Capitolo 1

Condivisione delle risorse in un workgroup

Chiunque abbia seguito il settore informatico negli ultimi anni avrà notato come la curva delle vendite dei personal computer si sia progressivamente appiattita. Non vi è più una forte vendita di unità ma una tendenza più pacata e lineare. Il fenomeno è dato dal fatto che non vi sono più nicchie significative da conquistare in questo settore. La corsa alla modernizzazione informatica da parte delle imprese si è avuta nel corso degli anni Novanta e oggi il mercato è retto dalla vendita di sistemi a sostituzione di macchine obsolete oppure da adeguamenti dovuti all'incremento del personale o delle necessità produttive.
La tendenza non è omogenea in tutti i segmenti: il settore dei portatili, per esempio, ha avuto un notevole incremento, come pure quello dei server.
I sistemi centrali sono sempre più venduti e il pubblico degli acquirenti si è allargato andando ad abbracciare anche le piccole e medie imprese. I produttori hanno subito colto questa tendenza proponendo server dedicati alle fascie più basse del mercato con dotazioni e costi adeguati.
Basta osservare i listini per appurare l'enorme varietà di prodotti che si possono acquistare oggi.
Nonostante tanta diversità, il compito che tutte queste macchine sono chiamate a svolgere è quasi sempre il medesimo: condividere file. Gli applicativi continuano infatti a girare sui singoli PC e non si è assistito al ritorno dei terminali "stupidi" come sembrava imminente nel 2000.
Un buon numero di questi server funzionano utilizzando le versioni di Windows. Microsoft, infatti, propone da anni protocolli di rete ad alto livello che permettono di condividere facilmente file e stampanti.

Reti in ambito Windows

Il modo più semplice per realizzare una LAN Windows consiste nella messa in opera di un workgroup. Si tratta di un modello in cui ogni macchina ha lo

stesso peso di tutte le altre presenti in rete. Non esiste la distinzione tra server e client in quanto ogni computer può essere al contempo server e client. Quando viene condivisa una cartella, il computer agisce come server ma l'utente presente su tale sistema può accedere a directory e stampanti di altri computer diventando così client per le operazioni di rete.
Come per ogni realtà connessa, gli accessi alle risorse sono regolati da user id e da password, ma in un workgroup ancora una volta non esiste un sistema centralizzato che gestisce gli accessi. Ogni computer possiede un proprio elenco di user id e di password e ogni risorsa condivisa può essere utilizzata da uno o più di questi account. È l'utente che opera su quella data macchina che stabilisce chi può accedere alle condivisioni locali compilando un elenco di utenti autorizzati tramite lo snap-in *Gestione computer* di Windows XP*,* in *Pannello di controllo/Strumenti di amministrazione*.
L'assegnazione degli utenti alle risorse avviene invece selezionando una directory, premendo il tasto destro del mouse, selezionando *Proprietà*, attivando la scheda *Condivisione* e infine attivando *Condividi la cartella in rete*.
All'interno delle piccole realtà è comune che non venga applicato alcun tipo di accesso regolamentato tramite password e che le risorse siano semplicemente visibili a tutti per motivi pratici. È altresì comune che le cartelle condivise siano concentrate su un unico computer per un semplice fattore di comodità. In questo modo è più semplice reperire il materiale di lavoro, più facile organizzare i dati e il backup risulta maggiormente efficace. Si viene così a simulare un piccolo server.
Il principio non è sbagliato nei termini generali ma comporta un grave problema. Non è infatti positivo che uno dei beni più importanti di una azienda, l'archivio dei file comuni, sia presente in un sistema che è prima di tutto una postazione di lavoro.
Chiunque abbia un po' di dimestichezza con i sistemi informatici di largo consumo sa quanto è semplice creare danni a una postazione per incuria o scarse conoscenze e quanto sia semplice contrarre virus. È anche facile che per errore o distrazione vengano cancellate intere directory o file di grande importanza.
Un'altra problematica molto frequente è legata alla scelta di acquistare PC economici o con garanzie limitate. Si preferisce spendere il meno possibile e concepire il computer come un mero strumento di lavoro, da "spremere" fino al limite e da cambiare con un'altra macchina altrettanto economica non appena si rompe.
Questa filosofia può essere valida per le macchine da lavoro, ma è diametralmente opposta all'idea di fondo che sta alla base di un sistema server. L'affidabilità e la stabilità dovrebbero in questo caso essere i concetti chiave, dato che un sistema di questo tipo raccoglie e smista tutti i file aziendali.
In molti casi non esiste la cultura del sistema server, ma spesso intervengono anche altri fattori, come l'impossibilità o il rifiuto di sostenere il costo di un sistema operativo server. È vero che una macchina server può funziona-

re anche con un normale sistema operativo client, ma in tal caso si perdono molti dei vantaggi connessi al sistema server e si preferisce nuovamente risparmiare e comprare una macchina standard, magari assemblata.
È possibile uscire da questo *empasse* aggirando i costi di licenza, mediante l'utilizzo del sistema operativo Linux. È infatti possibile configurare una macchina linux in modo da farla dialogare all'interno di un workgroup Windows e da supportare condivisioni di cartelle e stampanti, come se fosse dotata anch'essa di un sistema operativo Windows. Se non si verificano problemi di compatibilità, i computer Windows accederanno al sistema Linux in maniera completamente trasparente all'utente.
Può suonare strano il fatto che sia possibile simulare un ambiente di rete Windows con macchine che girano con sistemi operativi non Microsoft. Questo è possibile grazie alla stratificazione e all'apertura di protocolli.
Le operazioni di condivisione all'interno di una rete Windows funzionano attraverso un protocollo di alto livello che è preposto a proiettare sulla rete locale alcune entità locali, che sono i file, le cartelle e le stampanti. Il nome di questo protocollo è SMB (*Short Message Block*).
Qualunque sistema operativo che implementi questo protocollo è in grado di accedere a risorse in una rete Windows o a fornirne le proprie risorse a una rete di computer preesistenti.
Linux dispone di un'ottima implementazione SMB all'interno di un pacchetto denominato Samba (www.samba.org), un nome che altro non è che la sigla SMB con l'aggiunta di due vocali.
Il progetto è nato per la piattaforma Unix e ha acquisito nel tempo sempre maggiore importanza, diventando oggi uno dei pacchetti aperti più importanti. Grazie a esso è possibile interagire con i sistemi Microsoft così largamente usati.

Struttura base di Samba

Il pacchetto Samba può essere scaricato dal sito ufficiale del progetto dalla sezione di download. Si consiglia però di fare uso del pacchetto predisposto appositamente per la propria distribuzione. Questo semplifica notevolmente le operazioni di installazione, automatizzando molti passaggi.
La versione più diffusa di Samba è la 2, ora sostituita dalla più completa versione 3. Si consiglia di fare uso di questa release per avere un maggior numero di funzionalità compatibili con i sistemi Microsoft.
La versione 3 include molte caratteristiche importanti, elencate di seguito.

- Condivisione di cartelle e stampanti in rete.
- Possibilità di accedere a risorse condivise in altre macchine che usano SMB.
- Supporto di workgroup.

- Supporto per l'autenticazione su un dominio Windows.
- Gestore della master browser list.
- Funzioni WINS.
- Supporto per Active Directory in qualità di membro.
- Autenticazione Kerberos.
- Supporto al protocollo LDAP.
- Pubblicazione degli attributi delle stampanti in Active Directory.
- Relazioni di fiducia tra server NT4 e Samba.

Samba è composto da due demoni, `nmbd` e `smbd`: il primo si occupa di tutte le operazioni di risoluzione dei nomi. Questo demone è responsabile della gestione della master browser list e del servizio WINS.
La master browser list è un servizio molto importante. Quando si fa clic su *Risorse di rete* e si vedono le icone dei computer presenti si sta in realtà leggendo un elenco mantenuto da un computer ben preciso assegnato a questo scopo.
Quando si accende un altro computer in rete questo cerca il gestore dell'elenco e segnala a tale sistema la propria presenza, aggiornando l'elenco. È per questo motivo che i computer che entrano per la prima volta in rete impiegano un po' di tempo prima di essere visibili in *Risorse di rete*.
Allo stesso modo quando si spegne un computer questo rimane ancora visibile nell'elenco per un certo periodo di tempo.
Durante lo spegnimento non viene infatti eseguita una notifica al gestore e il proprio nome rimane attivo per un certo numero di minuti prima che venga automaticamente ripulito.
Il gestore dell'elenco dei computer è definito *Master Browser List*. Il ruolo viene assegnato a una delle macchine presenti attraverso un meccanismo di elezione. Durante questa fase sono presi in considerazione diversi parametri tra cui il periodo di uptime, la versione dei protocolli usati, il sistema operativo utilizzato (client o server) ecc. Il migliore sistema diventa il master browser list fino a quando viene spento o quando le elezioni sono nuovamente indette e il sistema locale risulta inferiore come numeri a un nuovo sistema. Questo può succedere se in una rete di soli client Windows si attiva un sistema Linux con Samba. Tale macchina ha i numeri migliori rispetto ai client, se non altro perché è un server e vince le elezioni acquisendo la gestione della master browser list.
Un altro servizio molto importante in una rete di computer Windows è WINS. Si tratta di un meccanismo che permette la risoluzione di un nome esteso (per esempio amministrazione4) nel suo indirizzo IP (per esempio 192.168.100.14).
Bisogna sempre tenere presente che l'accesso a un sistema in rete può avvenire solamente tramite indirizzo. I nomi testuali devono sempre essere tradotti in un indirizzo standard prima di poter essere usati.

Se non c'è un servizio WINS la risoluzione avviene con un broadcast. Il computer interroga tutti i computer connessi sul cavo di rete prima di scoprire quale macchina dispone del nome di rete a cui si vuole accedere. L'operazione comporta però carico inutile sulla rete e genera ritardi.
È molto meglio fare riferimento a un archivio in cui sono elencati i nomi dei computer con i relativi indirizzi IP. Scorrendo questo elenco alla ricerca del nome computer si ottiene immediatamente l'indirizzo IP richiesto senza generare traffico inutile sulla rete.
Il servizio WINS è molto simile al DNS. Anche questo infatti eroga un indirizzo IP a seguito di una interrogazione per nome. Sussistono però alcune differenze sostanziali.
Il DNS è un servizio gerarchico concepito per funzionare in reti geograficamente distribuite come Internet. Grazie a questa struttura ogni sistema DNS contiene i dati dei sistemi della propria tratta di rete controllata. Il sistema è però collegato a tutti gli altri e, quando non è possibile ottenere l'indirizzo di un computer dall'archivio locale, si può percorrere tutta la struttura DNS fino a trovare il sistema che contiene le informazioni alle quali si vuole accedere.
WINS non è in grado di fare tutto questo. L'archivio è un semplice elenco di macchine e indirizzi senza alcuna struttura gerarchica. Ogni WINS deve infatti contenere la totalità dell'elenco. Questa scelta è data dal fatto che il servizio nasce per servire reti locali.

Configurazione generale di Samba

Il secondo demone di Samba è `smbd`, che gestisce le connessioni e le operazioni di condivisione.
I due demoni devono essere lanciati nell'ordine corretto. Prima dovrebbe partire `nmbd` e poi `smbd`. Questi demoni possono essere lanciati manualmente invocandoli con la seguente sintassi:

```
/etc/init.d/nmbd start
/etc/init.d/smbd start
```

In alcune distribuzioni esiste una modalità "abbreviata" per l'attivazione di Samba:

```
/etc/init.d/smb start
```

Questo provvede a lanciare i singoli demoni `smbd` e `nmbd`.
I demoni dovrebbero essere comunque attivati durante la fase di avvio del sistema. A tal proposito bisogna verificare lo script di avvio, oppure usare tool specifici quali `ntsysv` per fare in modo che i demoni partano automaticamente durante l'avvio di Linux.

All'avvio di Samba viene letto il file `smb.cfg` presente dentro `/etc/smb`, che è un file di configurazione di esempio. Molti utenti aprono questo file e si ispirano alla configurazione presente per creare la propria installazione. Si sconsiglia questo approccio in quanto Samba è un pacchetto molto complesso e ricco di opzioni. L'utilizzo di un file di configurazione generico, scritto più che altro per scopi dimostrativi, non è il modo migliore per cominciare. Si rischia di fare molta confusione e di perdere il controllo della situazione. Molto meglio partire da capo e scrivere una configurazione mirata.
Prima di tutto bisogna fermare il servizio Samba e mettere al sicuro il file di configurazione iniziale salvandolo con un altro nome:

```
/etc/init.d/smb stop
cd /etc/samba
mv smb.conf smb.conf.old
```

A questo punto si può creare un nuovo file di configurazione:

```
vi smb.conf
```

Il file di configurazione di Samba ha una struttura lineare e precisa. È presente una sezione generale `global`, in cui sono contenute una serie di indicazioni che regolano il funzionamento del servizio come, per esempio, il nome del server, il tipo di server, la visibilità in rete, i criteri di sicurezza ecc. La sezione `global` è seguita da una serie di descrizioni più specifiche che dettagliano le condivisioni, che si vogliono mettere in atto nel proprio sistema.
Un file di configurazione di Samba può contenere anche solo la sezione generale, ma questo non ha molto senso perché, in tal caso, non si avrebbero risorse condivise nella rete locale. Si vuole invece utilizzare Samba per mettere in atto un file server e fornire servizi di condivisione agli utenti.
Per imparare a configurare Samba in modalità workgroup si prenderà in considerazione un piccolo studio professionale che ha la necessità di creare un'area file comune dove salvare i file di gruppo, e un'area file per contenere le utilità di uso generale (per esempio le patch del sistema operativo, l'ultima versione del browser, il software per comprimere archivi ecc.).
L'area dei file comuni viene chiamata `comune`, è di libero accesso e chiunque può leggere e scrivere dati. L'area con i file di utilità generale è invece gestita dall'amministratore locale, si chiama `software` ed è accessibile unicamente in lettura.
Per cominciare si deve scrivere la sezione generale:

```
[global]
workgroup = GRUPPOLAVORO
netbios name = SERVER1
server string = file server Linux
security = SHARE
```

La sezione comincia con l'etichetta `[global]` presente sulla prima riga. Tale indicazione deve essere riportata fedelmente.
La riga seguente `workgroup = GRUPPOLAVORO` indica a Samba qual'è il nome del workgroup della rete. Se le macchine Windows presenti fanno già parte di un gruppo di lavoro bisogna riportare a destra della direttiva `workgroup` il nome di quest'ultimo. Se si decide di utilizzare un altro nome, sfogliando l'icona di *Risorse di rete* si vedranno due gruppi di lavoro diversi.
Se si sta configurando la rete per la prima volta, bisogna verificare che i client di rete, per esempio le macchine con Windows XP, siano configurati per far parte dello stesso gruppo di lavoro del server (Figura 1.1).

Figura 1.1
Configurazione del gruppo di lavoro, in Windows XP.

La terza riga del file `netbios name = SERVER1` indica il nome con cui il server Linux sarà visibile quando si andranno a sfogliare le *Risorse di rete* dei client: facendo clic sul link *Risorse di rete,* poi sulle voci *Tutta la rete* e *Rete di Microsoft Windows,* potrete accedere al workgroup *GRUPPOLAVORO,* in cui troverete un'icona di computer seguita dal nome *SERVER1*. Questa è la macchina Linux che state configurando.
Di seguito all'indicazione del nome `netbios` compare la direttiva `server string`. Questa contiene il commento a testo libero che viene associato al nome del server in rete. Sfogliando la rete si vedrà così la macchina *SERVER1* con a fianco il commento file server Linux.
È possibile omettere questa indicazione ma, in tal caso, Samba includerà come commento il proprio numero di versione.

