```

```
# ipx_interface add eth1 802.2 0x0200000000
# ipx_interface add eth2 etherii 0x0300000000
# ipx_interface add eth3 etherii 0x0400000000
# ipxd
```

Dopo aver atteso qualche istante, controllando il file `/proc/net/ipx_route`, dovrebbe essere possibile osservare la presenza dei percorsi IPX rilevanti per la propria configurazione e di tutti quelli acquisiti da altri router nella rete.

8.1 È necessario configurare una rete interna?

Novell ha una caratteristica denominata *rete interna*, che viene utilizzata per semplificare l'instradamento nelle situazioni in cui un host è connesso a più dispositivi di rete. Può essere utile in caso di un fileserver connesso a più reti, poiché imposta un solo percorso come indicazione per raggiungere il server, indipendentemente dalla rete su cui si sta lavorando.

In caso di una configurazione senza un fileserver e nella quale la macchina agisca solo da router IPX, la questione non è altrettanto semplice. Si è notato che la configurazione per IPX/PPP funziona "meglio" se viene implementata anche una rete interna.

È un'operazione semplice da fare ma può richiedere la ricompilazione del kernel. È importante rispondere "**Y**", mentre si esegue `make config`, alla richiesta:

```
 ...
 ...
Full internal IPX network (CONFIG_IPX_INTERN) [N/y/?] y
 ...
 ...
```

Per configurare l'interfaccia della rete interna, utilizzare il comando `ipx_internal_net` descritto in precedenza nel paragrafo relativo agli 7.1 (strumenti di rete). La precauzione principale da prendere, consiste nell'assicurarsi che l'indirizzo IPX assegnato sia univoco nella rete e quindi che nessun'altra macchina, o rete, lo stia utilizzando.

9 Configurazione della propria macchina Linux come client NCP

È probabile che gli utenti di reti a tecnologia mista, comprendente protocolli IP e IPX, prima o poi desiderino che la propria macchina Linux possa accedere a dati memorizzati in un file server Novell sulla rete. Novell offre da tempo un pacchetto NFS per i propri file server in grado di consentire questa funzione; tuttavia, nel caso di una piccola installazione, o se solo un ristretto numero di persone è interessato a tale funzionalità, non è facile giustificare il costo del pacchetto commerciale.

Volker Lendecke lendecke@namu01.gwdg.de ha scritto un modulo che consente di montare volumi Novell sul file system Linux senza richiedere alcun prodotto addizionale per il file server. Volker ha denominato il pacchetto **ncpfs** e ha tratto le informazioni necessarie principalmente dal libro "Netzwerkprogrammierung in C" di Manfred Hill e Ralf Zessin (ulteriori dettagli sul libro sono contenuti nel file README del pacchetto **ncpfs**).

Il software consente a Linux di emulare una normale stazione di lavoro Novell per servizi di file. Include anche una piccola utility che consente di stampare le code Novell (questo argomento è documentato nel paragrafo relativo al 11 (Client di Stampa)). Il pacchetto **ncpfs** è in grado di lavorare con file server Novell a partire dalla versione 3.x, mentre non funziona con Novell 2.x. Il client **ncpfs** funziona anche con prodotti strettamente compatibili con Novell, ma sfortunatamente alcuni, pur dichiarandosi compatibili, non lo sono a sufficienza. Per utilizzare **ncpfs** con Novell 4.x, è preferibile configurare il fileserver per operare in modalità di emulazione bindery. Il supporto NDS è una aggiunta piuttosto recente alla versione beta di *ncpfs* e per di più in certi paesi è proibito a causa dell'inclusione di tecnologia protetta da brevetto.

9.1 Come ottenere ncpfs

L'ultimo pacchetto **ncpfs** fu progettato per versioni del kernel fino alla *1.2.13* o superiore alla *1.3.71* (questo include la versione *2.x.x*). Se il kernel in uso non rientra in queste categorie, è necessario eseguire un aggiornamento, operazione descritta in dettaglio nel *Kernel-HOWTO* .

Il pacchetto **ncpfs** può essere ottenuto tramite ftp anonimo con il sito di Volker presso *ftp.gwdg.de in /pub/linux/misc/ncpfs/*

oppure *sunsite.unc.edu in /pub/Linux/system/filesystems/ncpfs*

o siti mirror. La versione attuale nel momento in cui si scrive questo documento è: `ncpfs-2.0.11.tgz` oppure `ncpfs-2.2.0.tgz` che aggiunge il supporto NDS.

9.2 Compilazione di ncpfs per il kernel *1.2.13*

Compilazione di un kernel con supporto Ethernet e IPX

La prima cosa da fare è assicurarsi che il proprio kernel sia stato compilato con l'abilitazione al supporto IPX. Nella versione di kernel *1.2.13* basta assicurarsi di aver risposto "**Y**" alla domanda `The IPX protocol`, come illustrato nel seguito:

```
 ...
 ...
Assume subnets are local (CONFIG_INET_SNARL) [y]
Disable NAGLE algorithm (normally enabled) (CONFIG_TCP_NAGLE_OFF) [n]
The IPX protocol (CONFIG_IPX) [n] y
*
* SCSI support
 ...
 ...
```

Ovviamente è necessario includere il driver relativo per la propria scheda Ethernet. Per maggiori dettagli, si dovrebbe consultare *l'Ethernet-HOWTO* .

A questo punto è possibile procedere alla compilazione del kernel. Al termine potrebbe essere necessario eseguire `lilo` per installarlo.

Esecuzione del comando `tar` per il software ncpfs

```
# cd /usr/src
# tar xvfz ncpfs-2.0.10.tgz
# cd ncpfs
```

Controllo del Makefile

Se si desidera utilizzare `kerneld` per caricare automaticamente il modulo `ncpfs`, è necessario togliere il commento alla riga che si riferisce a `KERNELD` all'interno del `Makefile`. Se non si conosce il significato di questa azione, si dovrebbe leggere *Kernel-HOWTO* per familiarizzare con la configurazione del modulo kernel.

Esecuzione di `make` per il software ncpfs

Il software dovrebbe poter essere compilato correttamente senza che siano necessari ulteriori configurazioni, tramite il comando:

```
# make
```

Copia dei tool IPX, se non ancora presenti

Al termine dell'esecuzione di `make`, tutti gli strumenti necessari dovrebbero trovarsi nella directory **ncpfs/bin**. È possibile utilizzare il comando:

```
# make install
```

per installare i tool in directory scelte da Volker. Se si sta lavorando su un sistema basato su ELF, si dovrà eseguire nuovamente `ldconfig -v` per assicurarsi che venga trovata la libreria condivisa.

Copia del modulo `ncpfs.o` se necessario

Se si sta eseguendo la compilazione di un kernel *1.2.**, al termine dell'esecuzione di make, si troverà un file denominato *ncpfs.o* nella directory `ncpfs/bin`. Si tratta del modulo di kernel `ncpfs`, che dovrebbe essere copiato in luogo utile. Sul sistema **debian** dell'autore, è stato copiato nella directory `/lib/modules/1.2.13/fs` ed `ncpfs` aggiunto al file `/etc/modules` in modo che venga eseguito automaticamente all'avvio del sistema. Per una diversa distribuzione, si dovrebbe scoprire dove sono localizzati i moduli per copiarlo in quella locazione; in ogni altro caso semplicemente copiarlo nella directory `/etc`. Per caricare i moduli manualmente, è necessario utilizzare il comando:

```
# insmod ncpfs.o
```

9.3 Come compilare ncpfs per i kernel *1.3.71++/2.0.**

Per l'ultima versione di **ncpfs**, si dovrebbe utilizzare un kernel *1.3.71* o successivo; questo include i kernel *2.0.**.

Se si ha intenzione di utilizzare un kernel di versione *1.3.71* o successivo, allora il codice del kernel **ncpfs** è stato incluso nella distribuzione standard. È sufficiente rispondere "**Y**" alla domanda:

```
Networking options  --->
...
...
<*> The IPX protocol
...
Filesystems  --->
...
...
<*> NCP filesystem support (to mount NetWare volumes)
...
```

A questo punto è necessario seguire le istruzioni per la compilazione dei kernel *1.2.**, in modo da costruire i tool; tuttavia non ci sarà un file modulo da installare.

9.4 Configurazione e utilizzo di ncpfs

Configurazione del software di rete IPX

Esistono due modi per configurare il software di rete IPX. È possibile configurare manualmente tutte le informazioni di rete IPX, oppure lasciare che il software determini da solo delle impostazioni ragionevoli, tramite il comando:

```
# ipx_configure --auto_interface=on --auto_primary=on
```

Questo comando dovrebbe funzionare nella maggior parte dei casi, ma se ciò non accade, si invita a leggere il paragrafo 7 (Tools di IPX) per configurare il software manualmente. Si sono notati problemi nell'usare questo comando su reti in cui erano presenti client Windows 95.

Collaudo della configurazione

Dopo la configurazione della rete IPX, dovrebbe essere disponibile il comando `slist` che elenca tutti i fileserver Novell presenti:

```
# slist
```

Se il comando slist fornisce un messaggio del tipo: "`ncp_connect: Invalid argument`" significa che il kernel probabilmente non supporta IPX. Controllare che sia stato effettivamente avviato il kernel appropriato. Quando si avvia il sistema, i messaggi di startup dovrebbero contenere informazioni riferite a `IPX` e `ncpfs`. Se, nonostante tutto, il comando `slist` non elenca i file server, allora si dovrebbe utilizzare il metodo manuale per la configurazione della rete.

Montare un volume Novell(R)

Se il software di rete IPX funziona correttamente, si dovrebbe essere in grado di montare un volume file server Novell all'interno del file system Linux. Il comando `ncpmount` viene utilizzato per questo scopo, e richiede la specifica delle seguenti informazioni:

1. Nome del fileserver.
2. Identificativo di login nel fileserver. Se esiste, deve essere specificata anche una password.
3. Il punto di mount, ossia dove si desidera montare la condivisione. Si tratterà di una directory esistente sulla macchina.

Esiste un comando equivalente `ncpumount` per smontare un file system NCP precedentemente montato. I file system NCP saranno smontati in modo corretto se si esegue un normale shutdown della macchina; pertanto non bisogna preoccuparsi di eseguire il comando `npcumount` manualmente prima di arrestare il sistema con `halt` oppure `shutdown`.

Esempio di comando per montare il file server `ACCT_FS01`, con login `guest`, senza password, corrispondente alla directory `/mnt/Accounts`:

```
# ncpmount -S ACCT_FS01 /mnt/Accounts -U guest -n
```

Si noti l'utilizzo dell'opzione `-n` per indicare che il login non richiede alcuna password. Stesso login con password `secret` avrebbe forma:

```
# ncpmount -S ACCT_FS01 /mnt/Accounts -U guest -P secret
```

Se non si specifica né l'opzione `-n`, né `-P`, allora il sistema richiederà una password.

Verifica del mount

Se l'esecuzione è terminata con successo, tutte le unità accessibili all'utente specificato durante il login, saranno elencate come directory sotto il punto di mount. Dovrebbe anche essere possibile percorrere la struttura delle directory per trovare altri file. Alternativamente si può usare l'opzione `-V` per montare un singolo volume.

NCP non fornisce uid (identificativo utente) o gid (identificativo di gruppo) per la proprietà dei file, tutti i file avranno le autorizzazioni e le proprietà assegnate alla directory che rappresenta il punto di mount, condizionate dai permessi concessi dal server Novell. È necessario tenerne conto quando si effettuano condivisioni tra utenti Linux.

Configurazione per l'esecuzione automatica dei mount

Se si ha la necessità di avere un mount NCP permanente, è possibile configurare i comandi precedentemente descritti all'interno di file `rc`, in modo che vengano eseguiti all'avvio del sistema. Nel caso in cui la distribuzione utilizzata non fornisca un metodo per configurare IPX (come Debian), si consiglia di impostare i comandi nel file `/etc/rc.local`, se esiste. È possibile utilizzare comandi del tipo:

```
#
# Avvio del filesystem NCP
/sbin/insmod /lib/modules/1.2.13/fs/ncpfs.o

# Configurazione della rete IPX
ipx_configure --auto_interface=on --auto_primary=on

# login all'Accounting fileserver
ncpmount -S ACCT_FS01 /mnt/Accounts -U guest -n

#
```

Un altro metodo di configurazione consiste nell'elencare nel file `$HOME/.nwclient` i mount NCP temporanei o specifici per l'utente che verranno eseguiti regolarmente. È possibile memorizzare specifiche in modo da impostarle senza doverle indicare ogni volta. Il formato è molto semplice:

```
# Il primo inserimento rappresenta il server "preferenziale" che verrà
# utilizzato ogni volta non sia specificato esplicitamente un server.
#
# Login utente TERRY al fileserver DOCS_FS01 con password "password"
DOCS_FS01/TERRY password
#
# Login utente Guest al fileserver ACCT_FS01 senza password.
ACCT_FS01/GUEST -
```

Per attivare questi mount si può utilizzare il comando:

```
$ ncpmount /home/terry/docs
```

per montare DOCS_FS01 con login TERRY sotto la directory `/home/terry/docs`. Si noti che è stato utilizzato il fileserver DOCS_FS01 perché non ne è stato specificato alcun altro nel comando di mount. Se fosse stato utilizzato il comando:

```
$ ncpmount -S ACCT_FS01 /home/terry/docs
```

allora sarebbe stato montato il fileserver ACCT_FS01 con login GUEST.

Nota: affinché questo meccanismo funzioni, l'autorizzazione per il file `$HOME/.nwclient` deve essere `0600`. Potrebbe essere necessario utilizzare il comando:

```
$ chmod 0600 $HOME/.nwclient
```

Se anche degli utenti non root devono essere autorizzati a utilizzare questo meccanismo, allora il comando `npcmount` deve essere *Set Userid Root*. Ne consegue che potrebbe essere necessario attribuirgli le autorizzazioni:

```
# chmod 4755 ncpmount
```

Collaudo dell'utility `nsend`.

Il pacchetto include anche un utility per inviare messaggi a utenti Novell. Il suo nome è `nsend` e viene utilizzato come di seguito riportato:

```
# nsend rod hello there
```

invia il messaggio “hello there” a un utente con login “rod” sul file server “primario” (ossia il primo che appare nel proprio file `.nwclient`). È possibile specificare un altro file server con la stessa sintassi descritta per il comando `ncpmount`.

10 Configurazione della macchina Linux come server NCP

Sono disponibili due pacchetti che consentono a Linux di fornire le funzionalità di un file server Novell. Entrambi permettono di condividere file sulla propria macchina Linux con utenti che utilizzano il software client NetWare di Novell. Gli utenti possono fare in modo che dei file system appaiano come dischi locali sulle proprie macchine, proprio come farebbero con un reale file server Novell. Utilizzandoli entrambi, è possibile rendersi conto di quale dei due meglio soddisfa le proprie esigenze.

10.1 Il pacchetto mars_nwe

Martin Stover (mstover@freeway.de) ha sviluppato **mars_nwe** per consentire a Linux di fornire ai client NetWare sia servizi di file, che servizi di stampa.

Il nome **mars_nwe** significa Martin Stovers Netware Emulator.

10.1.1 Capacità di mars_nwe

mars_nwe implementa un sottoinsieme del più completo NCP Novell per servizi di file e di bindery basati su dischi o anche di stampa. Probabilmente contiene errori, tuttavia attualmente viene utilizzato da molte persone, pertanto il numero di errori è in continua diminuzione man mano che vengono rilasciate nuove versioni.

10.1.2 Come ottenere mars_nwe

È possibile ottenere **mars_nwe** da *ftp.gwdg.de /pub/linux/misc/ncpfs/*

o da *sunsite.unc.edu /pub/Linux/system/filesystems/ncpfs/* .

La versione nel momento in cui questo documento viene scritto è: `mars_nwe-0.98.pl10.tgz`.

10.1.3 Come compilare il pacchetto mars_nwe

Compilazione di un kernel con supporto Ethernet e IPX

Con un kernel di versione *1.2.13* è sufficiente assicurarsi di aver risposto “**Y**” alla domanda: `The IPX protocol` e “**N**” alla domanda: `Full internal IPX network`, come di seguito descritto:

```
 ...
 ...
The IPX protocol (CONFIG_IPX) [n] y
 ...
 ...
Full internal IPX network (CONFIG_IPX_INTERN) [N/y/?] n
 ...
 ...
```

In kernel più recenti viene adottato un processo simile ma il testo del messaggio potrebbe essere leggermente diverso.

È anche necessario includere il driver appropriato per la propria scheda Ethernet. Maggiori dettagli sono presenti in *Ethernet-HOWTO* .

Dopo di che è possibile procedere con la compilazione del kernel. Ricordarsi, una volta terminata l'operazione, di eseguire `lilo` per installarlo.

Esecuzione del comando tar per il pacchetto mars_nwe

```
# cd /usr/src
# tar xvfz mars_nwe-0.99.pl10.tgz
```

Esecuzione del comando make per mars_nwe

Si tratta di un'operazione molto semplice. Il primo passo consiste nell'eseguire semplicemente `make`, che creerà un file `config.h`. Assicurarsi della presenza del file e modificarlo se necessario. Il file consente di impostare oggetti come, per esempio, le directory di installazione o il numero massimo di sessioni e unità che il server sarà in grado di supportare. Gli inserimenti veramente importanti da guardare sono:

```
FILENAME_NW_INI        locazione del file di inizializzazione
PATHNAME_PROGS         percorso in cui trovare i programmi eseguibili
                       di supporto
PATHNAME_BINDERY       percorso a cui indirizzare i file "bindery"
PATHNAME_PIDFILES      directory in cui scrivere i file "pid"
MAX_CONNECTIONS        numero massimo di connessioni simultanee ammesse
MAX_NW_VOLS            numero massimo di unità che mars_nwe potrà
                       supportare
MAX_FILE_HANDLES_CONN  numero massimo di file aperti per connessione
WITH_NAME_SPACE_CALLS  se si desidera supportare i client ncpfs
INTERNAL_RIP_SAP       se si desidera che mars_nwe fornisca routing
                       rip/sap
SHADOW_PWD             se si utilizzano shadow password.
```

Le impostazioni predefinite dovrebbero andare bene, tuttavia è sempre meglio controllare. Una volta fatto, con i comandi

```
# make
# make install
```

si eseguiranno la compilazione dei server e l'installazione nelle directory appropriate. Verrà anche installato il file di configurazione `/etc/nwserv.conf`.

Configurazione del server

La configurazione è piuttosto semplice: consiste nel modificare il file `/etc/nwserv.conf`. Il formato di questo file a prima vista appare enigmatico, ma in realtà è molto lineare. Il file contiene diverse istruzioni di configurazione composte da una sola riga. Ogni riga è delimitata da spazi bianchi e inizia con un numero che indica l'argomento da impostare. Tutti i caratteri che seguono un "#" sono considerati un commento, e pertanto ignorati. Martin fornisce un esempio nel pacchetto, e di seguito viene riportato un altro semplificato per offrire una possibile alternativa.

```
# VOLUMI (max. 5)
# Solo il volume SYS è obbligatorio. La directory contenente il volume SYS
# deve contenere le directory: LOGIN, PUBLIC, SYSTEM, MAIL.
```

```
# L'opzione "n" ignora le lettere maiuscole/minuscole.
# L'opzione "k" converte tutti i nomi di file all'interno di richieste NCP
# in lettere minuscole.
# L'opzione "m" marca il volume come rimovibile (utile, ad esempio,
# per cdrom).
# L'opzione "r" imposta il volume come a sola lettura.
# L'opzione "o" indica che il volume è un file system montato singolarmente.
# L'opzione "P" consente di utilizzare i comandi come file.
# L'opzione "O" consente l'utilizzo del namespace OS/2.
# L'opzione "N" consente l'utilizzo del namespace NFS.
# È predefinito l'uso delle maiuscole.
# Sintassi:
# 1 <Nome del Volume> <Percorso del Volume>   <Opzioni>

1   SYS     /home/netware/SYS/              # SYS
1   DATA    /home/netware/DATA/   k             # DATA
1   CDROM   /cdrom                kmr       # CDROM

# NOME DEL SERVER
# Se non impostato, l'hostname Linux sarà convertito in lettere maiuscole
# e utilizzato. È opzionale, se non configurato verrà utilizzato
# l'hostname.
# Sintassi:
#       2 <Nome del Server>

2   LINUX_FS01

# INDIRIZZO DI RETE INTERNA
# L'Indirizzo di Rete Interna IPX rappresenta una caratteristica che
# semplifica il routing IPX per host che possiedono porte su più di
# una rete IPX.
# Sintassi:
#       3 <Internal Network Address> [<Node Number>]
# oppure:
#       3 auto
#
# Se si utilizza "auto", verrà utilizzato il proprio indirizzo IP host.
# NOTA: questo potrebbe essere pericoloso, pertanto assicurarsi di
# assegnare alla propria rete un numero univoco.
# Gli indirizzi sono esadecimali a 4byte (è necessario che inizino
# con 0x).

3       0x49a01010  1

# DISPOSITIVI DI RETE
# Questa riga configura la propria rete IPX. Se la propria rete IPX
# è già stata configurata, questo inserimento è superfluo. Corrisponde
# all'utilizzo diipx_configure/ipx_interface prima di attivare il server.
# Sintassi:
#       4 <Numero di Rete IPX > <nome del device> <frametype> [<ticks>]
#                         Frame types: ethernet_ii, 802.2, 802.3, SNAP
4  0x39a01010  eth0  802.3  1


# SALVATAGGIO DEGLI INSTRADAMENTI IPX DOPO LO SPEGNIMENTO DEL SERVER
```

```
# Sintassi:
#    5 <flag>
#       0 = non salvare i route, 1 = salvare i route

5 0

# VERSIONE NETWARE
# Sintassi:
#    6 <versione>
#      0 = 2.15, 1 = 3.11

6 1

# GESTIONE DELLA PASSWORD
# i client Novell DOS reali supportano una caratteristica per criptare
# la password quando viene modificata. È possibile scegliere se si
# desidera che il proprio server supporti o meno questa caratteristica.
# Sintassi:
#       7 <flag>
#       dove <flag> può essere:
#       0 per forzare la codifica della password. (I client non possono
#         modificare la password)
#       1 forza la codifica della password, consente la modifica delle
#         password non criptate.
#       7 consente password non criptate ma non password vuote.
#       8 consente password non criptate, comprese password vuote.
#       9 per password completamente non criptate (non funziona con OS/2).

7 1

# DIRITTI GID e UID MINIMALI
# abilitazioni utilizzate per connessioni senza login. Tali abilitazioni
# saranno utilizzate per i file nella propria connessione di server
# primarie.
# Sintassi:
#       10 <gid>
#       11 <uid>
#       <gid> <uid> provengono da /etc/passwd, /etc/groups

10  200
11  201

# PASSWORD DEL SUPERVISORE
# Può essere rimosso dopo aver avviato il server per la prima volta. Il
# server cripterà questa informazione nel file bindery dopo la sua esecuzione.
# Si dovrebbe evitare di utilizzare l'utente "root", al contrario è
# consigliabile usare un altro account per amministrare il fileserver mars.
#
# Questa riga viene letta e criptata all'interno dei file bindery del server,
# pertanto è necessario che esista la prima volta che si avvia il server
# per assicurarsi che la password non venga scoperta.
#
# Sintassi:
#       12 <Login del Supervisore> <Username Unix> [<password>]
```

```
12  SUPERVISOR  terry  secret

# ACCOUNT UTENTE
# Questo associa login NetWare ad account Unix. Le password sono
# opzionali.
# Sintassi:
#       13 <Login Utente> <Username Unix> [<password>]

13  MARTIN martin
13  TERRY  terry

# CONFIGURAZIONE DI LAZY SYSTEM ADMIN
# Se esiste un grande numero di utenti e non è possibile utilizzare
# mappe di utenti individuali di tipo 13, è possibile mappare
# automaticamente i login a mars_nwe con nomi di utenti Linux.
# TUTTAVIA, attualmente non esiste un modo per fare uso di password
# di login Linux, pertanto, tutti gli utenti configurati in questo modo
# utilizzeranno la password impostata in questo punto. Il consiglio è di
# non farlo, a meno che la sicurezza non sia un optional.
# Sintassi:
#       15 <flag> <password comune>
#   <flag> :    0   - non mappare gli utenti automaticamente
#               1   - mappa automaticamente gli utenti non configurati
#              99   - mappa automaticamente tutti gli utenti.