I nomi usati nella seconda e nella terza riga del file di configurazione sono di libera scelta, anche se devono soddisfare alcuni criteri. Il nome del gruppo di lavoro dovrebbe essere privo di spazi e non superare la lunghezza di 15 caratteri. Anche il nome NetBIOS del computer non dovrebbe superare i 15 caratteri di lunghezza.
La quarta riga contiene una direttiva che indica il tipo di sicurezza che si vuole implementare per l'accesso al server da parte dei client. Le opzioni base sono due: SHARE e USER.
Nella modalità SHARE l'accesso alle condivisioni avviene semplicemente fornendo una password, se richiesta. Non vi è alcuna altra forma di autenticazione.
Questa modalità è simile alle condivisioni di directory che si possono creare in Windows 98 o Windows XP, in cui è possibile anche assegnare una password per l'accesso (Figura 1.2).

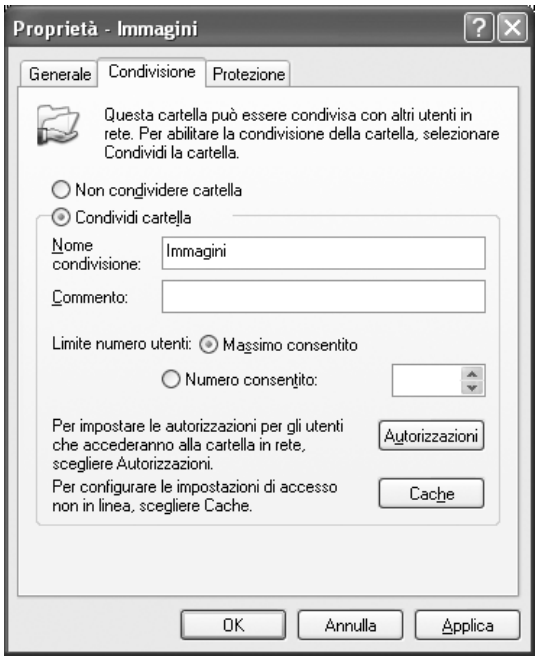

Figura 1.2
La scheda Condivisione relativa a una cartella

La modalità *user* è invece più articolata e richiede anche una user id, durante la fase di autenticazione.
Per cominciare si prenderà in considerazione la modalità SHARE e per inquadrare meglio il concetto si scriveranno le direttive necessarie per creare una delle due cartelle condivise dello studio professionale. Si lavora nuovamen-

te sul file di configurazione precedente e in fondo si aggiungono le seguenti righe:

```
[comune]
comment = cartella comune
path = /home/comune
public = YES
writable = YES
```

Il nome del blocco scritto tra parentesi quadre corrisponde al nome che la directory condivisa avrà in rete. In pratica si sta creando un blocco denominato `comune`, che definisce le proprietà della condivisione `comune`.
La direttiva `comment` è a testo libero e rappresenta il commento che si può osservare quando si chiedono i dettagli nella visualizzazione dei file e delle directory.
La direttiva `path` indica la posizione nel file system di Linux dove la directory è effettivamente collocata. In questo caso si è scelto di localizzare la directory `comune` nella directory `home`. Bisogna a tal proposito verificare che i permessi di accesso siano corretti.
Trattandosi di una directory a libero accesso si possono usare i diritti `777`, per fare in modo che tutti abbiano diritti pieni:

```
chmod 777 /home/comune
```

Le due direttive seguenti indicano rispettivamente che la cartella è di pubblico accesso ed è possibile scrivere all'interno della directory condivisa.
Ora si procederà a realizzare la cartella software accessibile in sola lettura agli utenti. La definizione del blocco è molto simile a quella della directory `comune`:

```
[software]
comment = file di utilita'
path = /home/software
public = YES
writable = NO
```

L'unica differenza rilevante è il fatto che la direttiva `writable` è impostata a `NO`.
Ogni volta che si esegue una modifica nel file di configurazione di Samba bisogna ricordarsi di riavviare il servizio per fare in modo che le modifiche siano immediatamente applicate. Samba ricarica a intervalli regolari il file generale di configurazione, ma l'intervallo è relativamente lungo e le nuove modifiche non sono perciò applicate in tempo reale.
Per riavviare manualmente il servizio basta utilizzare questa sintassi:

```
/etc/init.d/smb restart
```

Prima di questa operazione è bene verificare che la sintassi apportata sia corretta. Per farlo si può impartire il seguente comando:

```
testparm
```

Questa utility è parte della suite Samba ed è preposta a verificare la correttezza sintattica del file di configurazione. Il comando legge il file di configurazione `smb.conf`, esegue una scansione delle sezioni e indica eventuali errori presenti.
Se tutto funziona, verrà indicato anche il ruolo del server configurato (in questo caso `STANDALONE`) e premendo il tasto Invio si vedrà anche il dump dei parametri principali della configurazione realizzata:

```
Load smb config files from /etc/samba/smb.conf
Processing section "[comune]"
Processing section "[software]"
Loaded services file OK.
Server role: ROLE_STANDALONE
Press enter to see a dump of your service definitions

# Global parameters
[global]
        workgroup = GRUPPOLAVORO
        server string = file server Linux
        security = SHARE

[comune]
        comment = cartella comune
        path = /home/comune
        read only = No
        guest ok = Yes

[software]
        comment = file di utilita'
        path = /home/software
        guest ok = Yes
```

Questo comando di diagnosi è molto importante perché i demoni di Samba leggono il file di configurazione ma non segnalano eventuali errori presenti. L'amministratore di sistema potrebbe essere così indotto a pensare che la configurazione realizzata sia regolare, quando invece sono presenti errori di sintassi.
Il comando `testparm` accetta alcune opzioni. Una di queste è `-v`, tramite la quale è possibile visualizzare i valori di tutte le direttive che non sono state utilizzate nel file di configurazione. Alcune di queste hanno valori di default assegnati automaticamente. È così possibile vedere come queste variabili sono state assegnate dal sistema.
Un ulteriore strumento di diagnosi e analisi del server è l'utility `smbclient`. Si tratta di un'interfaccia testuale verso i servizi di Samba. Le funzionalità incluse sono numerose. Si consiglia a tal proposito di leggere il manuale del comando, tramite l'istruzione `man smbclient`. Uno di queste opzioni è `-L`. Questa visualizza lo stato del servizio `smb` sul sistema.
La sintassi da usare è `smbclient -L` *`nomeserver`*, per esempio:

```
smbclient -L 127.0.0.1
```

Questo è l'output di un server Linux che gestisce il workgroup GRUPPOLAVORO e che si trova in una rete dotata di secondo gruppo di lavoro, denominato SEDE:

```
Domain=[GRUPPOLAVORO] OS=[Unix] Server=[Samba 3.0.4-1.FC1]

        Sharename       Type      Comment
        ---------       ----      -------
        comune          Disk      cartella comune
        software        Disk      file di utilita'
        IPC$            IPC       IPC Service (file server Linux)
        ADMIN$          IPC       IPC Service (file server Linux)

Domain=[GRUPPOLAVORO] OS=[Unix] Server=[Samba 3.0.4-1.FC1]

        Server               Comment
        ---------            -------
        SERVER1              file server Linux

        Workgroup            Master
        ---------            -------
        GRUPPOLAVORO         SERVER1
        SEDE                 TECRA-M1
```

Gestione degli utenti

Le esigenze di gestione dei file dello studio tecnico sono state pienamente risolte con la configurazione precedentemente enunciata. Col tempo però le esigenze possono cambiare e potrebbe sorgere la necessità di realizzare accessi differenziati per gli utenti.
Lo studio tecnico, a seguito di un incremento degli affari, potrebbe, per esempio, aver bisogno di utilizzare un software gestionale. La soluzione è di disporre di un'area condivisa di supporto dove memorizzare i database che saranno usati da tutte le postazioni client presenti in rete.
Non si vuole però che l'area sia disponibile a tutti ma solo al personale dell'amministrazione, più precisamente agli utenti amm1, amm2 e amm3. Gli altri utenti della rete non devono invece averne accesso.
Si vuole anche che la cartella sia vista in rete con il nome gestionale.
Per portare a termine questa necessità bisogna modificare alcuni aspetti sostanziali della configurazione precedente. Innanzitutto non va più bene la protezione a livello di condivisione ma bisogna piuttosto adottare la sicurezza a livello utente. Non si vuole che ci sia una semplice password di blocco ma si deve fare in modo che solo gli account amm1, amm2 e amm3 possano accedere con le relative credenziali.
La configurazione del blocco global deve essere modificata come nel seguente listato:

```
[global]
workgroup = GRUPPOLAVORO
netbios name = SERVER1
```

```
server string = file server Linux
security = USER
smb passwd file = /etc/samba/smbpasswd
encrypt passwords = YES
```

Un aspetto molto importante è che il tipo di security è stato impostato a `USER`. Ora l'autenticazione avviene tramite la coppia di valori user id/password, fornita all'accesso delle condivisioni.
Sotto l'opzione `security` sono state inserite due direttive supplementari. La prima indica dove si trova il file che mantiene le password di accesso per gli utenti di Samba, e la seconda determina che si stanno utilizzando password cifrate per accedere a Samba. In tal caso bisogna tener presente che macchine Windows 95, NT 3 e NT4 pre-SP3 potrebbero aver problemi di accesso per via dell'uso di password trasmesse in chiaro. Si consiglia per motivi di sicurezza di aggiornare il sistema operativo di questi sistemi.
Le definizioni per le condivisioni `comune` e `software` restano invariate, mentre cambia la struttura della definizione per la condivisione `gestionale`:

```
[gestionale]
comment = area supporto software gestionale
path =/home/gestionale
valid users = amm1 amm2 amm3
writable = YES
```

La direttiva importante è `valid users`, all'interno della quale è possibile elencare gli utenti abilitati ad accedere alla directory `/home/gestionale`, condivisa con il nome `gestionale`.
Eventualmente è possibile inserire una direttiva `invalid users` ed elencare gli account che non possono accedere a gestionale, per esempio i restanti utenti della rete:

```
[gestionale]
comment = area supporto software gestionale
path =/home/gestionale
valid users = amm1 amm2 amm3
invalid users = ufftec1 ufftec2 ufftec3 ufftec5 marketing reception comm1
comm2
writable = YES
```

Indicare gli utenti nel file di configurazione non è sufficiente. Bisogna anche creare gli stessi account sul server Linux. Samba si appoggia infatti sul sistema degli utenti di Linux per funzionare.
Il primo passaggio consiste nel creare gli utenti sulla macchina Linux, avvalendosi degli strumenti standard del sistema:

```
useradd amm1
useradd amm2
useradd amm3
```

In seguito bisogna fornire le password singolarmente a tutti gli utenti:

```
[root@server1 samba]# passwd amm1
Changing password for user amm1.
New password: ********
Retype new password: ********
passwd: all authentication tokens updated successfully.
```

La stessa procedura deve essere seguita per `amm2` e `amm3`.
Il server dispone ora dei nuovi account utente con tanto di directory utente in `/home` e di password di accesso al sistema.
Il sistema Samba utilizza gli account di Linux per erogare le funzioni di autenticazione ma usa un proprio file di password per regolare l'accesso alle risorse. La password di accesso a Linux e quella di Samba per un dato account potrebbero quindi essere differenti. Si sconsiglia però di compiere una scelta simile per motivi di omogeneità della configurazione di sistema.
L'utility di Samba per l'impostazione delle password si chiama `smbpasswd`, non fa altro che aggiungere una riga all'interno del file delle password, `smbpasswd`, presente dentro `/etc/samba` con la user id e la password cifrata.
La sintassi di base è molto semplice in quanto basta digitare il comando seguito dall'opzione `-a` e dal nome utente. Poi si inserisce la password e si conferma:

```
[root@server1 samba]# smbpasswd -a amm1
New SMB password: ********
Retype new SMB password: ********
Added user amm1.
```

Ora l'utente è correttamente definito all'interno del server Linux e di Samba.
Le copie di user id e password utilizzate in fase di configurazione dovrebbero essere le stesse adottate nei sistemi Windows client presenti in rete. In questo modo tutta la procedura di accesso dalle postazioni utente è automatica in quanto Windows comunicherà le credenziali al momento dell'accesso alla directory. Se le credenziali di Windows e quelle di Samba/Linux sono differenti comparirà una finestra con la richiesta di inserimento manuale delle credenziali.

Directory utente

Samba dispone di una funzionalità molto interessante, che permette la creazione veloce di directory utente. Fino a questo punto della trattazione si è osservato che ogni singola condivisione deve essere generata manualmente, all'interno di un blocco contenente specifiche direttive di funzionamento, come i blocchi `comune` o `gestionale` creati precedentemente.

Questo modo di procedere può risultare scomodo, se si hanno decine di utenti e si vogliono creare directory utente per ognuno di essi. Si dovrebbe infatti creare un blocco specifico per ciascun utente.
È possibile risolvere questo problema ricorrendo al blocco `homes`, che provvede alla creazione automatica di una condivisione personale per ogni utente correttamente configurato in Samba. Da un punto di vista pratico, viene letto l'elenco degli utenti di Samba (in pratica gli utenti creati con il comando `smbpasswd` e perciò presenti nel file `smbpasswd`).
Di ogni utente presente nel file sarà condivisa la directory utente Linux, presente nella directory `/home`. Si tratta nuovamente di sfruttare il legame tra Samba e il sistema operativo.
Sarà possibile visualizzare la condivisione utente aprendo l'icona del server in *Risorse di rete*. Gli utenti ordinari potranno lavorare senza nessun problema, mentre i tecnici che accedono anche al server Linux potranno vedere una cartella utente unica, sia per Linux sia per Windows.
Una configurazione standard per il blocco `homes` può essere la seguente, generalmente inserita di seguito al blocco `global`:

```
[homes]
comment = cartella utente
writable = YES
```

Il blocco contiene il commento a testo libero e l'indicazione che è accessibile in lettura.
Accedendo al server tramite *Risorse di rete* si vedranno le cartelle condivise generali, la cartella personale dell'utente e la cartella *homes*.
La cartella personale dell'utente avrà lo stesso nome del proprio account di login, per esempio `amm1`. Il contenuto sarà liberamente accessibile e si potranno apportare tutte le modifiche necessarie.
La cartella *homes* visibile nell'elenco delle condivisioni è in questo caso un doppione che punta alla propria cartella utente personale.
Si tratta di una scelta progettuale di Samba ma, volendo, è possibile fare in modo che la condivisione `homes` non sia visibile. Basta utilizzare la direttiva browsable impostata a `NO`.
La direttiva `browsable = NO` fa in modo che la condivisione in oggetto non sia visibile (pur rimanendo presente). È la versione Samba delle condivisioni nascoste di Windows. La direttiva può essere usata nella definizione di qualunque blocco come prima misura di sicurezza.
Il file di configurazione assume ora questo aspetto:

```
[homes]
comment = cartella utente
writable = YES
browsable = NO
```

Pur vedendo solo la propria cartella personale è possibile avere accesso a tutte le cartelle personali presenti sul server. Se l'utente `amm1` vuole accedere

alla cartella personale di `amm4` può usare questa sintassi sulla barra dell'indirizzo di qualunque finestra:

```
\\server1\amm4
```

Premendo Invio si aprirà la finestra relativa con il contenuto della home di `amm4`.
Non si tratta di un problema di funzionamento di Samba come molti amministratori sono indotti a pensare. Si tratta ancora una volta di una scelta progettuale ben precisa.
Spetta infatti al gestore della rete applicare opportuni diritti alle cartelle oppure impiegare ulteriori direttive sul file di configurazione di Samba per incrementare la sicurezza.
Si potrebbe in questo caso limitare l'accesso tramite la direttiva `valid users`. Non è però possibile scrivere una singola regola statica che permetta l'accesso a una determinata cartella utente alla sola persona autorizzata e non agli altri utenti di Samba, a meno che non si utilizzino le variabili.
Samba permette di specificare nei campi del file di configurazione informazioni variabili che sono gestite durante il funzionamento del sistema stesso. Le variabili in Samba iniziano sempre con il carattere `%`. La Tabella 1.1 elenca le variabili di uso più comune.

Tabella 1.1 Variabili di uso comune in Samba

Variabili che riguardano il client che esegue il login	
`%a`	Tipo di sistema operativo client (esempio Samba, WinNT, WfWg, Win95 o UNKNOWN)
`%I`	Indirizzo IP
`%m`	Nome NetBIOS
`%M`	Nome DNS
Variabili che riguardano l'utente	
`%u`	User id Unix corrente
`%U`	User id del client che ha richiesto l'accesso
`%g`	Gruppo dell'utente %u
`%H`	Directory utente di %u
Variabili che riguardano le condivisioni	
`%S`	Nome della condivisione corrente
`%P`	Radice della condivisione corrente
Variabili che riguardano il server	
`%h`	Nome DNS del server Samba
`%L`	Nome NetBIOS del server Samba
`%v`	Versione di Samba

Per risolvere il problema dell'accesso alle condivisioni si può utilizzare la variabile `%S`, che contiene il nome della condivisione corrente. Dal momen-

to che la directory utente personale ha il nome uguale a quello proprio di login è possibile usare la variabile come vincolo di accesso per la direttiva `valid users`:

```
valid users = %S
```

Riavviando il servizio Samba si potrà accedere alla propria cartella utente ma non a quelle di altri utenti. Questo perché l'utente valido per le cartelle `amm1`, `amm2`, `amm3` e così via sarà rispettivamente solo gli utenti `amm1, amm2, amm3` ecc. Durante l'implementazione delle directory utente potrebbero verificarsi alcuni errori come, per esempio, il non poter accedere o scrivere nelle aree utenti, pur avendo configurato correttamente le direttive di Samba.
L'accesso alle directory utente da parte di Samba avviene con lo stesso utente con cui il client ha eseguito il login. Se si è autenticati come `amm1` dalla postazione Windows, l'accesso alle cartelle condivise avverrà attraverso l'utente `amm1`. Questo è ancora una volta il motivo per cui gli utenti Samba devono anche avere un utente Linux.
Le singole directory utenti dentro `/home` dovrebbero essere di proprietà del rispettivo utente. La directory `amm1` deve appartenere all'utente `amm1`, la directory `amm2` deve appartenere all'utente `amm2` e così via. Si deve usare a tal proposito il comando `chown`:

```
chown amm1 amm1
chown amm2 amm2
chown amm3 amm3
```

Alle singole cartelle si dovrebbero assegnare i diritti 770 per fare in modo che solo l'utente e il proprio gruppo possano accedere alla cartella. Questo è utile anche come misura di sicurezza su Linux.
Se l'utente `amm2` esegue il login su Linux non può leggere il contenuto della directory `/home/amm1`. Se si preferisce, è possibile creare le directory utente in una posizione alternativa e non intaccare la struttura delle cartelle utente di Linux. Questa scelta può essere utile per evitare che gli utenti Windows vedano i file di supporto che vengono creati automaticamente da Linux per un nuovo utente.
Per creare le directory utente in un altro punto del file system si utilizza la seguente sintassi:

```
path = /usr/local/sambahome/%S
```

In questo caso, si sta istruendo Samba a considerare la radice delle directory utente in `/usr/local/sambahome`. I percorsi verso le singole cartelle sono generati con la variabile `%S`.
Bisogna naturalmente avere l'accortezza di creare la radice (`/usr/local/sambahome`), le singole cartelle utente e applicare i diritti e i proprietari nel modo indicato più in alto.

Condivisione dell'unità CD-ROM

Un caso particolare di condivisione è l'unità CD-ROM in quanto l'ambiente Linux richiede che questa sia montata prima dell'utilizzo.
Se si intende condividere un particolare disco CD-ROM in maniera permanente, per esempio un elenco telefonico, si può allora entrare nel sistema Linux come amministratore, montare l'unità con il comando `mount` (esempio `mount/dev/cdrom`) e creare un blocco del tutto simile a quelli precedenti:

```
[cdrom]
comment = Elenco telefonico
browsable = YES
read only = YES
path = /mnt/cdrom
```

Se si vuole invece cambiare liberamente il CD-ROM sull'unità condivisa è necessario ricorrere a due parametri specifici, che eseguono il mounting automatico all'ingresso della condivisione e l'operazione di unmounting all'uscita:

```
[cdrom]
comment = Unita' cdrom
browsable = YES
read only = YES
path = /mnt/cdrom
root preexec = mount /dev/cdrom
root postexec = umount /mnt/cdrom
```

Il comando `root preexec` esegue le operazioni indicate a destra del simbolo `=` durante l'accesso alla condivisione. Le operazioni sono eseguite dall'utente root. Il comando `root postexec` esegue l'operazione indicata all'uscita della condivisione. In questo caso sono indicati due comandi di sistema per i dispositivi, ma è possibile inserire qualunque altro comando o script si riveli necessario ai propri scopi.
Queste direttive non sono inoltre limitate a un blocco di definizione per un CD-ROM ma possono essere presenti in qualunque blocco di definizione delle condivisioni.

Sicurezza di Samba

Esistono alcune semplici direttive che permettono di incrementare la sicurezza della propria installazione di Samba e possono essere incluse dentro un blocco di definizione di una condivisione oppure nel blocco `global`.

Prima di tutto è possibile discriminare gli accessi anche in base agli indirizzi IP:

```
hosts allow = 127.0.0.1 192.168.100.0/24
hosts deny = 192.168.100.13
```

La prima direttiva permette di specificare quali nodi o sottoreti possono accedere alla condivisione. In questo caso il sistema locale `127.0.0.1` e la sottorete `192.168.100.0`.
È possibile indicare anche solo stringhe parziali come `192.168.200.` e `127.` oppure nomi simbolici risolvibili dal server, come `localhost` o `server1`.
La direttiva `host deny` esegue la procedura opposta: specifica quali nodi o reti non possono accedere.
Samba risponde alle richieste provenienti da tutte le interfacce fisiche e logiche attivate sul server locale. Questo significa che se è presente un modem ADSL connesso, utenti esterni possono accedere al sistema Samba e sfogliare le condivisioni. Si tratta di una situazione molto pericolosa.
Per risolvere il problema si deve usare la direttiva `interfaces` e indicare le interfacce abilitate a inoltrare traffico per Samba:

```
interfaces = eth0 eth1 lo
bind interfaces = yes
```

Vengono abilitate le due schede di rete presenti nel sistema e l'interfaccia di loopback locale. La direttiva `bind interfaces` fa in modo che il traffico sia possibile solo sulle interfacce indicate nella direttiva interfaces.
Queste direttive sono posizionate nel blocco `global`.
Naturalmente è fondamentale implementare un firewall perimetrale sulla rete oppure sul computer locale (indicazioni in merito sono presenti sul capitolo dedicato ai firewall). Le porte di servizio usate dal protocollo SMB/CIFS sono la 137, 138, 139 e 445. Queste devono essere chiuse dall'esterno e aperte all'interno.
All'interno delle cartelle utente potrebbero essere presenti file con link simbolici che puntano a file e a directory presenti in altri punti del file system. Per evitare che sia possibile seguire i link simbolici si può usare la seguente direttiva:

```
follow symlinks = no
```

Per quanto riguarda la registrazione degli eventi è presente un meccanismo di logging attivato per default. Eventualmente è possibile modificare le impostazioni, inserendo nel blocco `global` la direttiva `log`:

```
log file = /var/log/samba/%m.log
max log size = 100
log level = 3
```

La prima direttiva indica il percorso dove salvare i messaggi di log. Si noti che è stata usata una variabile per fare in modo di avere un file di log per ogni sistema differente che accede a Samba. Per sistema si intende in questo caso il nome NetBIOS assegnato alla macchina.
Con `max log size` si indica la dimensione massima per il file di log. Il valore `0` sta a indicare nessun limite. Superato questo limite il file di log corrente viene rinominato come `.old` e si crea un nuovo file.
Con `log level` si stabilisce il livello di dettaglio del file di log. Il valore `3` è molto elevato e fornisce un output particolarmente dettagliato sul file di log, utile per le operazioni di debugging o di verifica delle configurazioni. Non si consiglia questo livello, come pure il `2` (se pur meno prolisso) in quanto comporta un carico sul sistema Samba con un conseguente calo delle prestazioni. Meglio usare il livello `1` in produzione e passare a livelli superiori in caso di necessità.

Configurazione dei client

La configurazione dei client è rapida: basta portare il puntatore del mouse sopra l'icona delle *Risorse del computer* di Windows XP, premere il tasto destro del mouse e selezionare *Proprietà*. Compare una finestra da cui bisogna selezionare la scheda *Nome computer*, poi si deve premere il pulsante *Cambia*. Nella finestra che si apre, indicare il nome `GRUPPOLAVORO` per il gruppo di lavoro, confermare e riavviare il computer. All'attivazione seguente di Windows, la macchina farà parte del workgroup indicato.
Per velocizzare le operazioni di accesso in rete è importante attivare un sistema DNS, inserire le generalità del server e di tutti i client e poi configurare ogni macchina dalle proprietà di *Risorse di rete* per utilizzare il DNS. Informazioni sulla creazione di un sistema DNS con Linux vengono fornite nel relativo capitolo.
Nel listato che segue è riportato il contenuto del file di configurazione `smb.cfg` di Samba, così come lo abbiamo costruito nel corso di questo capitolo.

```
[global]
workgroup = GRUPPOLAVORO
netbios name = SERVER1
server string = file server Linux
security = USER
smb passwd file = /etc/samba/smbpasswd
encrypt passwords = YES
log file = /var/log/samba/%m.log
max log size = 100
log level = 1

[homes]
comment = cartella utente
writable = YES
browsable = NO
valid users = %S
```

```
# path = /usr/local/sambahome/%S

[comune]
comment = cartella comune
path =/home/comune
public = YES
writable = YES

[software]
comment = file di utilita'
path =/home/software
public = YES
writable = NO

[gestionale]
comment = area supporto software gestionale
path =/home/gestionale
valid users = amm1 amm2
writable = YES

[cdrom]
comment = Elenco telefonico
browsable = YES
read only = YES
path = /mnt/cdrom
root preexec = mount /dev/cdrom
root postexec = umount /mnt/cdrom
```

Checklist

1. Verificare che il pacchetto Samba sia installato nel proprio sistema, digitando il comando `testparm`: se compare un messaggio di errore significa che Samba non è presente e bisogna procedere alla sua installazione, dai CD-ROM del sistema operativo oppure attraverso risorse online.
2. Salvare in un luogo sicuro il file di configurazione di default di Samba, quindi creare un nuovo file `smb.conf` nella directory `/etc/samba`.
3. Aprire il file `smb.conf` e creare la sezione `global`.
4. Inserire nella sezione `global` le direttive `workgroup`, `netbios name` e `server string`.

Cartelle condivise di accesso pubblico

1. Impostare nella sezione `global` il livello di sicurezza della condivisione con la direttiva `security = SHARE`.
2. Impostare una condivisione, creando nel file di configurazione una sezione che dovrà contenere almeno le direttive `comment` e `path`.

3. Inserire la direttiva `public = YES` per rendere pubblica la condivisione.
4. Inserire la direttiva `writable = YES` per rendere accessibile in scrittura la condivisione.
5. Impostare correttamente i permessi nella directory relativa alla condivisione.
6. Ripetere i passaggi da 6 al 9 per tutte le ulteriori condivisioni.
7. Impostare le direttive di sicurezza.
8. Verificare la correttezza della configurazione, richiamando il comando `testparm`.
9. Avviare i demoni di Samba e fare in modo che siano sempre lanciati a ogni avvio del sistema.
10. Configurare i client.

Cartelle condivise protette da user id e password

1. Impostare nella sezione `global` il livello di sicurezza dell'utente con la direttiva `security = USER`.
2. Impostare il file delle password tramite la direttiva `smb passwd file`.
3. Indicare la gestione delle password cifrate tramite la direttiva `encrypt passwords = YES`.
4. Impostare una condivisione, creando nel file di configurazione una sezione che deve contenere almeno le direttive `comment` e `path`.
5. Indicare l'elenco degli utenti autorizzati ad accedere alla condivisione con la direttiva `valid users`.
6. Specificare gli eventuali utenti non autorizzati utilizzando la direttiva `invalid users`.
7. Inserire la direttiva `writable = YES` per rendere accessibile in scrittura la condivisione.
8. Impostare correttamente i permessi per directory della condivisione.
9. Ripetere i passaggi da 6 al 12 per tutte le ulteriori condivisioni.
10. Verificare la correttezza della configurazione richiamando il comando `testparm`.
11. Creare sul sistema Linux gli account degli utenti Windows che potranno accedere alle condivisioni.
12. Creare le password di Samba per gli account appena creati; a tal proposito si utilizza il comando `smbpasswd`.
13. Impostare le direttive di sicurezza per il sistema.

14. Verificare la correttezza della configurazione utilizzando il comando `testparm`.
15. Avviare i demoni di Samba e fare in modo che siano sempre lanciati a ogni avvio del sistema.
16. Configurare i client.

Nel caso siano necessarie directory utente

1. Creare nel file di configurazione di Samba il blocco `homes`.
2. Impostare la direttiva `comment`.
3. Rendere le directory accessibili in scrittura utilizzando la direttiva `writable = YES`.
4. Inserire la direttiva `browsable = NO` per fare in modo che le condivisioni non siano visibili nei sistemi Windows.
5. Fare in modo che ogni utente possa accedere solo alla propria directory, sfruttando le variabili di Samba.
6. Verificare nel sistema Linux che a ogni utente corrisponda un profilo e una directory in `/home`.
7. Verificare che i proprietari e i permessi delle cartelle utente presenti nella diirectory `/home` siano corretti.
8. Se si vuole fare in modo che le directory utente non corrispondano alle directory utente di Linux, specificare nella direttiva `path` un percorso alternativo, all'interno del quale devono essere presenti le directory utente.
9. Verificare la correttezza della configurazione, utilizzando il comando `testparm`.
10. Riavviare i demoni di Samba.

Capitolo 2

Realizzazione di un dominio

Nel Capitolo 1 si è visto quanto sia semplice e veloce creare una rete locale attraverso la modalità di workgroup. Basta il pacchetto Samba su una macchina Linux e una semplice configurazione per avere un file server pronto per un numero arbitrario di client.
Le postazioni periferiche basate su Windows sono ancora più semplici da configurare. Basta fare in modo che il workgroup di riferimento coincida con quello impostato su Samba.
In un workgroup non esiste una vera distinzione tra client e server in quanto ogni macchina svolge entrambi i compiti. Qualunque postazione può mettere in condivisione proprie cartelle e diventare server per tali aree comuni. Allo stesso tempo queste macchine possono accedere a condivisioni di altri sistemi e agire come client.
Per questo motivo non è neppure necessario avere un computer server in un workgroup. Anche due computer Windows XP possono infatti costituire un workgroup a tutti gli effetti.
La facilità con cui si può creare e utilizzare un workgroup è il fattore chiave che ha permesso a Microsoft negli anni Novanta di battere importanti aziende nel settore del networking.
Microsoft non ha però limitato la propria presenza nel campo delle piccole reti ma ha piuttosto supportato attivamente una seconda modalità di rete più sicura e articolata, che si contrappone al workgroup e si chiama dominio.
In un dominio esiste almeno una macchina principale che svolge la funzione di PDC (*Primary Domain Controller*). Questa macchina svolge diverse funzioni chiave tra cui l'autenticazione degli utenti e la gestione dell'elenco degli utenti autorizzati a operare in rete.
In un workgroup non esiste niente di tutto questo. Ogni utente esegue un logon locale sul proprio computer e ogni macchina ha un proprio elenco privato di utenti che hanno il diritto di accedere alle risorse condivise localmente. Nessuna funzione è in questa modalità centralizzata.

In un dominio la situazione è ben diversa. Ogni utente che desidera aver accesso alla rete deve prima autenticarsi sul PDC, che contiene l'elenco di computer e utenti abilitati a operare in rete. Gli utenti che hanno fornito le credenziali corrette possono effettuare il logon, caricare dal server il proprio profilo personale e sfruttare risorse comuni.
Le macchine membro del dominio possono così accedere a directory condivise sul server ma anche condividere le cartelle presenti sul proprio sistema locale, come accadeva nel workgroup. La condivisione è regolata però da permessi più rigidi. Quando si esegue una condivisione viene caricata dal server l'elenco degli utenti abilitati al dominio e il proprietario del computer può attivare la condivisione a uno o più di tali utenti abilitati.
Il server è quindi sempre l'elemento centrale, anche quando si lavora con risorse locali.