15  0  duzzenmatta

# CONTROLLO
# se si imposta questo flag  mars_nwe si assicurerà automaticamente
# dell'esistenza di certe directory.
# Sintassi:
#       16 <flag>
#       <flag> può essere 0 per no, o 1 per sì.

16  0

# CODE DI STAMPA
# Questo associa le stampanti NetWare alle stampanti Unix. Le directory
# per le code devono essere create manualmente prima di tentare la stampa.
# Le directory per le code NON sono code lpd.
# Sintassi:
#       21 <nome della coda> <directory della coda> <unix_print_cmd>

21  EPSON  SYS:/PRINT/EPSON lpr -h
21  LASER  SYS:/PRINT/LASER lpr -Plaser


# FLAG DI DEBUG
# Normalmente non sono necessari, ma possono essere utili se si sta
# eseguendo il debug di un problema.
# Sintassi:
#       <debug_item> <debug_flag>
#
#    100 = IPX KERNEL
#    101 = NWSERV
```

```
#    102 = NCPSERV
#    103 = NWCONN
#    104 = start NWCLIENT
#    105 = NWBIND
#    106 = NWROUTED
#                 0 = disabilita il debug, 1 = abilita il debug

100 0
101 0
102 0
103 0
104 0
105 0
106 0

# ESECUZIONE DI NWSERV IN BACKGROUND E UTILIZZO DEL FILE DI LOG
# Sintassi:
#       200     <flag>
#       0 = esegue NWSERV in foreground e non utilizza il file di log
#       1 = esegue NWSERV in background e utilizza il file di log

200  1

# NOME DEL FILE DI LOG
# Sintassi:
#       201     <file di log>

201  /tmp/nw.log

# SCRITTURA SUL FILE DI LOG IN CODA O IN SOVRASCRITTURA
# Sintassi:
#       202     <flag>
#       0 = scrive aggiungendo in coda ad un file di log esistente
#       1 = sovrascrive un file di log esistente

202  1

# TEMPO DI SPEGNIMENTO DEL SERVER
# Questa riga imposta quanto tempo deve passare dallo spegnimento del
# server per lo spegnimento effettivo del server.
# Sintassi:
#       210     <tempo>
#       in secondi (il valore predefinito è 10)

210  10

# INTERVALLO DI ROUTING BROADCAST
# Esprime il tempo in secondi tra broadcast del server
# Sintassi:
#       211     <tempo>
#       in secondi (il default è 60)

211  60

# INTERVALLO DI ROUTING LOGGING
```

```
# Imposta quanti broadcast avvengono prima di effettuare la
# registrazione delle informazioni di rete.
# Sintassi:
#       300     <numero>

300  5

# FILE DI LOG DEL ROUTING
# Imposta il nome del file di registrazione del routing
# Sintassi:
#       301     <nomefile>

301  /tmp/nw.routes

# APPEND/SOVRASCRITTURA DEL FILE DI LOG DEL ROUTING
# Imposta se si desidera scrivere aggiungendo in coda su un file di
# log esistente, oppure se si vuole sovrascriverlo.
# Sintassi:
#       302     <flag>
#       <flag> 0 per append, 1 per creazione/sovrascrittura

302  1

# TEMPO DI GUARDIA
# Imposta il tempo per messaggi di guardia in grado di assicurare che
# la rete sia ancora viva.
# Sintassi:
#       310     <valore>
#       <valore> = 0 - invia sempre dei messaggi
#                < 0 - (-ve) per disabilitare i messaggi
#                > 0 - invia messaggi quando il traffico di rete
#                      scende al di sotto di "n" ticks

310  7

# STATION FILE
# Imposta il nome per i station file che determinano per quali
# macchine il fileserver si comporterà come fileserver primario.
# La sintassi di questo file è descritta nella directory "examples"
# del codice sorgente.
# Sintassi
#       400     <nomefile>


400  /etc/nwserv.stations

# GESTIONE DEL GET NEAREST FILESERVER
# Imposta la gestione delle richieste "Get Nearest Fileserver" SAP.
# Sintassi:
#       401     <flag>
#       <flag> può essere: 0 - disabilita le richieste "Get Nearest
#                              Fileserver".
#                          1 - Il file "stations" contiene le stazioni
#                              da escludere.
#                          2 - Il file "stations" contiene le stazioni
```

```
#                              da includere.

401  2
```

Avviamento del server

Se il server è stato configurato in modo tale da aspettarsi che programmi esterni configurino la rete o forniscano funzionalità di instradamento, allora tali programmi devono essere eseguiti prima di avviare il server. Invece, supponendo che il server sia stato configurato affinché esso stesso configuri le interfacce e fornisca i servizi di instradamento, sarà sufficiente il solo comando:

```
# nwserv
```

Collaudo del server

Per controllare un server bisognerebbe in primo luogo collegarsi da un client NetWare sulla propria rete. Quindi impostare un `CAPTURE` dal client ed eseguire una stampa. Se entrambe queste operazioni hanno successo, allora il server è funzionante.

10.2 Il pacchetto lwared

Ales Dryak (A.Dryak@sh.cvut.cz) ha sviluppato `lwared` per consentire a Linux di agire come fileserver basato su NCP.

Ales ha chiamato il pacchetto `lwared`, abbreviazione per **LinWare Daemon**.

10.2.1 Capacità di lwared

Il server `lwared` è in grado di fornire un sottoinsieme delle funzionalità complete di NCP Novell. Comprende la messaggistica ma non fornisce alcuna funzionalità di stampa. Attualmente non funziona molto bene con client Windows 95 o Windows NT. Il server `lwared` si basa sulla presenza di programmi esterni per costruire e aggiornare le tabelle di instradamento e SAP di IPX. Client che non si comportano correttamente possono causare problemi al server. È importante sottolineare che le funzionalità di traduzione dei filename non sono state incluse.

Il server funziona per shell NETX e VLM NetWare.

10.2.2 Come ottenere lwared

Il pacchetto `lwared` può essere compilato per qualsiasi kernel successivo al *1.2.0*, anche se è consigliata la versione *1.2.13* poiché in tal modo non sono necessari patch per il kernel. Alcune delle funzionalità IPX sono cambiate con kernel di versione *1.3.**: questo significa che sono necessari patch specifici per fare in modo che `lwared` funzioni correttamente. Esistono patch appropriati per i nuovi kernel, pertanto se si deve utilizzare un kernel alpha dovrebbe essere possibile avere `lwared` correttamente funzionante.

Il pacchetto `lwared` può essere ottenuto tramite ftp anonimo da: *klokan.sh.cvut.cz /pub/linux/linware/*

o da: *sunsite.unc.edu /pub/Linux/system/network/daemons*

o siti mirror. La versione al momento in cui questo documento viene scritto è: `lwared-0.95.tar.gz`.

10.2.3 Compilazione di lwared

Esecuzione del comando `tar` per il pacchetto `lwared`

Qualcosa del tipo:

```
# cd /usr/src
# tar xvpfz lwared-0.95.tar.gz
```

Compilazione di un kernel con supporto Ethernet e IPX

Coloro che utilizzano un kernel `alpha 1.3.*`, dovrebbero passare alla versione *1.3.17* o superiore poiché le patch supportate sono state create per questa versione. I kernel *1.3.** più vecchi della versione *1.3.17* richiederanno una installazione manuale delle patch (il file `INSTALL` contenuto nel pacchetto riporta alcune informazioni su come eseguire questa operazione). Per installare le patch su un kernel *1.3.17* si può digitare il comando:

```
# make patch
```

Il passo successivo all'applicazione delle patch, consiste nell'assicurarsi che sia stata eseguita la compilazione del kernel con abilitazione del supporto IPX. In un kernel di versione *1.2.13* è sufficiente aver risposto "**Y**" alla domanda: "`The IPX protocol`", come illustrato nell'esempio che segue:

```
 ...
 ...
Assume subnets are local (CONFIG_INET_SNARL) [y]
Disable NAGLE algorithm (normally enabled) (CONFIG_TCP_NAGLE_OFF) [n]
The IPX protocol (CONFIG_IPX) [n] y
*
* SCSI support
 ...
 ...
```

In kernel più recenti il procedimento è analogo, tuttavia il testo del messaggio potrebbe essere leggermente diverso.

Inoltre, è necessario includere il driver appropriato per la propria scheda Ethernet. Maggiori informazioni in l' *Ethernet-HOWTO* .

A questo punto è possibile procedere con la compilazione del kernel. Non si dimentichi di eseguire `lilo` una volta terminata l'operazione.

Compilazione e installazione di `lwared`

Per compilare `lwared` controllare in primo luogo, e modificare se necessario, il file `server/config.h`. Il file contiene diverse impostazioni che guideranno il comportamento del server. I valori predefiniti dovrebbero essere corretti, tuttavia si dovrebbe controllare che le directory specificate per i file di log corrispondano alle impostazioni del proprio sistema.

```
# make depend
# make
# make install
```

Può accadere che il comando "`make depend`" sembri non trovare il file `float.h` sul sistema, tuttavia dovrebbe funzionare comunque. Può anche accadere che, effettuando la compilazione con *gcc 2.6.3* sia necessario modificare la riga:

```
#include <net/route.h>
```

con

```
#include <net/if_route.h>
```

all'interno di `/lib/ipxkern.c` poiché, talvolta questo file ha un nome diverso.

Il comando "`make install`" tenterà di installare il server e i programmi del demone di routing all'interno della directory `/usr/sbin`, il programma `lwpasswd` in `/usr/bin`, i programmi di utilità IPX in `/sbin` e infine, le pagine di manuale verranno installate nella directory `/usr/man`. Se qualcuna di queste locazioni non fosse disponibile nel sistema, sarà necessario modificare il `Makefile` per impostare le directory destinazione.

10.2.4 Configurazione e utilizzo di `lwared`

Configurazione della rete IPX

La prima cosa da fare consiste nella configurazione delle interfacce Ethernet per supportare le reti IPX supportate dal proprio server. È necessario conoscere gli indirizzi di rete IPX per ciascuno dei propri segmenti LAN, quale dispositivo Ethernet (`eth0`, `eth1` ecc.) si trova su ciascun segmento, quale tipo di trama (`802.3`, `EtherII` ecc.) viene utilizzata da ogni segmento LAN e quale indirizzo di Rete Interna deve essere utilizzato dal proprio server (fondamentale se il server offre servizi a più di un segmento LAN). Esempio di configurazione per un server su due segmenti dissimili con indirizzi di rete IPX `23a91300` e `23a91301` e indirizzo di rete interna `bdefaced`:

```
# ipx_internal_net add BDEFACED 1
# ipx_interface add eth0 802.3 23a91300
# ipx_interface add eth1 etherii 23a91301
```

Avvio dei demoni di instradamento

Il kernel instrada i pacchetti IPX come pure per IP, tuttavia sono richiesti programmi aggiuntivi per gestire l'aggiornamento delle tabelle. Nel caso di IPX, sono necessari due demoni, entrambi supportati da `lwared`: `ipxripd` gestisce le informazioni di instradamento e `ipxsapd` le informazioni di SAP. Per avviare i demoni è sufficiente specificare la locazione in cui devono scrivere i propri messaggi di log:

```
# ipxripd /var/adm/ipxrip
# ipxsapd /var/adm/ipxsap
```

Configurazione del server lwared

È necessario eseguire la configurazione manuale di due file per consentire il login al server `lwared`. Tali file sono:

/etc/lwpasswd

In questo file vengono memorizzate le informazioni di account dell'utente. Il programma `lwpasswd` è utilizzato per mantenerlo aggiornato. La sua forma più semplice del file `/etc/lwpasswd` è:

```
ales:
terryd:
guest:
```

Si tratta di una semplice lista di identificativi di login, seguiti dal carattere "`:`" e dalla versione criptata della password. Due importanti avvertimenti sono: l'assenza di password criptate significa che non ne esiste alcuna, gli utenti LinWare devono avere un account per Linux quindi ogni utente presente nel file `/etc/lwpasswd` deve apparire anche in `/etc/passwd`; l'utente `root` è l'unico in grado di modificare le password di un altro utente LinWare come di seguito indicato:

```
# lwpasswd rodg
Changing password for RODG
Enter new password:
Re-type new password:
Password changed.
```

/etc/lwvtab

Si tratta delle tabelle relative al volume LinWare e in esse sono memorizzate le informazioni su directory da rendere disponibili agli utenti LinWare (la natura del file è simile a quella di `/etc/exports` di NFS). Un semplice esempio del suo formato è:

```
SYS             /lwfs/sys
DATA            /lwfs/data
HOME            /home
```

Il nome del Volume è seguito da spazi bianchi seguiti dalle directory Linux da esportare. Ci deve essere almeno una riga relativa al **volume SYS** affinché un server possa avviarsi. Se si desidera che i propri utenti DOS siano in grado di utilizzare il proprio server LinWare come server primario, allora deve essere installata una struttura standard per il volume **SYS** sotto la directory che viene esportato come volume **SYS**. Dal momento che il copyright di questi file appartiene a Novell Corporation, è necessario possedere una licenza per il loro utilizzo. Se gli utenti utilizzeranno un file server Novell come server primario, allora la licenza non è necessaria.

Avvio del server lwared

`lwared` viene avviato tramite il comando:

```
# lwared
```

Se il server è stato avviato come mostrato, il nome del server LinWare dipenderà dalla risposta dell'**hostname** Linux. Se si desidera un nome diverso, è possibile indicarlo esplicitamente. Ad esempio, il comando:

```
# lwared -nlinux00
```

avvia il server con nome **linux00**.

Collaudo del server `lwared`

La prima cosa da controllare è la presenza del proprio *server LinWare* nell'elenco fornito dal comando `slist` (eseguito su un client DOS della propria rete). Il programma `slist` è presente nel volume **SYS** di un fileserver Novell, pertanto può essere eseguire solo da una macchina già collegata. Se l'esito è negativo, controllare che `ipxsapd` e `lwared` siano entrambi in esecuzione. Al contrario, se l'esito è positivo, si tenti la connessione al server e il mount di un volume, tramite:

```
C:> attach linux00/ales
 ...
 ...
C:> map l:=linux00/data:
C:> l:
```

A questo punto dovrebbe essere possibile lavorare con il nuovo disco proprio come con qualsiasi altro disco. Le abilitazioni ai file corrisponderanno a quelle dell'account a Linux parallelo al proprio login LinWare.

11 Configurazione della macchina Linux come client di stampa Novell

Il pacchetto `npcfs` contiene due piccoli programmi che consentono di indirizzare una stampa dalla propria macchina Linux verso una stampante collegata a un server di stampa Novell. Il comando `nprint` permette spedire un file a una coda di stampa Netware. Il comando `pqlist` fornisce la lista delle code di stampa disponibili.

Per ottenere e installare questi comandi, è sufficiente seguire le istruzioni relative al client NCP descritte in precedenza.

Entrambi i comandi richiedono che siano forniti il proprio username e password, qundi per facilitare le operazioni, a volte si preferisce realizzare qualche shell script. Un esempio per i comandi potrebbe essere:

```
# pqlist -S ACCT_FS01 -U guest -n
# nprint -S ACCT_FS01 -q LASER -U guest -n filename.txt
```

La sintassi per il login è simile a quella del comando *ncpmount*. Negli esempi sopra riportati si assume che il fileserver `ACCT_FS01` abbia un account "**guest**" senza password, che esista una coda di stampa di nome `LASER` e che l'utente "**guest**" sia in grado di eseguire stampe su di essa.

Nella macchina Linux dell'autore c'è un corto shell script per ciascuna stampante Novell. Questo può essere usato come filtro di stampa per consentire la stampa usando lo spooler standard di Linux.

12 Configurazione di una macchina Linux come print server Novell

Il pacchetto `npcfs` contiene un programma che consente alla macchina Linux di comportarsi come un server di stampa su una rete NetWare. Per sapere come ottenere ed eseguire la compilazione di tale programma, seguire le istruzioni contenute nel paragrafo relativo al 11 (Client NetWare). Alternativamente, il supporto è incluso nel pacchetto *mars_ nwe*.

12.1 Prerequisiti

La configurazione è piuttosto semplice ma si basa sul fatto che la configurazione di stampa sia già stata completata e sia funzionante sotto Linux. Questo argomento è trattato più in dettaglio all'interno del *Printing-HOWTO*

12.2 Configurazione

Quando si ha a disposizione una configurazione di stampa funzionante, e l'utility `pserver` è stata compilata e installata, è necessario aggiungere dei comandi per avviarla all'interno dei propri *file rc*.

Il comando da utilizzare dipende da come si desidera che funzioni, tuttavia nella sua forma più semplice il comando sarà del tipo:

```
# pserver -S ACCT_01 -U LASER -P secret -q LASERJET
```

Questo comando richiede all'utility `pserver` di effettuare un login al file server `ACCT_01` con username "**LASER**" e password "**secret**" e di prelevare dei job dalla coda di stampa `LASERJET`. Quando viene ricevuto un job di stampa, il comando predefinito `lpr` lo invierà al demone di stampa Linux. La coda di stampa deve già essere definita sul fileserver e lo username deve avere privilegi per la coda.

Alternativamente, sarebbe stato possibile utilizzare qualsiasi comando Linux per accettare e stampare il job. L'opzione `-c` consente di specificare il corretto comando di stampa. Ad esempio, il comando:

```
# pserver -S ACCT_01 -U LASER -P secret -q LASERJET -c "lpr -Plaserjet"
```

avrebbe il medesimo risultato dell'esempio precedente, ma il job verrebbe inviato alla configurazione **laserjet** del `printcap` in luogo di quella predefinita.

13 Panoramica sui comandi utente e di amministrazione di ncpfs

Le versioni più recenti del pacchetto `ncpfs` di Volker includono diversi comandi utente e di amministrazione. I tool sono installati come parte del processo di installazione di `ncpfs`, pertanto se non fossero disponibili, si consiglia di seguire le istruzioni contenute nel paragrafo relativo ai "11 (Client Novell)" per eseguire compilazione ed installazione.

Informazioni dettagliate sono presenti all'interno del manuale; tuttavia, nel seguito viene riportata una breve descrizione dei comandi.

13.1 Comandi utente

ncopy

Network Copy – copia file in modo efficiente tramite l'utilizzo di una funzione NetWare in luogo di eseguire una copia attraverso la rete.

nprint

Network Print – stampa un file indirizzato a una coda di stampa Netware di un server Netware.

nsend

Network Send – invia messaggi ad altri utenti su un server Netware.

nwbols

List Bindery Objects – elenca informazioni di bindery di un server Netware.

nwboprops

List Properties of a Bindery Object – elenca le proprietà di un oggetto bindery Netware.

nwbpset

Set Bindery Property – imposta le proprietà di un oggetto bindery Netware.

nwbpvalues

Print Netware Bindery Objects Property Contents – stampa il contenuto di un oggetto bindery Netware.

nwfsinfo

Fileserver Information – stampa le informazioni di base di un server Netware.

nwpasswd

Netware Password – modifica la password di utenti Netware.

nwrights

Netware Rights – visualizza i diritti associati a un determinato file o directory.

nwuserlist

Userlist – fornisce l'elenco di utenti collegati a un file server Netware.

pqlist

Print Queue List – visualizza il contenuto di una coda di rete Netware.

slist

Server List – visualizza un elenco di file server Netware conosciuti.

13.2 Strumenti per la gestione

nwbocreate

Create a Bindery Object – crea un oggetto bindery Netware.

nwborm

Remove Bindery Object – cancella un oggetto bindary Netware.

nwbpadd

Add Bindery Property – imposta il valore di una proprietà esistente di un oggetto bindary Netware.

nwbpcreate

Create Bindery Property – crea una nuova proprietà per un oggetto bindary Netware esistente.

nwbprm

Remove Bindery Property – elimina una proprietà di un oggetto bindary Netware.

nwgrant

Grant Trustee Rights – assegna diritti di amministrazione a una directory su un file server Netware.

nwrevoke

Revoke Trustee Rights – rimuove diritti di amministrazione da una directory su un fileserver Netware.

14 Configurazione di PPP per reti con supporto IPX

Le nuove versioni per il demone PPP `pppd` di Linux consentono il trasporto dei pacchetti IPX attraverso un collegamento seriale **PPP**. È necessaria almeno la versione `ppp-2.2.0d` del demone. Si faccia riferimento al *PPP-HOWTO* per informazioni dettagliate su dove trovarlo. Quando si compila `pppd` è necessario abilitare il supporto IPX aggiungendo le due righe seguenti:

```
IPX_CHANGE = 1
USE_MS_DNS = 1
```

al file: `/usr/src/linux/pppd-2.2.0f/pppd/Makefile.linux`.

IPX_CHANGE

configura il supporto IPX all'interno di PPP.

USE_MS_DNS

abilita le macchine Microsoft Windows 95 a eseguire Name Lookups.

L'unica particolarità per farlo funzionare consiste nella configurazione.

Esistono diversi modi per farlo, tuttavia ne verranno descritti solo due, entrambi non ancora verificati. Pertanto, si consideri questo paragrafo come sperimentale, e chiunque ottenga dei risultati positivi è invitato a comunicarlo.

14.1 Configurazione di un server IPX/PPP

Innanzi tutto, è necessario configurare la propria macchina Linux come server IP/PPP. Si tratta di un'operazione tutt'altro che difficile. Ancora una volta, si invita a seguire le istruzioni contenute in *PPP-HOWTO* e si otterranno i risultati desiderati. Dopo di che sono sufficienti modifiche minime per fare in modo che IPX funzioni con la stessa configurazione.

14.1.1 Primi passi

Uno dei primi passi consiste nella configurazione della propria macchina Linux come router IPX come descritto in un paragrafo precedente. Non sarà necessario utilizzare il comando `ipx_route` per l'interfaccia `ppp` poiché `pppd` stesso effettuerà la configurazione come viene fatto per IP. Nel momento in cui il demone `ipxd` entra in funzione, rileverà automaticamente ogni nuova interfaccia IPX e propagherà instradamenti per ognuna di esse. In tal modo, i propri ***host dialup*** potranno essere automaticamente visti da altre macchine al momento della loro connessione.

14.1.2 Progettazione

Quando la macchina lavora come server, sarà propria responsabilità assegnare indirizzi di rete a ciascuna delle connessioni PPP nel momento in cui vengono stabiliti. È molto importante sapere che ogni connessione PPP sarà una rete IPX e avrà indirizzo di rete IPX univoco. Questo significa che è necessario decidere come saranno allocati gli indirizzi e la loro corrispondenza. Una semplice convenzione consiste nell'allocare un indirizzo di rete IPX a ogni dispositivo seriale che supporterà IPX/PPP. È possibile allocare gli indirizzi di rete IPX in relazione all'identificativo di login dell'utente connesso, tuttavia in genere non esistono particolari motivi per farlo.

Nel seguito si suppone che questo sia stato fatto, e che vengano usati due dispositivi seriali (modem). Gli indirizzi assegnati in questo esempio sono:

```
device IPX Network Address
------ -------------------
ttyS0  0xABCDEF00
ttyS1  0xABCDEF01
```

14.1.3 Configurazione di pppd per il server

Si esegua la configurazione del file `/etc/ppp/options.ttyS0` nel modo seguente:

```
ipx-network 0xABCDEF00
ipx-node 2:0
ipxcp-accept-remote
```

e quella del file `/etc/ppp/options.ttyS1`:

```
ipx-network 0xABCDEF01
ipx-node 3:0
ipxcp-accept-remote
```

Questi comandi richiederanno a `pppd` di allocare alla connessione, nel momento in cui viene stabilita, gli indirizzi di rete IPX appropriati, di impostare il numero di nodo locale a `2` o `3` e lascerà che il nodo remoto sovrascriva il proprio numero di nodo con quello che esso stesso conosce. Si noti che ognuno degli indirizzi è composto da cifre esadecimali e che `0x` è necessario all'inizio dell'indirizzo di rete, ma non all'inizio dell'indirizzo del nodo. Esistono altri metodi per configurare le stesse informazioni. Disponendo di un solo modem, è sufficiente modificare il file `/etc/ppp/options`. In alternativa, queste informazioni possono essere passate a `pppd` tramite righe di comando.

14.1.4 Verifica della configurazione del server

Per verificare la configurazione è necessario avere a disposizione un client del cui funzionamento si sia certi. Quando il chiamante compone il numero, si collega e `pppd` viene avviato, esso assegnerà l'indirizzo di rete, comunicherà al client il numero di nodo del server e negozierà il numero di nodo del client. Una volta completata questa operazione, e dopo che `ipxd` abbia rilevato la nuova interfaccia, il client dovrebbe essere in grado di stabilire le connessioni con gli host remoti.

14.2 Configurazione di un client IPX/PPP

Durante la configurazione di un client, il fatto di configurare o meno la propria macchina Linux come router IPX dipende dall'avere o meno una LAN locale che si desidera funzioni come un router IPX. Se si sta collegando una singola macchina (standalone) a un server IPX, non è necessario eseguire `ipxd` come descritto. Una configurazione di questo tipo è molto più semplice poiché non devono essere configurati diversi dispositivi seriali.

14.2.1 Configurazione di pppd per il client

La configurazione più semplice è quella che consente al server di fornire tutte le informazioni circa la configurazione della rete IPX. Tale configurazione sarà compatibile con quella del server sopra descritta. È necessario aggiungere alcune opzioni al proprio file `/etc/ppp/options`:

```
ipxcp-accept-network
ipxcp-accept-remote
ipxcp-accept-local
```

Queste opzioni richiedono a `pppd` di comportarsi in modo completamente passivo e di accettare tutti i dettagli di configurazione dal server. Potrebbero essere forniti dei valori di default per i server che non forniscono dettagli aggiungendo le righe relative a `ipx-network` e `ipx-node` in modo analogo a quanto descritto per la configurazione del server.

14.2.2 Verifica del client IPX/PPP

Per verificare il client è necessario collegarsi a un server del cui funzionamento si sia certi. Dopo aver attivato il collegamento e dopo che `pppd` sia stato avviato, dovrebbe essere possibile vedere i dettagli IPX configurati sul proprio dispositivo `ppp0` se si esegue il comando `ifconfig`. Allo stesso modo, dovrebbe essere possibile utilizzare `ncpmount`.

Non è certa la necessità di aggiungere manualmente gli instradamenti IPX in modo da poter raggiungere fileserver distanti. Se qualche lettore è in grado di dare dei suggerimenti, sarebbero estremamente graditi.

15 Il tunnel IPX su IP

Molto frequentemente si verifica la situazione in cui due Reti Locali (LAN) Novell sono in collegamento tramite un'unica connessione IP. In questo caso ad esempio, com'è possibile, giocare a DOOM per DOS in più giocatori? *Andreas Godzina (ag@agsc.han.de)* è in grado di fornire una risposta che consiste in `ipxtunnel`.

`ipxtunnel` incapsula pacchetti IPX con datagrammi `TCP/IP`, in modo che possano essere trasportati con una connessione `TCP/IP`. `ipxtunnel` resta in ascolto e quando rileva un pacchetto IPX, lo ingloba in un datagramma `TCP/IP` e lo instrada verso l'indirizzo IP remoto specificato. Affinché questo i procedimento funzioni, la macchina verso cui viene instradato l'IPX incapsulato deve necessariamente avere in esecuzione un copia della stessa versione di `ipxtunnel`.

15.1 Come ottenere ipxtunnel

`ipxtunnel` può essere ottenuto da *sunsite.unc.edu /pub/Linux/system/network/daemons*

o siti mirror.

15.2 Compilare ipxtunnel

È possibile eseguire la compilazione di ipxtunnel tramite i comandi:

```
# cd /usr/src
# tar xvfz .../ipxtunnel.tgz
# cd ipxtunnel
# make
```

15.3 Configurazione di ipxtunnel

La configurazione di `ipxtunnel` è molto semplice. Si supponga che la macchina di un amico sia *gau.somewhere.com* e che la propria macchina sia denominata *gim.sw.edu*. `ipxtunnel` utilizza il file di configurazione `/etc/ipxtunnel.conf`, che consente di specificare la porta UDP predefinita da utilizzare per la connessione `TCP/IP` alla quale verranno inviati i dati incapsulati, e su quali delle proprie interfacce locali `ipxtunnel` dove stare in ascolto e consegnare pacchetti IPX.

Una configurazione molto semplice è riportata nell'esempio che segue:

```
#
# /etc/ipxtunnel.conf per gim.sw.edu
#
# Porta UDP da utilizzare: (valore predefinito 7666)
port 7777
#
# Macchina remota a cui inviare i pacchetti IPX:
# (nessun valore predefinito)
remote gau.somewhere.com
#
# Interfacce locali su cui restare in ascolto di IPX:
# (valore predefinito eth0)
interface eth0
interface eth1
```

Ovviamente, l'altra macchina possiederà un file di configurazione simile che definisce questa macchina come *host remoto*.

15.4 Verifica e utilizzo di `ipxtunnel`

`ipxtunnel` si comporta *come* un ponte (bridge) IPX, quindi le reti IPX che si trovano a entrambi gli estremi della connessione dovrebbero essere parti della stessa. Andreas non ha mai verificato il funzionamento di `ipxtunnel` in un ambiente davvero in grado di supportare servizi di file Novell, pertanto se qualche lettore dovesse realizzarlo è pregato di far sapere ad Andreas gli esiti della prova.

Se `ipxtunnel` funziona, è possibile l'avvio delle macchine DOOM a entrambi gli estremi della connessione in modalità IPX. Le due macchine dovrebbero essere in grado di vedersi. Andreas ha utilizzato questo codice solo su linee a buona velocità e quindi non è in grado di fare affermazioni sulle prestazioni con connessioni a bassa velocità. Ancora una volta, si invita a comunicare successi o insuccessi.

16 Supporto commerciale IPX per Linux

16.1 Network Desktop di Caldera

La società *Caldera Inc.* produce una distribuzione di Linux che caratterizza un insieme di miglioramenti supportati commercialmente, incluso il pieno supporto funzionale ai client NetWare Novell. La distribuzione di base è la ben nota *Red Hat Linux Distribution* e Caldera vi ha aggiunto il proprio prodotto "Network Desktop". Il supporto NetWare fornisce un client con piene caratteristiche NetWare Novell costruito sulla tecnologia della Novell Corporation. Il client fornisce pieno accesso ai `fileserver 3.x` e `4.x` e include caratteristiche come NetWare Directory Service (NDS) e crittografia RSA.

È possibile ottenere ulteriori informazioni e ordinare dei dettagli da: *Caldera Inc Web Server (http://www.caldera.com/)* .

Se si lavora in un ambiente Netware 4.x e/o NDS, il Caldera Netware Client rappresenta l'unica soluzione disponibile.

Se un'applicazione "industriale" è critica per il supporto Novell, allora sarebbe opportuno dare un'occhiata al prodotto della Caldera.

17 Alcune domande frequenti (FAQ)

Dove si può trovare il software IPX per Linux supportato commercialmente?

La Caldera Corporation offre un client con licenza e con pieno supporto Netware 3.x e 4.x. È possibile ottenere informazioni a riguardo dal *Caldera Inc Web Server (http://www.caldera.com/)* .

Il software IPX lavora con Arcnet/Token Ring/ ecc...?

Il software IPX è in grado di lavorare con le interfacce Arcnet e Token Ring. Non si hanno ancora notizie di tentativi con AX.25. La configurazione è la stessa che per ethernet, tranne la necessità di sostituire i nomi di dispositivi appropriati in luogo di "eth0" e gli appositi indirizzi hardware dove necessario.

Come si configura più di un'interfaccia?

Se la propria macchina contiene più di un'interfaccia, non si dovrebbe usare la configurazione "plug'n'play", ma si dovrebbe utilizzare il comando `ipx_interface` per configurare ciascuna di esse manualmente.

Come scegliere gli indirizzi IPX?

Il networking IPX è simile, ma non identico, al networking IP. La differenza principale consiste nel modo di utilizzo degli indirizzi. IPX non utilizza il concetto di "*subnetworking*", pertanto l'ordinamento delle associazioni esistenti tra indirizzi di rete e reti è differente. Le regole sono molto semplici:

- Ogni indirizzo di rete IPX deve essere univoco su una rete geografica. Questo include gli Indirizzi di Rete Interna. Molte organizzazioni che utilizzano IPX su una rete geografica avranno una sorta di indirizzamento standard che è utile seguire.
- Ogni indirizzo di host su una rete individuale deve essere univoco. Questo significa che ogni host su ciascuna rete IPX deve avere un indirizzo assegnato in modo univoco. Nel caso di una rete ethernet è molto semplice, dal momento che le schede stesse hanno un indirizzo univoco. Nel caso di IPX/PPP bisogna assicurarsi di allocare indirizzi univoci a tutti gli host della rete, indipendentemente dall'estremo delle connessioni ai quali sono collegati. Non è necessario che gli indirizzi degli host siano univoci lungo una rete geografica, dal momento che l'indirizzo di rete viene utilizzato insieme all'indirizzo dell'host per identificarlo in modo univoco.

Quali sono i tipi di trama da utilizzare?

Esistono molti "frame type" sui quali è possibile eseguire IPX. I più comuni sono descritti nel paragrafo 4 (termini comuni) di questo documento (alla voce: 4 (tipo di frame)). Se si sta installando la propria macchina su una rete esistente, si dovrebbero utilizzare quelli che sono già in uso per avere la possibilità di interagire con gli altri host della rete, ma se l'installazione avviene su una rete completamente nuova e si desidera gestire traffico sia IPX, sia IP, allora deve essere utilizzato il frame type `Ethernet_II`.

Le macchine Windows95 possono interferire con l'autorilevamento del tipo di frame?

Apparentemente sembra accadere. Si suggerisce di utilizzare la configurazione manuale (e si tratta comunque del metodo migliore) del tipo di frame in luogo di quella automatica.

Perché appare il messaggio '*invalid argument*' durante la configurazione di IPX?

Probabilmente non è in esecuzione un kernel in grado di supportare IPX. Le possibili soluzioni consistono nel ricompilare il proprio kernel in modo da abilitare il supporto, oppure controllare nuovamente di aver utilizzato lilo per installare e avviare il nuovo kernel.

Perché appare il messaggio '*package not installed*' durante la configurazione di IPX?

Probabilmente non è in esecuzione un kernel in grado di supportare IPX. Le possibili soluzioni consistono nel ricompilare il proprio kernel in modo da abilitare il supporto, oppure controllare nuovamente di aver utilizzato lilo per installare e avviare il nuovo kernel.

Perché appare il messaggio '*IPX support not in kernel*' da pppd?

Probabilmente IPX è stato compilato come modulo senza assicurarsi di averlo caricato prima di avviare `pppd`.

Come eseguire l'esportazione NFS di un filesystem NCP montato?

L'utilizzo di NFS per esportare un filesystem NCP richiede che il suo collegamento sia stato eseguito utilizzando l'opzione `ncpmount -V`. Questa opzione consente di montare un solo volume di un fileserver anziché di tutti i volumi contemporaneamente. In questo modo, il proprio demone NFS consentirà di esportare tale file system con le solite modalità.

Perché il comando '`slist`' non funziona se la rete interna contiene `mars_nwe`?

È necessario abilitare "*get nearest server*". Ossia, la riga `401` nel file `/etc/nwserv.conf` dovrebbe essere `0` a meno che non esistano particolari motivi per non voler rispondere a "*get nearest servers*". Se si desidera solamente che il comando "`slist`" funzioni senza rispondere a ogni richiesta "*get nearest server*", includere i propri numeri di rete interna e di nodo nel file `/etc/nwserv.stations`, quindi impostare la riga `401` nel file `/etc/nwserv.conf` con il valore `2`.

Il pacchetto ncpfs funziona con mars_nwe?

Il codice di Martin e Volker sta lentamente iniziando a convergere. Versioni recenti di `mars_nwe` contengono un'opzione che consente il funzionamento con `ncpfs`. È necessario abilitare `WITH_NAME_SPACE_CALLS` nel file `mars_nwe config.h`.

Esiste software per MSDOS gratuito funzionante con mars_nwe?

Martin possiede un pacchetto distribuito insieme a `mars_nwe` che offre supporto client DOS gratuito per il server `mars_nwe`. È disponibile nello stesso sito del server, con nome *mars_ dosutils-0.01.tgz*. Include codice sorgente in C, compilabile con Borland(tm) C, per programmi come `slist.exe`, `login.exe`, `map.exe` ecc.

18 Copyright

The IPX-HOWTO, a guide to software supporting the IPX protocol for Linux. Copyright (c) 1995 Terry Dawson.

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the:

Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

> L'unica licenza valida è quella originale in lingua inglese. Di seguito ne trovate una traduzione abbastanza fedele che però non ha alcun valore legale.

IPX-HOWTO (*traduzione italiana* `http://www.pluto.linux.it/ildp/HOWTO/IPX-HOWTO.html`), guida al supporto software del protocollo IPX per Linux. Copyright (c) 1995 Terry Dawson.

Questo HOWTO è gratuito; può essere distribuito e/o modificato sotto i termini della `GNU General Public License` come pubblicato dalla `Free Software Foundation`; sia la versione 2 della Licenza, sia ogni altra versione successiva.

Questa documentazione viene distribuita con la speranza che possa essere utile, ma *SENZA ALCUNA GARANZIA*, senza neanche la garanzia implicita di *COMMERCIABILITA'* o *IDONEITA' PER UNO SCOPO PARTICOLARE*. Si faccia riferimento alla `GNU General Public License` per ulteriori dettagli.

Copia della `GNU General Public License` dovrebbe essere allegata a questa documentazione; se così non fosse, scrivere a:

```
Free Software Foundation, Inc., 675 Mass Ave, Cambridge,
MA 02139, USA.
```

19 Varie e ringraziamenti

Terry Dawson <`terry@perf.no.itg.telstra.com.au`> per il documento originale.

David E. Storey (dave@tamos.gmu.edu) e *Volker Lendecke (lendecke@namu01.gwdg.de)*

entrambi hanno dato una grande assistenza fornendomi informazioni per questo documento. *Gilbert Callaghan (gilbert@pokey.inviso.com)* , *David Higgins (dave@infra.com)* e *Chad Robinson (chadr@brtgate.brttech.com)* hanno contribuito alle informazioni sulla configurazione di IPX/PPP. *Bennie Venter (bjv@Gil-galad.paradigm-sa.com)*

ha contribuito fornendo utili informazioni relative ai tipi di frame. *Christopher Wall (vergil@idir.net)* ha fornito utili suggerimenti per migliorare la leggibilità e il layout del documento. Anche *Axel Boldt (boldt@math.ucsb.edu)* ha contribuito con utili suggerimenti. *Erik D. Olson (eriko@wrq.com)*

ha fornito utili informazioni sulla configurazione di PPP per IPX. *Brian King (root@brian.library.dal.ca)*

ha contribuito al paragrafo delle domande frequenti.

“NetWare” è un marchio registrato (registered trademark) di *Novell Corporation* .

“Caldera” è un marchio registrato (registered trademark) di *Caldera Corporation* .

Saluti `Kevin Thorpe.`

`<kevin@pricetrak.com>`