Esempio pratico

In questo capitolo si prenderà in considerazione il caso di una piccola impresa di produzione che ha la necessità di creare un ambiente di rete ordinato.
L'azienda dispone di cinque postazioni in amministrazione (`amm1`, `amm2`, `amm3`, `amm4` e `amm5`), di dieci postazioni in ufficio tecnico (`ufftec1`, `ufftec2`, `ufftec3` ecc.), di due postazioni commerciali (`comm1` e `comm2`) e di un computer in direzione (`dir`). Tutte le postazioni sono basate su Windows 2000 Professional oppure Windows XP Professional.
Tutte le postazioni devono autenticarsi sul PDC. Durante il logon dovranno essere attivate sul client locale una serie di cartelle condivise quali area comune, area con i file di utilità, area di supporto per il gestionale e area riservata all'ufficio tecnico. Ogni utente dovrà inoltre vedere in *Risorse del computer* una propria cartella personale privata.
I profili personalizzati dei singoli computer, come lo sfondo, lo screen saver, la disposizione delle icone, le impostazioni del browser, le impostazioni del desktop, la configurazione di Office e così via, dovranno essere memorizzate sul server. Questa funzionalità si chiama "roaming profiles" (profili mobili) e permette agli utenti di avere il proprio ambiente di lavoro su qualunque macchina della rete. All'accesso nel dominio tutte le impostazioni personalizzate sono scaricate in locale e applicate sulla macchina. L'utente è libero di lavorare normalmente sul computer e all'uscita dal dominio tutte le modifiche apportate saranno salvate sul server.
I profili mobili sono molto comodi ma si limitano alle impostazioni personalizzate. Non vengono cioè condivisi i programmi. Office, per fare un esempio, dovrà essere regolarmente installato su tutte le postazioni in cui è necessaria la presenza della suite.
Durante la configurazione di Samba come PDC si daranno per scontati i concetti espressi nel Capitolo 1 e le configurazioni applicate in questo capi-

tolo sono in parte un'estensione della configurazione di base realizzata nel Capitolo 1.
Prima di tutto si deve bloccare il servizio Samba, rinominare il vecchio file di configurazione, salvarlo e creare un nuovo file `smb.conf`.
In apertura si deve scrivere la sezione `global`:

```
[global]
workgroup = INCIPIT
netbios name = SERVER1
server string = PDC Linux
security = USER
smb passwd file = /etc/samba/smbpasswd
encrypt passwords = YES
log file = /var/log/samba/%m.log
max log size = 100
log level = 1
```

Non ci sono modifiche sostanziali rispetto a quanto si era visto nel Capitolo 1. Cambia in questo caso il nome del server in rete (`netbios name`), il commento testuale (`server string`) e il nome del dominio che viene comunque scritto nella direttiva workgroup.
In questo tipo di configurazione è fondamentale che la sicurezza sia a livello utente (`security = USER`) e le password siano cifrate.
La configurazione della sezione `global` deve ora proseguire indicando l'intenzione di impostare il sistema Samba come PDC:

```
os level = 255
preferred master = YES
local master = YES
domain master = YES
wins support = YES
```

Il PDC di una rete è anche il sistema che svolge le funzionalità di Master Browser List. Si tratta cioè della macchina che si preoccupa di mantenere aggiornato l'elenco di sistemi che appare facendo clic su *Risorse di rete*. Questo elenco non è infatti dinamico o generato in tempo reale dai client. Quando in realtà si fa un doppio clic su *Risorse di rete* si accede al Master Browser List e si scarica l'elenco da questo sistema.
Master Browser List non è un computer pensato per essere statico, o configurato per esserlo. Tutti i sistemi presenti svolgono elezioni per decidere quale delle macchine presenti è più idonea per questo delicato compito. Vengono considerati molti criteri tra cui il tempo di uptime e il tipo di sistema operativo.
In Samba, come nel mondo reale, è possibile sembrare migliori di quello che si è e tentare così di vincere le elezioni. Con la direttiva `preferred master = YES` si forza un'elezione e tramite la direttiva `os level = 255` ci si pone in cima a tutte le preferenze. Questo valore indica infatti la tipologia del sistema operativo (i client hanno valori bassi mentre i server ne hanno

di più alti). Maggiore è il numero e maggiori sono le possibilità di vincita. Il valore di default è 20.
Con la direttiva `local master = YES` si comunica alla rete che il server Samba intende diventare il Master Browser List per il dominio `INCIPIT`. La direttiva seguente è molto simile ma opera a livello più ampio quando un dominio è sparso su più sottoreti. In queste situazioni potrebbero esserci diversi sistemi a gestire gli elenchi per le singole sottoreti. La direttiva `domain masters = yes` opera come collante e fa in modo che il server Samba riceva gli elenchi di tutte le sottoreti. Questi elenchi saranno unificati sul server Samba e il risultato complessivo sarà distribuito a tutte le sottoreti. Gli utenti, sfogliando *Risorse di rete*, potranno vedere un elenco che comprende tutti i sistemi presenti in tutti i segmenti della struttura della rete.
Ora che il server Samba è il Master Browser List del dominio `INCIPIT`, che gestisce gli elenchi locali, e per tutte le sottoreti, si può andare oltre e inserire una direttiva di compatibilità per le macchine Windows 95/98. Questi sistemi operativi non sono infatti in grado di accedere al dominio a tutti gli effetti per limiti progettuali. È quindi necessario utilizzare la seguente direttiva per attivare la compatibilità:

```
domain logons = YES
```

Il sistema Samba può agire anche da server WINS e fornire così servizi di risoluzione per i client Windows presenti. È sufficiente un'unica direttiva per attivare la funzionalità:

```
wins support = YES
```

Per accedere a questo server WINS bisogna andare sui singoli client e specificare l'indirizzo IP del server nell'apposita finestra di configurazione delle proprietà di rete.
Prima di attivare la funzionalità WINS bisogna controllare bene che non ve ne siano altri sulla sottorete locale in quanto si potrebbe avere comportamenti erronei.

Abilitazione delle directory utente

È possibile fare in modo che ogni utente della rete disponga di una propria directory personale privata. Questa cartella può essere assegnata in fase di logon a un lettera di unità ben precisa, per esempio `U`. Per attivare la funzionalità si utilizza la seguente sintassi:

```
logon home = \\server1\homedir
logon drive = U:
```

La prima direttiva esplicita che le cartelle utente si trovano nella condivisione `\\server1\homedir`. Attenzione al fatto che non si tratta di un percorso

all'interno del sistema Linux ma appunto di un percorso di rete. In tal caso si fa riferimento a una condivisione chiamata `homedir` presente sulla macchina Linux locale `server1`. La cartella utente sarà unita alla lettera `U` sul client locale grazie alla seconda direttiva.
La condivisione è definita più in basso sul file di configurazione, in un apposito blocco:

```
[homedir]
path = /home/%u
read only = NO
writable = YES
browsable = NO
create mask = 0600
directory mask = 0700
hide dot files = YES
```

La configurazione del blocco non si discosta molto da quanto si è visto nel Capitolo 1. Ci sono comunque alcuni punti su cui è interessante soffermarsi. Prima di tutto, il percorso è specificato tramite una variabile. Questo fa in modo che l'utente `amm2` abbia come cartella utente il percorso `/home/amm2`, l'utente `amm4` avrà `/home/amm4` e così via.
Come si può notare si è scelto di utilizzare le cartelle utente di Linux e avere in questo modo un ambiente omogeneo dove gli utenti vedono la stessa directory home sia su Linux, sia su Windows.
Ci sono anche alcune direttive che non si erano viste nel Capitolo 1. La prima di queste è `create mask`, che fa in modo che tutti i file creati sulla condivisione acquisiscano i diritti `0600`. La direttiva successiva svolge la stessa funzionalità ma questa volta sulle directory create.
La direttiva `hide dot files = YES` forza il bit nascosto su tutti i file che iniziano per punto. Questo è utile per fare in modo che gli utenti Windows non vedano i file di configurazione di Linux presenti nella directory `/home`, come `.bashrc`, `.bash_profile` e `.bash_logout`.
La direttiva rende i file nascosti, ma se l'utente ha attivato su Windows la visualizzazione dei file nascosti vedrà tutti i file che iniziano per punto.

Profili mobili (roaming profiles)

I profili mobili (*roaming profiles*) permettono di salvare i profili di tutti gli utenti della rete sul server e fare in modo che ci si possa spostare di postazione e disporre sempre del proprio ambiente di lavoro tra cui la cartella *Documenti*, i preferiti del browser, il desktop, l'elenco dei file recenti, la struttura del *menu Avvio*, la propria posta elettronica ecc.
Oltre alla comodità di poter avere la propria scrivania su qualunque computer della rete si ha anche una funzione di backup sul server di molte informazioni che generalmente sono salvate solo localmente. Si pensi per esempio alla posta di Outlook Express. Sono poche le realtà che eseguono un

backup di queste informazioni tipicamente locali. In caso di rottura del disco o di danno al file system si perde tutta la base di messaggi archiviata nel corso degli anni.
Lo svantaggio maggiore comportato dai roaming profiles consiste nel carico di rete. Ogni volta che si entra o si esce avviene un'operazione di sincronizzazione con il server e questo comporta un utilizzo della banda locale. Se le configurazioni locali sono molto complesse, nel caso per esempio in cui si abbiano archiviati decine di megabyte di messaggi o di documenti si sperimenterà una lentezza rilevabile. Se questo aspetto può risultare problematico è possibile omettere la configurazione di questa sezione ed evitare l'uso dei profili mobili.

```
logon path = \\server1\profili\%u
```

Questa direttiva attiva i profili mobili segnalando a Samba che la directory dove memorizzare i profili si trova nella condivisione `\profili\%u` di `server1`. Ancora una volta non si tratta di un riferimento al file system di Linux ma di una condivisione.
Viene anche in questo caso impiegata una variabile per fare in modo che sia utilizzata una directory separata per ogni utente.
La condivisione `profili` è definita più in basso, in un apposito blocco:

```
[profili]
path = /usr/local/samba/profili
read only = NO
writable = YES
browsable = NO
create mask = 0600
directory mask = 0700
```

È da segnalare il fatto che si è scelto di rendere questa condivisione non visibile da *Risorse di rete*, utilizzando la direttiva `browsable = NO`.
Bisogna poi prestare attenzione ai diritti impostati sulla directory `/usr/local/samba/profili`; tutti gli utenti devono potervi accedere, leggerla e scrivervi mediante il comando `chmod`:

```
chmod 777 /usr/local/samba profili
```

Windows 95 e 98 utilizzano un meccanismo differente per gestire i profili mobili. In tal caso non viene utilizzata la direttiva `logon path` ma la direttiva `logon home` per specificare la cartella utente. La documentazione ufficiale di Samba consiglia di specificare questa sintassi nel caso di Windows 95 e 98:

```
logon home = \\server1\%U\profile
```

Samba gestirà questo percorso in maniera dinamica. Nel caso cioè di assegnamento di una lettera alla home directory verrà considerata solo la porzione `\\server1\%U`. Nel caso invece di utilizzo dei profili mobili sarà usato il percorso completo e il profilo sarà memorizzato dentro la directory `profile`.

Script di logon

Quando il client accede al PDC può ricevere automaticamente dal server un file batch con una serie di comandi DOS. In questo batch possono essere presenti diverse attività come la sincronizzazione dell'orologio del computer locale con il server e l'attivazione di un certo numero di condivisioni.
Si tratta di un buon sistema per rendere omogeneo l'ambiente di rete ed evitare di dover visitare tutte le postazioni ogni volta che si crea una nuova condivisione aziendale. Basta infatti aggiungere una riga al batch e in questo modo tutte le macchine attiveranno le nuove condivisioni all'ingresso.
Per attivare gli script di logon basta indicare la direttiva seguente:

```
logon script = logon.bat
```

Durante il logon, il file `logon.bat` sarà automaticamente scaricato dal server ed eseguito. Non viene indicato alcun percorso per individuare il file in quanto esiste una condivisione di Windows dedicata a questo scopo. Si tratta di `netlogon`, che viene creata di default sui PC Windows: su Samba, invece, è necessario replicare tale comportamento creando un opportuno blocco di definizione:

```
[netlogon]
path = /usr/local/samba/netlogon
read only = YES
write list = root
```

La directory è a sola lettura e solo l'utente root ha la facoltà di scrivere in questa condivisione (direttiva `write list = root`). La directory è visibile da *Risorse di rete*, come avviene con i PDC basati su Windows.
Dentro la condivisione `netlogon` è salvato il file `logon.bat`, che deve essere scritto su una macchina Windows e poi salvato su Linux in quanto questi due sistemi operativi gestiscono in maniera differente gli a capo. Un batch scritto su Linux potrebbe avere problemi di funzionamento sulle macchine Windows.
Un file batch di logon potrebbe avere la fisionomia seguente:

```
NET TIME \\server1 /SET /YES
NET USE G: \\server1\comune
NET USE M: \\server1\gestionale
```

La prima riga sincronizza l'ora locale con l'orario del server. Le righe seguenti si limitano invece ad agganciare alcune condivisioni a due lettere di unità. Tutte le macchine vedranno così l'area comune e l'area gestionale come `G:` e `M:`.

Naturalmente è necessario scrivere due blocchi con le relative definizioni delle condivisioni:

```
[comune]
comment = cartella comune
path =/home/comune
public = YES
writable = YES

[gestionale]
comment = area supporto software gestionale
path =/home/gestionale
public = YES
writable = YES
```

Bisogna stare attenti se si decide di limitare alcune condivisioni; per esempio fare in modo che solo le postazioni in amministrazione possano vedere l'area gestionale. In fase di logon gli utenti dell'ufficio tecnico vedrebbero un messaggio di errore in quanto lo script di logon tenterebbe di agganciarsi a una directory a cui l'utente non ha accesso.
Per risolvere questi problemi bisogna ricorrere a programmi di scripting più complessi dotati di clausole condizionali. In questo modo è possibile specificare per esempio che se l'utente fa parte del gruppo UT deve vedere la condivisione `ufftec` ma non l'area gestionale mentre, viceversa, gli utenti del gruppo `Amministrazione` vedranno la condivisione gestionale e non `ufftec`. Kixtart (`www.kixtart.org`) è una soluzione di scripting ottima e molto ben documentata.
Se non si vuole qualcosa di così complesso, si può semplicemente avere un file batch per ogni utente. Ogni file di login sarà in questo modo dedicato e conterrà solo le condivisioni rilevanti per l'utente in oggetto. Si devono in questo caso creare tanti file batch quanti sono gli utenti e salvarli con il nome dell'utente seguito dall'estensione `.bat`, per esempio `amm1.bat`, `amm2.bat`, `ufftec1.bat`, `dir.bat` ecc.
Per fare in modo che Samba scarichi il file corretto si deve usare una direttiva dotata di variabile, simile a quella riportata di seguito:

```
logon script = %u.bat
```

Attraverso la variabile `%u` sarà composto dinamicamente il nome del batch da caricare dalla condivisione `netlogon`.

Abilitazione di utenti e computer

La configurazione di Samba può considerarsi conclusa. Questo è l'elenco completo delle operazioni eseguite sul file `smb.conf`:

```
[global]
workgroup = INCIPIT
```

```
netbios name = SERVER1
server string = PDC Linux
security = USER
smb passwd file = /etc/samba/smbpasswd
encrypt passwords = YES
log file = /var/log/samba/%m.log
max log size = 100
log level = 1

# impostazione del server come domain master browser
os level = 255
preferred master = YES
local master = YES
domain master = YES
wins support = YES

# abilitazione dei logon W95/W98
domain logons = YES

# attivazione supporto WINS
wins support = YES

# impostazione homedir
logon home = \\server1\homedir
logon drive = U:

# impostazione profili mobili
logon path = \\server1\profili\%u

# impostazione script di logon
logon script = logon.bat
#logon script = %u.bat

[netlogon]
path = /usr/local/samba/netlogon
read only = YES
write list = root

[profili]
path = /usr/local/samba/profili
read only = NO
writable = YES
browsable = NO
create mask = 0600
directory mask = 0700

[homedir]
path = /home/%u
read only = NO
writable = YES
browsable = NO
create mask = 0600
directory mask = 0700
hide dot files = YES
```

```
[comune]
comment = cartella comune
path =/home/comune
public = YES
writable = YES

[gestionale]
comment = area supporto software gestionale
path =/home/gestionale
public = YES
writable = YES

[software]
comment = file di utilita'
path =/home/software
public = YES
writable = NO
```

Come per la configurazione di un workgroup è necessario creare sia gli utenti abilitati al dominio all'interno di Linux sia un utente con lo stesso nome sul sistema Samba tramite l'utility smbpasswd.
Su un dominio però anche i computer devono essere autenticati per poter accedere alla rete. Tale scelta permette una maggiore sicurezza in quanto ogni singola macchina è dotata di una chiave di protezione univoca. Questo impedisce che qualcuno possa accedere al dominio cambiando nome alla propria macchina e fingendosi una macchina abilitata.
L'abilitazione di un computer è una procedura molto simile a quella eseguita per abilitare gli utenti. Prima di tutto è necessario creare un utente sul sistema Linux con lo stesso nome della macchina e digitando questa particolare sintassi:

```
useradd -g domaincomputers -d /dev/null -s /bin/false nomemacchina$
```

Il parametro `-g` indica che il nuovo utente apparterrà al gruppo domaincomputers. Questo gruppo dovrà essere stato preventivamente creato tramite `groupadd domaincomputers`.
Il parametro `-d` specifica la directory home su Linux. In questo caso non ne serve nessuna, perché non si sta creando un utente ma piuttosto un profilo per un computer. Tale profilo non accederà mai al sistema Linux dalla shell interattiva. Si specifica che la directory si trova nella directory `dev/null`, una sorta di sinonimo per indicare nessuna posizione.
Il parametro `-s` indica la shell di login per l'utente. Anche in questo caso non ne serve alcuna perché si tratta di un computer e non di un utente. Si specifica allora l'opzione `false`. Questo non svolge alcuna attività e si limita a uscire appena viene invocato.
Si ha infine il nome della macchina che si sta abilitando, seguito dal simbolo obbligatorio di `$` (convenzione NetBIOS).

Il nome della macchina deve essere ricavato da Windows XP trascinando il mouse sopra *Risorse del computer*, facendo clic sul tasto destro e selezionando la voce *Proprietà*. Dalla finestra relativa si deve fare clic sulla scheda *Nome computer* (Figura 2.1). Il nome della macchina è indicato in *Nome completo computer*. È fondamentale che sia il nome su Windows XP sia quello su Samba coincidano.

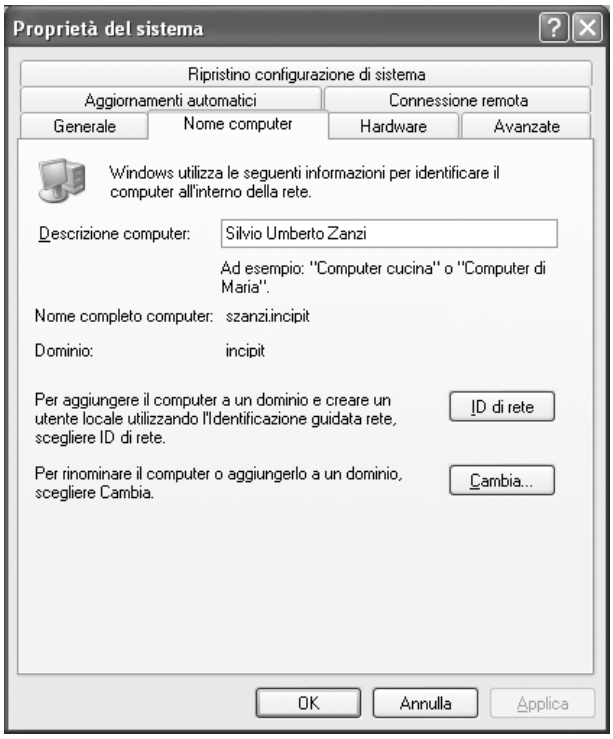

Figura 2.1
La scheda Nome computer contiene informazioni per l'identificazione del pc all'interno della rete.

Ora bisogna utilizzare il comando `smbpasswd` e creare un riferimento alla macchina sul file delle password di Samba:

```
smbpasswd -a -m nomemacchina
```

Il parametro `-a` indica che si tratta di un nuovo riferimento e che il nome deve essere aggiunto al file delle password di Samba. Il parametro `-m` indica che si sta aggiungendo l'account di un computer e non di un utente, ed è seguito dal nome del computer, questa volta senza il simbolo di `$`. Premendo il tasto Invio della tastiera si procede alla registrazione della entry nel file delle password di Samba. Se ora si accede alla directory `/etc/samba` e si apre il file `passwd`, in fondo a esso si potrà notare l'account creato.

Durante le operazioni di abilitazione delle macchine è importante evitare che i nomi dei computer coincidano con i nomi degli utenti abilitati in Samba.

Unione del client al dominio

Il sistema è pronto: non resta che unire la macchina al dominio. Si deve nuovamente trascinare il mouse sopra *Risorse del computer*, fare clic con il tasto destro del mouse e selezionare la voce *Proprietà*. Dalla finestra che si apre si dovrà fare clic sulla scheda *Nome computer* e premere il pulsante *ID di rete*, per attivare la procedura guidata di annessione al dominio.
Al primo passaggio, nella finestra *Identificazione guidata rete* bisogna fare clic nella casella di selezione presente in alto per indicare che il computer fa parte di una rete aziendale (Figura 2.2).

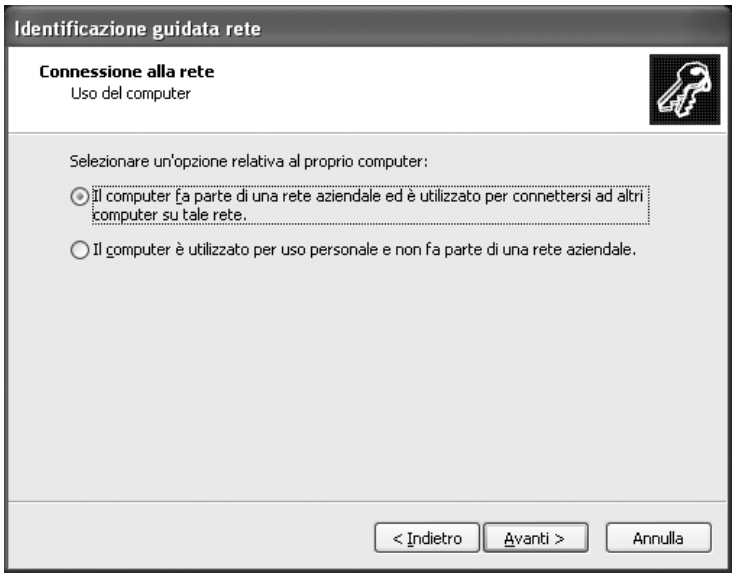

Figura 2.2
Nell'Identificazione guidata rete, prima di tutto, si deve deve indicare come si utilizza il computer.

Nel passaggio seguente bisogna ancora selezionare la voce in alto per indicare che la propria azienda utilizza un dominio (Figura 2.3).
Al passaggio seguente si deve indicare un nome utente abilitato in Samba (per esempio `amm1`), la password relativa e il dominio (`INCIPIT`), come mostra la Figura 2.4.
Confermando potrebbe essere necessario indicare manualmente il nome del computer e il dominio di appartenenza (Figura 2.5).

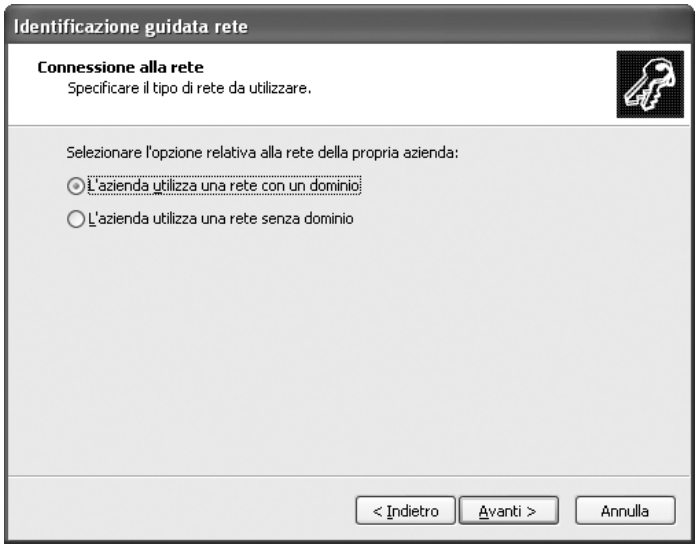

Figura 2.3
Successivamente, bisogna specificare il tipo di rete da utilizzare.

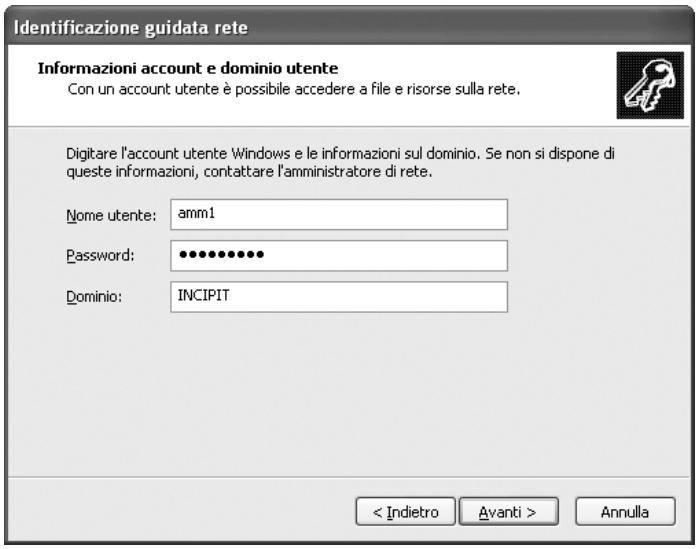

Figura 2.4
Vanno poi fornite informazioni relative ad account e dominio utente.

Bisogna infine indicare un account di amministrazione con autorizzazione di accesso al dominio (Figura 2.6). In questo passaggio bisogna indicare l'utente root di Samba, la relativa password e nuovamente il dominio.
Attenzione a non fare confusione perché si deve usare la password per l'utente root specificata in Samba, non quella di Linux. Se si è dimenticata la

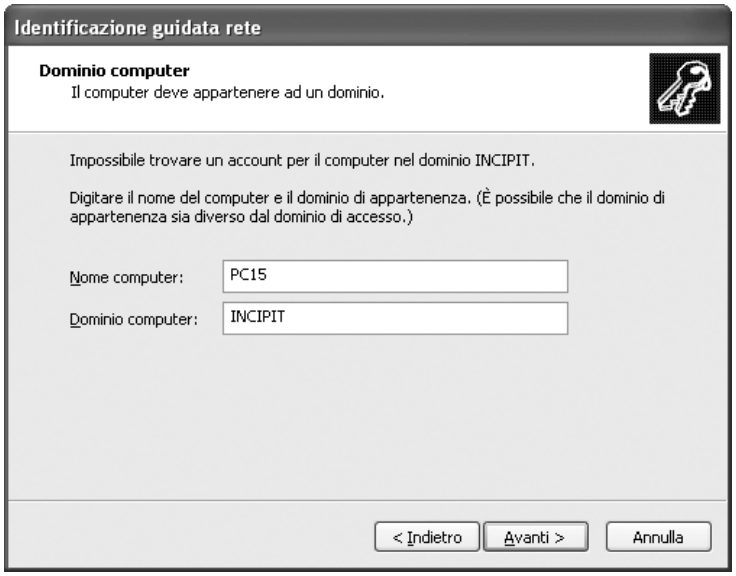

Figura 2.5
È possibile che venga richiesto di digitare il nome del computer e il suo dominio di appartenenza.

Figura 2.6
Infine, bisogna inserire il nome di un account che ha pieni poteri sul dominio.

parola chiave si può usare nuovamente il comando `smbpasswd -a root` per sostituire la vecchia password di Samba con una nuova.
Premendo il tasto Invio della tastiera si conclude la procedura. Le modifiche non vengono comunque applicate subito in quanto è necessario eseguire un riavvio del computer.
All'accesso si dovrà digitare il dominio `INCIPIT` nella finestra di logon, inserire il nome utente e la password: quest'ultima sarà verificata sul server e, in caso di successo, verrà creato il profilo mobile, sarà collegata la directory home e lanciato lo script di login. Si farà a questo punto parte del dominio. Si potrà immediatamente notare il cambiamento di gestione della protezio-

ne, cercando di condividere una cartella: sarà infatti richiesto a quali degli utenti presenti sul server concedere l'accesso alla cartella in oggetto.
Nel caso non sia possibile condividere la cartella, o non si abbiano i privilegi per eseguire le configurazioni sulla macchina locale, bisognerà ricordare che le macchine Windows collegate a un dominio hanno due profili: uno locale e uno di rete.
Il profilo di rete è quello su cui si è lavorato nel corso di questo capitolo. In sostanza è il profilo utente creato sul server e gestito completamente dal PDC. Il profilo locale, invece, regola l'accesso alle entità del sistema locale come le configurazioni del computer locale, le directory, i file e le stampanti presenti.
Il nuovo utente creato potrebbe non avere un profilo sufficientemente alto sulla macchina locale ed essere un semplice utente. Tale soluzione può andare bene in un ambiente di lavoro in quanto impedisce che gli utenti possano riconfigurare la macchina. Se si desidera però dare pieno accesso, bisogna allora entrare come administrator specificando come dominio il nome della macchina. Si deve andare nel *Pannello di controllo*, fare clic sull'icona *Account utente* e aggiungere il nuovo utente locale. Bisogna specificare il nome utente (lo stesso di quello di dominio, per esempio `amm3`), il dominio e confermare tutto. Sulla finestra seguente bisogna specificare che si vuole il livello *Administrator locale*.
Ora si può rientrare nel dominio e l'utente avrà piena libertà sulla macchina locale. Sul dominio resteranno validi i diritti impostati nel server.

Checklist

1. Verificare che il pacchetto Samba sia installato nel proprio sistema, digitando il comando `testparm`: se compare un messaggio di errore significa che Samba non è presente e bisogna procedere alla sua installazione, dai CD-ROM del sistema operativo oppure attraverso risorse online.
2. Salvare in un luogo sicuro il file di configurazione di default di Samba, quindi creare un nuovo file `smb.conf` nella directory `/etc/samba`.
3. Aprire il file `smb.conf` e creare la sezione `global`.
4. Inserire nella sezione `global` le direttive `workgroup`, `netbios name` e `server string`.
5. Impostare nella sezione `global` il livello di sicurezza dellutente, aggiungendo la direttiva `security = USER`.
6. Impostare il file delle password tramite la direttiva `smb passwd file`.
7. Indicare la gestione delle password cifrate tramite la direttiva `encrypt passwords = YES`.

8. Inserire le direttive necessarie per specificare le funzionalità di log.
9. Impostare la macchina Samba come Master Browser List e come Domain Master, utilizzando le direttive `os level`, `preferred master`, `local master` e `domain master`.
10. Attivare il servizio WINS di Samba con la direttiva `wins support = YES`.
11. Aggiungere la direttiva `domain logos = YES` solo nel caso in cui sia necessario unire al dominio sistemi Windows 95/98/ME.
12. Abilitare le directory utente, se necessario.
13. Abilitare i roaming pprofiles, se necessario.
14. Abilitare lo script di logon e creare su una macchina Windows il file relativo batch, per evitare i problemi di compatibilità tra Linux e Windows nella gestione degli a capo.
15. Impostare una condivisione, creando nel file di configurazione una sezione che deve contenere almeno le direttive `comment` e `path`.
16. Indicare l'elenco degli utenti autorizzati ad accedere alla condivisione, con la direttiva `valid users`.
17. Specificare gli eventuali utenti non autorizzati, utilizzando la direttiva `invalid users`.
18. Inserire la direttiva `writable = YES` per rendere accessibile in scrittura la condivisione.
19. Impostare correttamente i permessi per directory della condivisione.
20. Ripetere i passaggi da 15 al 19 per tutte le condivisioni necessarie.
21. Creare sul sistema Linux gli account degli utenti Windows che potranno accedere alle condivisioni.
22. Creare le password di Samba per gli account appena creati; a tal proposito si utilizza il comando `smbpasswd`.
23. Abilitare i computer che potranno accedere al dominio di Samba. Si deve creare sulla macchina Linux un utente con lo stesso nome del computer, poi bisogna impostare una password utilizzando il comando `smbpassswd`.
24. Impostare le direttive di sicurezza per il sistema.
25. Verificare la correttezza della configurazione utilizzando il comando `testparm`.
26. Avviare i demoni di Samba e fare in modo che siano sempre lanciati a ogni avvio del sistema.
27. Configurare i client per unire la macchina al dominio.

Capitolo 3

Creazione di un print server

Un *print server* è un elemento preposto ad accogliere le richieste di stampa inoltrate dai client presenti in rete e di indirizzarle a una o più stampanti fisicamente collegate all'unità. Il print server permette in sostanza di condividere in rete una stampante che originariamente è nata come unità locale da collegarsi tramite la porta parallela o USB.
Si tratta di una funzionalità utile e molto apprezzata dalle aziende, soprattutto quelle più piccole, dal momento che le stampanti dotate di porte di rete Ethernet hanno costi sensibilmente più alti. Per contro è molto semplice ed economico realizzare un print server attraverso Linux e risolvere questa necessità di condivisione.
Come prerequisito si considererà un sistema Linux dotato di Samba e configurato in modalità di workgroup o di dominio a seconda del tipo di rete che si sta utilizzando. A tal proposito si daranno per scontati i concetti e le tecniche messe in atto nei Capitoli 1 e 2. Questa base sarà arricchita con i blocchi e le direttive necessarie per la gestione della stampa su Samba.
Il primo punto da tenere ben presente è che Samba è preposto unicamente alla gestione della condivisione della stampante e della relativa coda di stampa in rete. Il pacchetto non ha la capacità di interagire direttamente con la stampante o di eseguire alcun tipo di elaborazione sui dati in stampa. Queste operazioni sono invece svolte da un meccanismo software specifico denominato "sistema di stampa". Samba si appoggia completamente a questo strato, inoltrando il flusso di dati ricevuti dai client presenti in rete. Il sistema di stampa provvederà alla comunicazione con la stampante e alla stampa fisica dei lavori.
Questo sistema è un elemento cruciale e prima di eseguire qualunque configurazione su Samba è necessario dare uno sguardo a questo elemento.

Configurazione di sistemi di stampa diversi

La stampa è una delle aree in cui Linux è stato storicamente considerato debole. Fino a qualche tempo fa si aveva infatti un supporto limitato, riservato a stampanti per i soli testi o a costose unità dotate di una scheda interprete per il Postscript. Non si poteva in questo scenario pensare di andare dal proprio negozio di fiducia e comprare una stampante scegliendola solamente in base alle proprie esigenze di velocità, risoluzione, qualità e costo di esercizio. Nella maggior parte dei casi questa stampante non avrebbe funzionato su Linux (a meno che non fosse stata un'unità specifica per il testo o un sistema dotato di PostScript).
Per questo motivo la stampa di lavori complessi su Linux era qualcosa di riservato alle organizzazioni più facoltose, in grado di permettersi stampanti laser di buon livello.
Ora la situazione è ben diversa. Da parecchio tempo esiste un meccanismo di stampa aperto denominato CUPS (*Common Unix Printing System*) e disponibile all'indirizzo `www.cups.org`. CUPS fornisce un supporto completo per la stampa, permettendo a qualunque costruttore di creare driver specifici per le proprie stampanti. Linux diventa in questo modo del tutto simile a Windows, sistema operativo che ha da sempre un sistema di stampa univoco, articolato e documentato.
Qualunque applicazione può agganciarsi a CUPS e usarlo per trasportare su carta qualsiasi tipo di documento elaborato sui programmi applicativi.
CUPS non è l'unico sistema di stampa disponibile per Linux ma è quello che ha certamente maggiore successo e diffusione. Gran parte delle distribuzioni integra infatti questo meccanismo in maniera nativa.
Il successo è dato in buona misura dalla scelta di aderire al protocollo IPP. Si tratta di un standard promosso da IEEE per la stampa aperta in ambienti di rete. Samba supporta il sistema di stampa CUPS e per questo motivo sarà utilizzato nella configurazione del print server.

Configurazione della stampante con CUPS

La configurazione del sistema CUPS sarà in questo caso molto semplice da realizzarsi perché il sistema di stampa su Linux non dovrà gestire alcuna attività di traduzione o di conversione dei flussi. La procedura di stampa avverrà infatti sulle macchine Windows usando il driver specifico per la stampante condivisa. La macchina Linux riceverà quindi dalla rete un flusso di stampa già elaborato e convertito per essere utilizzato dalla stampante collegata.
Il sistema CUPS dovrà semplicemente prendere questo flusso di dati e dirottarli alla porta fisica dove è presente l'unità, poi la stampante eseguirà la stampa autonomamente su carta.

Per cominciare si deve verificare che il sistema CUPS sia presente sul sistema Linux. Per farlo si può utilizzare il sistema di gestione dei pacchetti e verificare la presenza del pacchetto. Su un sistema Red Hat Fedora si può impartire la sintassi seguente:

```
rpm -q cups
```

Se il pacchetto è presente dovrebbe comparire una riga di indicazione.
Nel caso CUPS non sia installato si deve procedere alla sua installazione attraverso meccanismi messi a disposizione dalla propria distribuzione. Si tratta comunque di una situazione poco comune nelle distribuzioni presenti oggi sulla scena: tutte le installazioni di default, anche quelle minime includono CUPS nell'elenco dei pacchetti base.
Verificata la presenza del sistema di stampa si può procedere alla sua configurazione. Si prenderà come esempio il caso di un ufficio tecnico di una piccola azienda di elettronica. In questa realtà ci sono cinque programmatori e tutti hanno bisogno di stampare. Non sussiste però l'esigenza di fornire una stampante per ogni utente in quanto i volumi personali sono molto limitati. Si è perciò optato di condividere una stampante HP Laserjet 1100 acquistata qualche tempo prima in rete.
Si comincia accedendo agli strumenti di amministrazione forniti dal sistema di stampa. Nel caso specifico di CUPS si ha un pannello web per la configurazione intuitiva della stampante.
Dalla propria macchina si deve digitare l'indirizzo web `http://127.0.0.1:631`. CUPS risponde con il pannello principale di gestione e configurazione del sistema (Figura 3.1).
Si deve scegliere *Manage Printers* e poi *Add Printer*. Potrebbe in tal caso essere necessario immettere le credenziali di accesso, tipicamente l'utente root e la password a esso relativa.
Nella riga *Name* si deve inserire il nome della coda di stampa. Deve essere un stringa corta e senza spazi, per esempio `HPLJ1100`. In *Location* e in *Description* vanno inseriti testi liberi per indicare il luogo in cui si trova l'unità (per esempio *Ufficio tecnico*) e la descrizione estesa dell'unità (per esempio `HP Laserjet 1100`). Inserite queste informazioni si fa clic su *Continue*.
Nella sezione seguente si deve indicare dove si trova fisicamente la stampante, in questo caso la porta parallela locale del computer. Poi si fa clic su *Continue*.
Il passaggio seguente è estremamente importante in quanto stabilisce il tipo di elaborazione che dovranno subire i lavori indirizzati a questa porta di stampa. Come si è puntualizzato precedentemente, non serve alcuna elaborazione. Sono i driver presenti sulle singole macchine Windows a realizzare la conversione del lavoro in un flusso di dati compatibile con la stampante utilizzata. Per questo motivo bisogna tassativamente scegliere la modalità *Raw*, che non applica alcuna elaborazione e si limita a inoltrare il job di stampa alla stampante. Successivamente si fa clic su *Continue*.

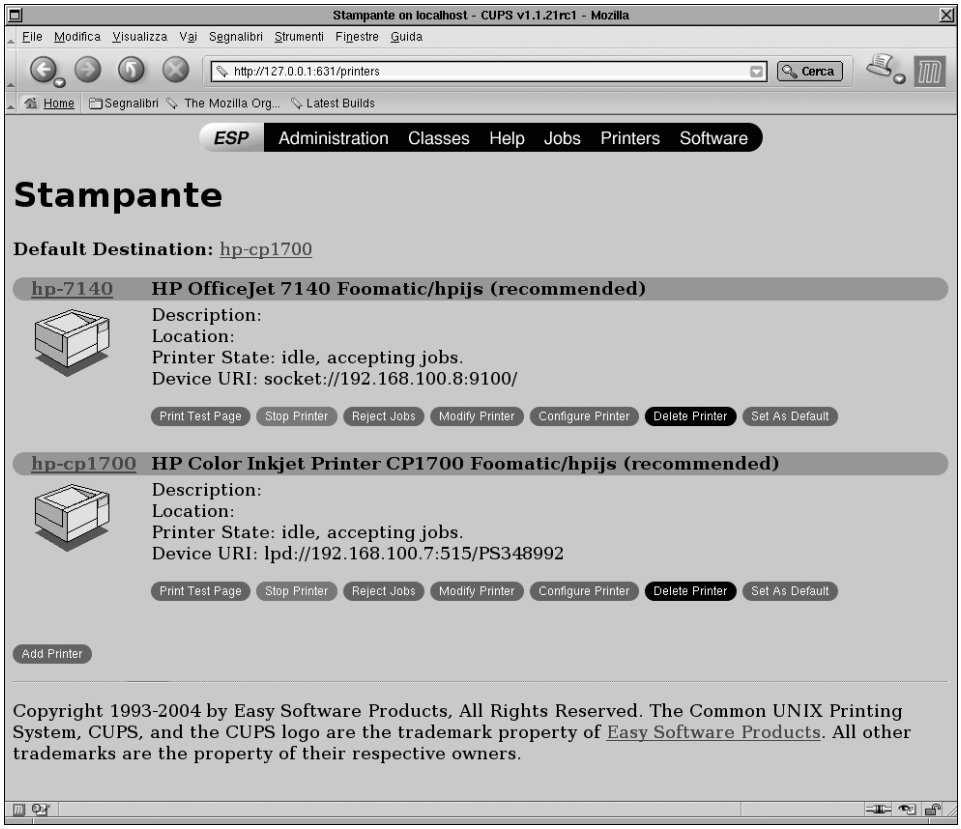

Figura 3.1
Configurazione della stampante con CUPS.

Avendo scelto *Raw* si avrà un unico modello, ancora una volta di tipo Raw. Basta semplicemente confermare la scelta.
Ora la stampante di tipo *Raw* è presente nel sistema ed è pronta per essere utilizzata da qualunque applicazione, compreso Samba.

Configurazione della stampante in Red Hat

Se si utilizza una distribuzione Red Hat si può utilizzare un metodo testuale, per configurare le code di stampa su CUPS: digitando `setup` dalla linea di comando e scegliere la voce `Printer configuration`. verrà caricato un tool specifico per la creazione di code di stampa.
Questo strumento non carica le stamparti configurate direttamente su CUPS ma solo quelle configurate tramite lo strumento stesso. Se si è già configurata una coda tramite il pannello web è normale non vedere nessuna stampante nel pannello principale del programma.

Si comincia facendo clic su *New*. Una finestra richiederà il nome della coda e il tipo di coda che si vuole configurare, in questo caso *Local Printer Device* per indicare una stampante locale.
Quando si fa clic su *Next* verrà richiesta la periferica (*device*) Linux dove è fisicamente connessa la stampante. Nel caso della prima porta parallela si avrà `/dev/lp0`. La seconda parallela sarà `/dev/lp1` e così di seguito.
Confermando la scelta si deve indicare il tipo di driver. Come già spiegato si deve scegliere la modalità *Raw Print Queue*, presente in cima all'elenco. Poi si fa clic ancora su *Next*. Comparirà una finestra riassuntiva. Se i dati sono corretti si può selezionare *Finish* e concludere l'operazione.
Dalla finestra di configurazione (Figura 3.2) si deve selezionare *Exit*. Verrà chiesto se si vuole salvare stabilmente la configurazione appena creata. Facendo clic su *Yes* si conferma e si ritorna al menu principale dell'applicazione di setup. Infine, si fa clic su *quit*.

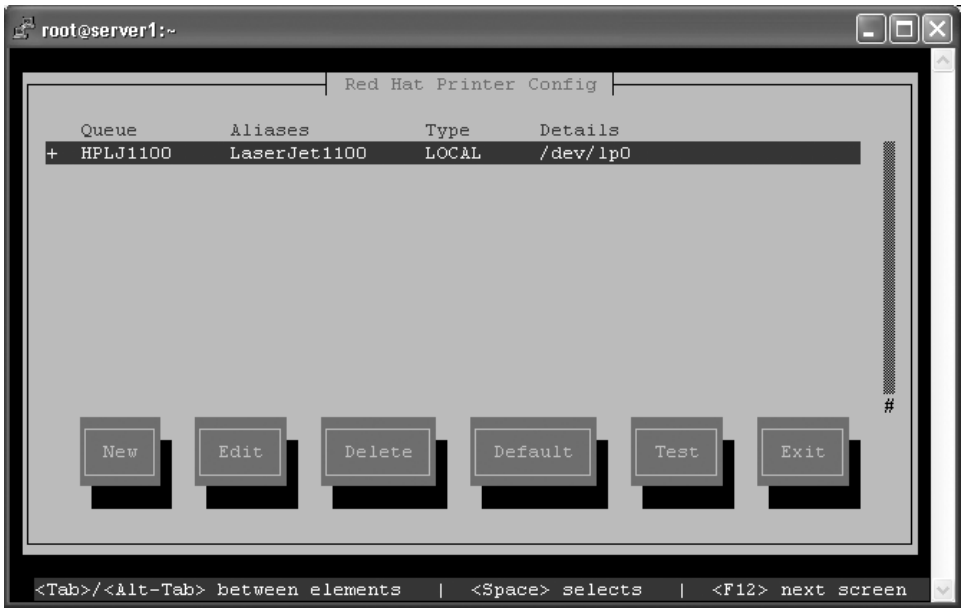

Figura 3.2
Configurazione della stampante in Red Hat.

Ritoccare la configurazione di CUPS

Avendo scelto la modalità *Raw* è necessario eseguire alcune configurazioni manuali nei file di CUPS.
Bisogna andare dentro `/etc/cups` e aprire il file `mime.types`. In fondo c'è una riga commentata che inizia con `#application/octet`. Bisogna cancellare il carattere `#` per togliere il commento e salvare.

Bisogna poi aprire il file `mime.convs` e togliere nuovamente il commento per la riga in fondo che inizia con `#application/octet`. Anche questo file va salvato.
Non resta che riavviare il sistema CUPS:

```
/etc/init.d/cups restart
```

Le impostazioni appena implementate permettono ai job in formato *Raw* di essere gestiti da CUPS.

Configurazione della stampante in SAMBA

Ora è il momento di configurare Samba.
In questo caso non si realizza una nuova configurazione ma si continua da una configurazione di rete già funzionante. Nell'eventualità che non se ne abbia una bisogna creare una configurazione per un workgroup o un dominio seguendo le indicazioni presenti nei Capitoli 1 e 2.
Prima di tutto bisogna indicare a Samba che si sta utilizzando il sistema CUPS per la stampa. Questo dettaglio va indicato nella sezione `global` tramite le due direttive seguenti:

```
printing = CUPS
printcap = CUPS
```

Ora bisogna creare una sezione denominata `printers` e specificare alcuni aspetti relativi alla stampa:

```
[printers]
comment = stampanti sul server
path = /var/spool/samba
printable = YES
use client driver = YES
```

Il nome della sezione, ovvero `printers`, è obbligatorio e indica a Samba che si stanno specificando informazioni relative alle stampanti condivise.
La riga `comment` contiene un'indicazione a testo libero per le stampanti condivise.
La direttiva `path` si riferisce al percorso sul file system del server Linux dove sarà eseguito lo spooling dei job di stampa. I dati in arrivo dai client e destinati alla stampa saranno raccolti in questo punto ma non saranno inoltrati direttamente alla stampante. Infatti, CUPS utilizza di default una propria area di spooling, generalmente `/var/spool/cups`. Ogni job di stampa transiterà quindi in due directory di spooling prima di giungere alla stampante.
La direttiva `printable = YES` attiva la possibilità di inoltrare job di stampa in spooling. Questo farà in modo che i client possano scrivere sullo spooler.
L'ultima direttiva, `use client driver = YES` serve per attivare la giusta modalità di stampa da parte dei client NT/2000/XP. Senza questa direttiva i si-

stemi suddetti tratterebbero la coda di stampa erroneamente e cercherebbero di accedere allo spooler di Samba tramite la user id e la password di accesso a Windows. Se queste stesse credenziali non sono presenti anche sulla macchina Linux con diritti di root si otterrebbe un messaggio di errore per accesso negato. La direttiva impedisce questo tipo di comportamento.
Si può a questo punto salvare il file di configurazione di Samba e riavviare il servizio. Samba caricherà automaticamente le stampanti precedentemente configurate e le farà comparire in *Risorse di rete* (Figura 3.3).

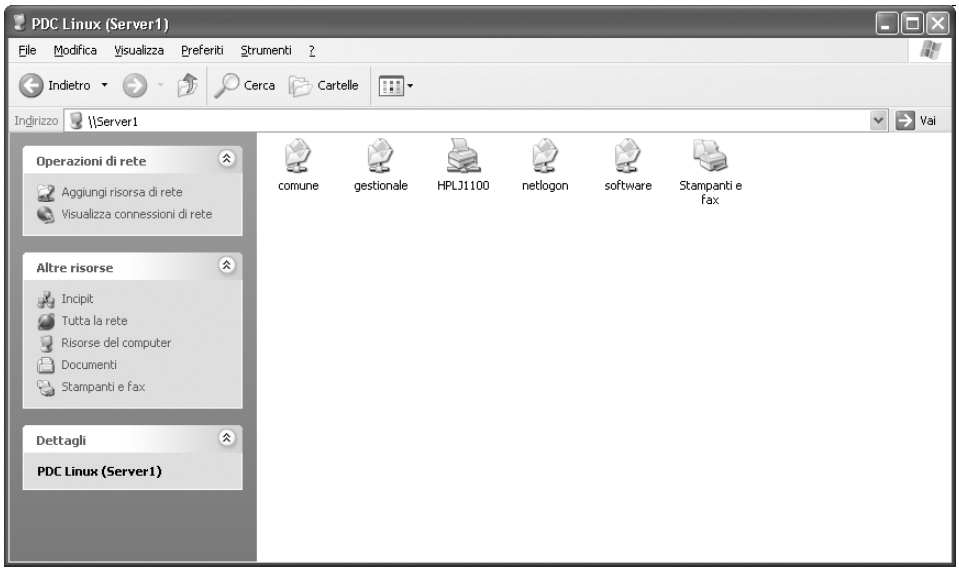

Figura 3.3
Nei sistemi Windows, le stampanti configurate appaiono in Risorse di rete.

Bisogna tenere ben presente che se si eseguono modifiche alle configurazioni delle stampanti tramite il pannello di amministrazione di CUPS o attraverso le utilità messe a disposizione dalla distribuzione bisogna riavviare il servizio Samba per avere le modifiche in linea.

Installazione dei driver sui client Windows

Il lavoro è praticamente concluso e non resta che caricare i driver di stampa sui client Windows.
Si deve andare in *Pannello di controllo*, selezionare l'icona *Stampanti* e *fax* e fare clic su *Aggiungi stampante*.
Al primo passaggio bisogna specificare che si sta installando una stampante di rete (Figura 3.4).
Bisogna selezionare l'opzione centrale, *Connetti alla stampante* (Figura 3.5).

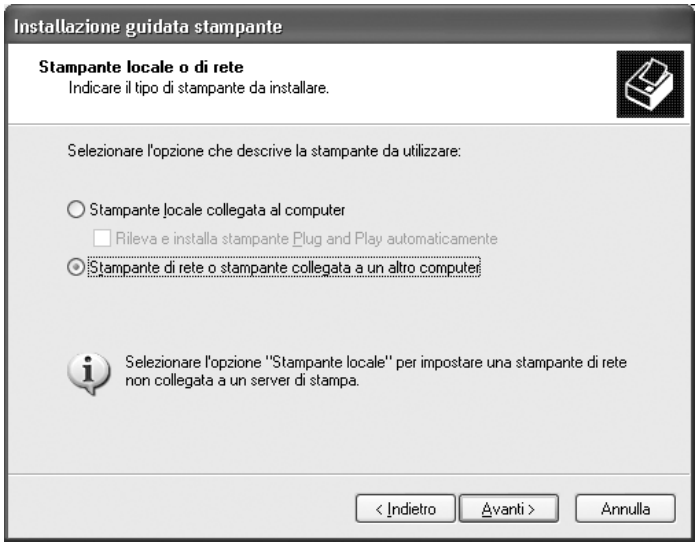

Figura 3.4
Prima di tutto, bisogna indicare se la stampante è locale o di rete.

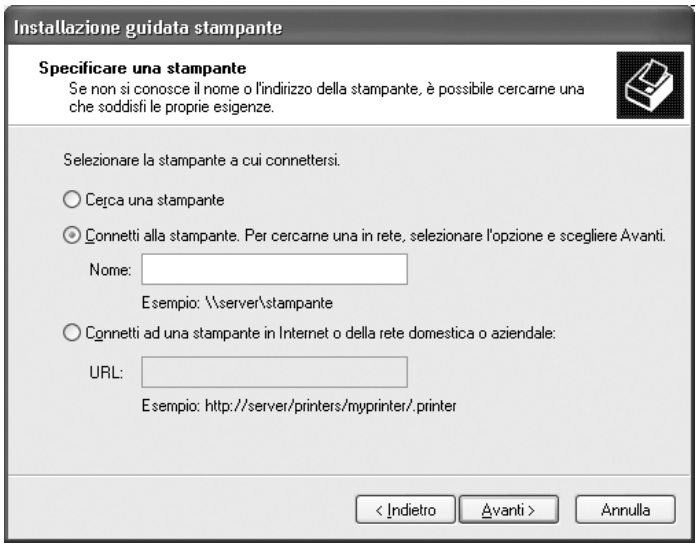

Figura 3.5
Successivamente bisogna indicare il percorso della stampante di rete.

Facendo clic su *Avanti* comparirà l'elenco dei sistemi presenti in rete. Si deve scegliere il print server, in questo caso `server1`, selezionare la stampante condivisa in precedenza e poi fare clic su *Avanti* (Figura 3.6).

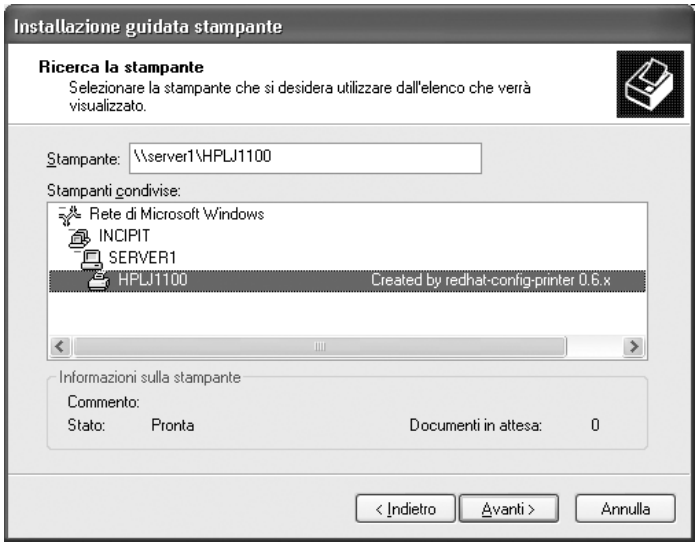

Figura 3.6
Selezione del print server e della stampante condivisa.

Il sistema indicherà che non è presente il driver per la stampante. Bisogna quindi fornire il percorso del driver ufficiale per Windows, scaricato dal sito del produttore (Figura 3.7).

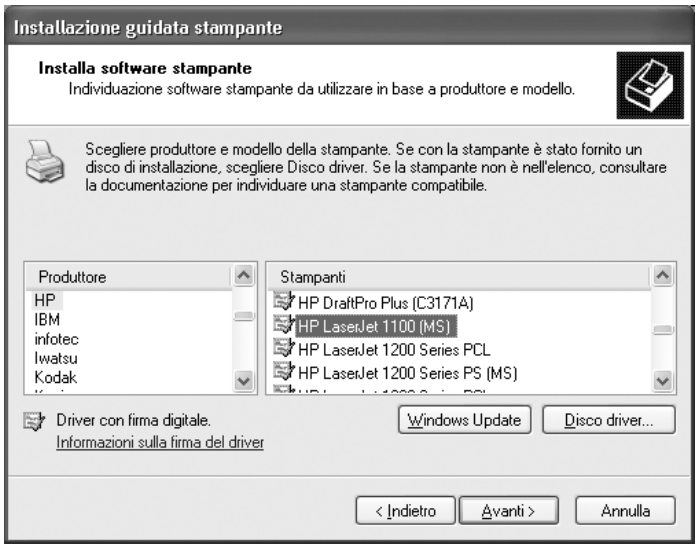

Figura 3.7
Scelta del produttore e del modello della stampante.

Per concludere la fase di installazione del driver, bisogna specificare se la stampante è predefinita o meno: eseguita la scelta si fa clic su *Avanti*.
Ora si può cominciare a stampare sul server di stampa. Da un punto di vista pratico non sussiste alcuna differenza tra una stampante locale e la stampante di rete appena realizzata. Non bisogna quindi cambiare alcun aspetto del proprio modo di lavorare. Il listato che segue mostra la configurazione di stampa del dominio.

```
#Configurazione della sezione di stampa per un dominio

[global]
workgroup = INCIPIT
netbios name = SERVER1
server string = PDC Linux
security = USER
smb passwd file = /etc/samba/smbpasswd
encrypt passwords = YES
log file = /var/log/samba/%m.log
max log size = 100
log level = 1
# impostazione del server come domain master browser
os level = 255
preferred master = YES
local master = YES
domain master = YES
# abilitazione dei logon W95/W98
domain logons = YES
# attivazione supporto WINS
wins support = YES
# impostazione homedir
logon home = \\server1\homedir
logon drive = U:
# impostazione profili mobili
logon path = \\server1\profili\%u
# impostazone script di logon
logon script = logon.bat
#logon script = %u.bat
# impostazione stampa
printing = CUPS
printcap = CUPS

[netlogon]
path = /usr/local/samba/netlogon
read only = YES
write list = root

[profili]
path = /usr/local/samba/profili
read only = NO
writable = YES
browsable = NO
create mask = 0600
directory mask = 0700
```

```
[homedir]
path = /home/%u
read only = NO
writable = YES
browsable = NO
create mask = 0600
directory mask = 0700
hide dot files = YES

[comune]
comment = cartella comune
path =/home/comune
public = YES
writable = YES

[gestionale]
comment = area supporto software gestionale
path =/home/gestionale
public = YES
writable = YES

[software]
comment = file di utilità
path =/home/software
public = YES
writable = NO

[printers]
comment = stampanti sul server
path = /var/spool/samba
printable = YES
use client driver = YES
```

Checklist

1. Utilizzare la configurazione precedentemente creata su Samba, per operare su un workgroup oppure su un dominio.
2. Verificare che il pacchetto CUPS sia installato sul sistema.
3. Collegarsi al pannello web di CUPS, specificando nell'indirizzo locale la porta 631.
4. Configurare su CUPS una nuova stampante facendo attenzione ad impostare la modalità di elaborazione RAW.
5. Aprire il file `/etc/cups/mime.types` ed eliminare il simbolo di commento dalla riga che inizia con `#application/octet` in fondo al file.
6. Aprire il file `/etc/cups/mime.convs` ed eliminare il simbolo di commento dalla riga che inizia con `#application/octet`.
7. Aprire il file di configurazione di Samba e, nella sezione `global`, inserire le direttive `printing = CUPS` e `printcap = CUPS`.

8. Creare una sezione `printers` nel file di configurazione di Samba.
9. Indicare la direttiva `comment`.
10. Indicare nella direttiva `path` il percorso della directory di spooling, che deve ovviamente essere una directory esistente e accessibile.
11. Inserire la direttiva `printable = YES`, per poter effettuare le operazioni di spooling.
12. Inserire la direttiva `use client driver = YES` per poter stampare in maniera corretta sui sistemi Windows NT/2000/XP.
13. Verificare la correttezza della configurazione utilizzando il comando `testparm`.
14. Riavviare i demoni di Samba.
15. Configurare i client.

Capitolo 4

Realizzazione di un DNS

All'inizio di Internet vi era solamente il numero: per accedere a qualunque sistema remoto era necessario conoscere e ricordarsi un indirizzo numerico che identificava quel particolare server. Questo poteva andare bene nella fase iniziale di Internet, quando c'erano pochi nodi connessi e l'utenza era prevalentemente d'indirizzo tecnico e scientifico. Con l'aumentare della complessità della Rete e del numero di utenti, cominciarono a essere chiari i limiti di un sistema che imponeva di ricordare valori composti anche da 12 cifre. Sarebbe come pretendere di conoscere a memoria i numeri telefonici di tutti gli amici, i colleghi, i negozi, le aziende e gli enti pubblici che vengono utilizzati quotidianamente.
Serviva una soluzione e una proposta si rivelò efficace: bastava fornire a ogni computer connesso un nome descrittivo e facile da ricordare, per esempio `apogeonline.com` e creare un elenco che associasse a ogni nome esteso l'indirizzo numerico corrispondente. Per accedere a un sistema bastava quindi digitare il nome mnemonico.
Questa soluzione non imponeva un cambio radicale della struttura tecnica di Internet: gli indirizzi numerici rimanevano alla base della Rete, ma un server eseguiva la conversione dal nome esteso al valore corrispondente, permettendo il collegamento. Questo sistema è valido ancora oggi, se pur con alcune modifiche e gli utenti di Internet ne fanno uso costantemente. Il nome di questo meccanismo è DNS (*Domain Name Server*).

Generalità sul DNS

Il DNS è un servizio di rete che viene installato su un server liberamente raggiungibile. Il suo scopo è ricevere interrogazioni dai client (per esempio `www.apogeonline.com`) e fornire in risposta gli indirizzi relativi (nel caso precedente, 212.239.21.50).

Ogni volta che si crea una rete di computer o di server che devono essere accessibili tramite nomi estesi è necessario installare e configurare un sistema DNS. Al suo interno saranno presenti i nomi descrittivi e gli indirizzi dei computer presenti nella propria rete, una sorta di elenco telefonico.
Per comprendere meglio il tutto si consideri il caso di un'azienda che intende attivare un sistema web per la vendita di prodotti sportivi. Il primo passo consiste nel prendere in affitto una certa quantità di banda passante. Qualcuno deve cioè fornire un cavo dati ad alta velocità connesso alla rete Internet mondiale. Generalmente questa fornitura viene compiuta da qualche grande carrier, per esempio Telecom Italia, che ha creato una rete dati ramificata nel territorio, connessa a tutti i punti di scambio nazionali e internazionali.
Ottenuta la connessione full time bisogna farsi assegnare un indirizzo IP fisso: si deve avere un valore univoco che identifichi il proprio server; chiunque potrà così accedervi semplicemente digitando quel numero.
Gli indirizzi IP sono assegnati da un organismo internazionale preposto alla distribuzione secondo regole ben precise. Purtroppo non si tratta di una risorsa illimitata ma bisogna piuttosto razionalizzare la distribuzione di questo bene. Secondo gli standard attuali si hanno circa due miliardi di indirizzi unici, buona parte dei quali già impegnati.
Gli indirizzi non sono quasi mai rilasciati all'utente finale bensì ad aziende che operano nel settore delle telecomunicazioni o di Internet e dotate di comprovati requisiti tecnici. L'utente finale riceve uno o più indirizzi statici solo a seguito della stipula di un contratto di connettività con uno di questi operatori.
Le connessioni professionali, anche quelle più economiche basate per esempio su ADSL, sono generalmente corredate di pacchetti di indirizzi statici proprio per andare incontro a coloro che intendono pubblicare servizi online.
A questo punto si ha tutto quello che serve. Il servizio di vendita di articoli sportivi risulta connesso a Internet a tempo pieno e accessibile via browser digitando l'indirizzo numerico nella forma `http://111.111.111.1`.
Manca però ancora il nome: qualunque utente, dovunque si trovi, deve poter digitare il nome `www.supersportshop.com` ed essere ridiretto al server che si trova all'IP 111.111.111.1. Bisogna, quindi, procedere con l'acquisto del nome di dominio attraverso un'apposita struttura di rilascio, per esempio Network Solutions (`www.netsol.com`). L'operazione è facile e permette di ottenere il dominio entro pochi minuti (Figura 4.1).
A questo punto si ha un indirizzo IP statico e un nome di dominio. Ora entra in gioco il DNS per l'associazione tra il nome simbolico e l'indirizzo numerico.
Bisogna installare il servizio sulla stessa macchina che fa da server web (111.111.111.1) oppure su un server a parte, per esempio 111.111.111.2, purché raggiungibile dall'esterno. Si sceglie per questo esempio la seconda soluzione.

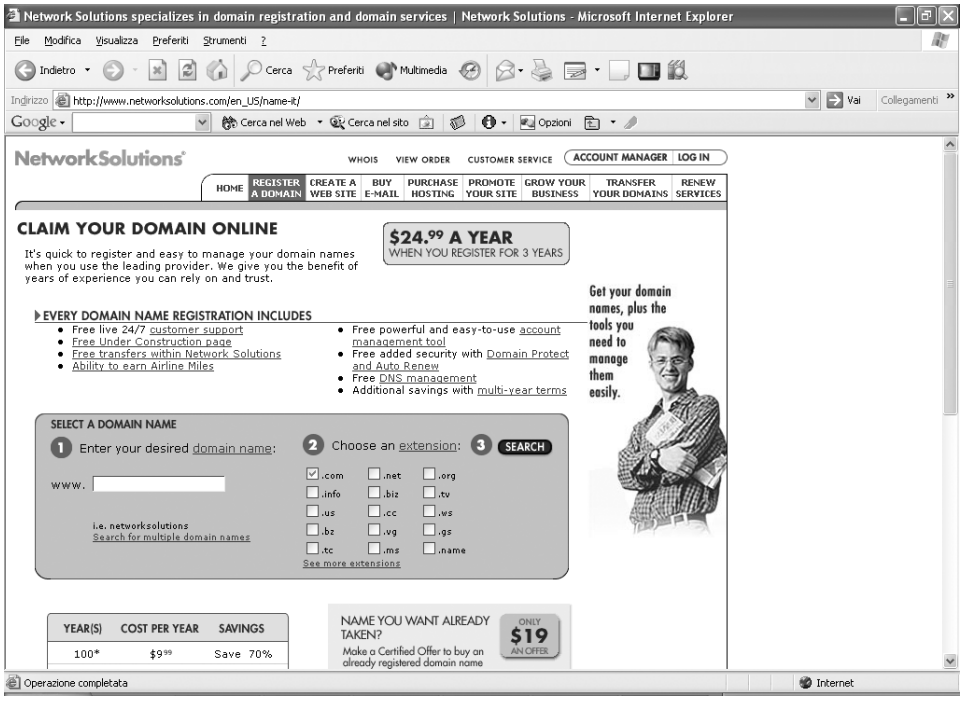

Figura 4.1
NetSol, il più famoso servizio di registrazione di domini.

L'elenco delle associazioni deve essere compilata manualmente dall'amministratore di sistema. Nel caso esaminato bisogna inserire una riga per il servizio `www.supersportshop.com` e abbinare l'indirizzo IP relativo. Con lo stesso principio si dovranno indicare eventuali altri servizi interni che si vogliono in qualche modo rendere pubblici.
Il DNS dovrà infine essere collegato agli altri sistemi DNS per fare in modo che l'elenco dei nomi delle proprie macchine sia consultabile da tutto il mondo; in caso contrario sarà visibile solo localmente. Per farlo si deve ritornare nuovamente su Network Solutions, entrare nel pannello di gestione dell'account e indicare che il DNS che gestisce il dominio `supersportshop.com` non è più il server di Network Solutions ma piuttosto il computer all'indirizzo 111.111.111.2. Fino a quel momento era il server di Network Solutions che risolveva l'indirizzo esteso puntando a una generica pagina.
Confermando la nuova configurazione si attiva una procedura di aggiornamento a livello internazionale e nel giro di un paio di giorni le nuove impostazioni saranno diffuse in tutti gli angoli del pianeta.
Quando un utente digiterà sul proprio browser l'indirizzo `www.supersportshop.com`, il sistema sarà istruito sul fatto che gli indirizzi di `supersportshop.com` sono gestiti presso 111.111.111.2. Il client accederà a questo DNS in maniera diretta, segnalerà la necessità di conoscere l'indirizzo del server

web e otterrà il valore 111.111.111.1. A questo punto il browser eseguirà un accesso diretto a questo indirizzo in maniera trasparente, recuperando le pagine.

Utilità del DNS in una rete Windows

L'esempio precedente dimostra l'importanza di un sistema DNS quando esiste la necessità di mettere online un servizio pubblico su Internet. Non tutte le realtà hanno però le risorse o la necessità di configurare e gestire internamente i servizi Internet. In questi casi viene completamente tralasciata la gestione locale del DNS e tutti i problemi relativi sono demandati a qualche azienda esterna, per esempio il fornitore di connettività o un'azienda di servizi informatici.
Purtroppo però il sistema DNS non è necessario solo nel caso in cui si vogliano pubblicare risorse Internet. Lo strumento risulta infatti vitale nel caso in cui si abbia una normale rete Windows con programmi 2000 o XP in versione Professional al suo interno.
Anche Windows ha infatti bisogno di un meccanismo di risoluzione dei nomi. Le reti Microsoft, come quelle Internet, sono composte da computer che hanno nomi simbolici descrittivi, per esempio *Amministrazione*, *UffTec1*, *ServerCentrale* ecc. I computer funzionano però in rete locale per via di indirizzi IP numerici e non di un nome; nasce, quindi, nuovamente l'esigenza di un sistema in grado di convertire nomi estesi in valori numerici.
In passato, nelle reti dotate di programmi 95, 98, ME con server Windows NT, sussisteva un meccanismo di conversione molto semplice. Se *UffTec1* aveva bisogno di accedere al pc *Amministrazione* per leggere un file di Excel, veniva prima di tutto cercato un server WINS. Si tratta di un componente di sistema su Windows NT 4 Server che è in grado di memorizzare in maniera automatica i nomi dei computer e i relativi indirizzi numerici. È qualcosa di molto simile al DNS, ma limitato all'ambito della rete locale e aggiornato automaticamente dal server stesso.
UffTec1 eseguiva allora un accesso a ServerCentrale dove era installato il WINS e interrogava il sistema per sapere l'indirizzo di *Amministrazione*. Il sistema WINS verificava il proprio database e forniva la risposta, utilizzata poi da *UffTec1*.
Come faceva però *UffTec1* a sapere dove si trovava il server WINS? Ogni computer ha una configurazione nel pannello del TCP/IP con l'indirizzo di questo servizio. *UffTec1* avrebbe quindi creato un accesso al WINS attraverso l'IP indicato dall'amministratore in fase di configurazione.
Non è però obbligatorio disporre di un server NT 4 o di un sistema WINS attivato per ottenere la risoluzione dei nomi. Le reti Windows basate su Windows 9x/ME e sistemi NT4 avevano anche altri meccanismi. Se non c'era il sistema WINS veniva fatta la ricerca all'interno del file hosts, localizzato den-

tro la directory di sistema. Questo non è altro che un file di testo con un elenco di associazioni tra nomi simbolici e indirizzi numerici.
Se non era presente neppure il file hosts veniva impiegata una politica drastica: si effettuava il broadcast. *UffTec1* mandava una comunicazione a tutti i computer presenti in rete, richiedendo ad *Amministrazione* di rispondere fornendo il proprio indirizzo. Si tratta di una sorta di appello del primo giorno di scuola. Il professore si trova davanti a 20 allievi che non conosce e rivolgendosi a tutti, chiede all'alunno Paolo Rossi di alzare la mano.
È un metodo funzionale che nel mondo informatico crea però traffico inutile sulla rete e inefficienza. Infatti, non è molto sensato interpellare tutti i computer se si vuole in realtà "parlare" con uno solo di questi. Si tratta comunque del metodo che è stato a lungo impiegato dalle aziende per scelta, necessità o poca conoscenza tecnica.

Cambiamenti in Windows 2000

La situazione è però cambiata in maniera significativa a partire da Windows 2000. Microsoft ora richiede la presenza di un server DNS per l'implementazione di una rete Windows moderna. La risoluzione dei nomi avviene primariamente con questo strumento e solo in seguito con i vecchi meccanismi WINS, Hosts e broadcast (nel caso il nome non sia presente nel DNS o non vi sia addirittura il DNS stesso).
In pratica la risoluzione avviene con un meccanismo a quattro fasi in cui il sistema operativo cerca di ottenere l'indirizzo usando strategie differenti. La procedura si ferma appena il nome viene trovato o con un errore, nel caso in cui il broadcast non dia risultati.
Molti tecnici di rete hanno però frainteso questo meccanismo. L'idea è che se il DNS non è presente, verrà comunque utilizzato uno dei meccanismi convenzionali in maniera automatica. Perché allora darsi la pena di implementare un DNS interno?
Si dimentica che tutti i sistemi di una rete moderna dispongono di un indirizzo DNS per gli accessi a Internet. Se manca un DNS interno sarà utilizzato questo sistema esterno ogni volta che si deve accedere a un computer della rete locale. Naturalmente sui DNS mondiali non ci saranno indicazioni sulla macchina interna denominata *Amministrazione*. L'interrogazione restituirà un messaggio di host non trovato. Questa procedura non è istantanea e richiede un tempo relativamente alto per essere portata a termine.
Solo dopo questo lasso di tempo verranno eseguiti tentativi di risoluzione interni. Gli utenti misureranno in tal caso una notevole lentezza nell'uso dei servizi della lan e si creerà un disservizio non indifferente.
In alcuni casi si possono avere ripercussioni anche economiche. In molte città non sono disponibili connessioni a Internet full time. In queste situazioni si ha, per esempio, un router ISDN che si collega ogni volta che viene

richiesto l'accesso a Internet. La connessione rimarrà attiva per un lasso di tempo e poi sarà interrotta automaticamente.
In un simile contesto ogni accesso a un computer interno comporterà l'accesso a Internet su ISDN per l'interrogazione di un DNS esterno. I tempi di accesso alle risorse saranno elevati e i costi in bolletta cresceranno inutilmente. Per risolvere tutti questi problemi è necessario installare un server DNS. Se si hanno però pochi client Windows 2000 o Windows XP non risulta conveniente acquistare una licenza di Windows 2000 Server. Bisogna infatti prevedere un server adeguato, comprare le licenze software e valutare i tempi necessari alla configurazione e alla manutenzione nel tempo di un prodotto complesso. Il problema si può risolvere con una soluzione Linux. Su Linux la gestione del DNS avviene tramite il pacchetto Bind, disponibile liberamente in tutte le distribuzioni, con l'indirizzo `http://www.isc.org/products/BIND/` (Figura 4.2).

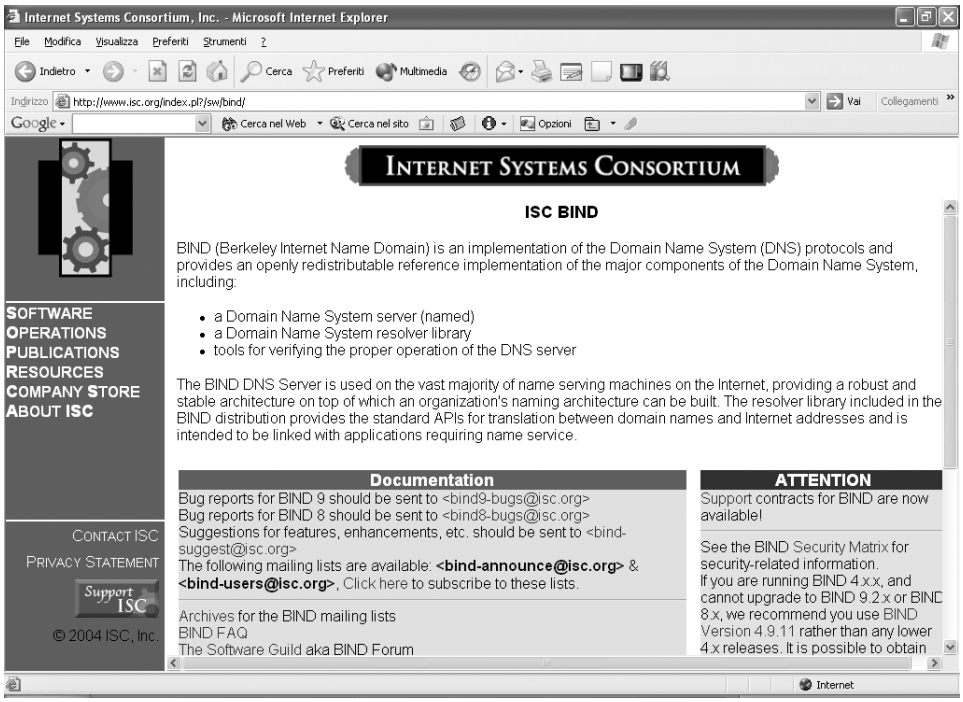

Figura 4.2
Sito ufficiale di Bind.

Bind permette di gestire un sistema DNS in maniera completamente standard e può essere di aiuto sia nel caso in cui si voglia mettere online un servizio pubblico su Internet sia quando necessita un'infrastruttura di risoluzione dei nomi per una rete Windows.

Gestione di una rete Windows

Si consideri il caso di voler gestire con Bind la risoluzione per una rete locale composta da tre client Windows XP Professional, un print server e una connessione a Internet tramite ADSL.
Le macchine si chiamano PC1, PC2 e PC3 e hanno indirizzi IP interni, rispettivamente 192.168.100.10, 192.168.100.11 e 192.168.100.12. Il print server è invece configurato per rispondere all'indirizzo IP 192.168.100.50.
Gli indirizzi devono essere fissati manualmente e non rilasciati con meccanismi di assegnazione automatica come DHCP. Non bisogna infatti mai dimenticare che il servizio DNS è di tipo statico e che tutte le macchine gestite dal sistema devono essere specificate manualmente. Se una macchina cambia indirizzo, bisogna applicare immediatamente una modifica anche nel DNS. In caso contrario, Bind fornirebbe il vecchio indirizzo a qualunque sistema che ne facesse richiesta.
Il sistema Linux può essere installato su una macchina di recupero dotata di un processore Pentium II a 350 MHz, di 256 MB di ram e con un HD da 20 GB. Una configurazione estremamente modesta per gli standard informatici odierni. Si tratta, comunque, di una macchina più che sufficiente per il funzionamento di un DNS interno per una piccola o media impresa; anche questo server avrà un indirizzo IP statico, 192.168.100.4 (Figura 4.3).

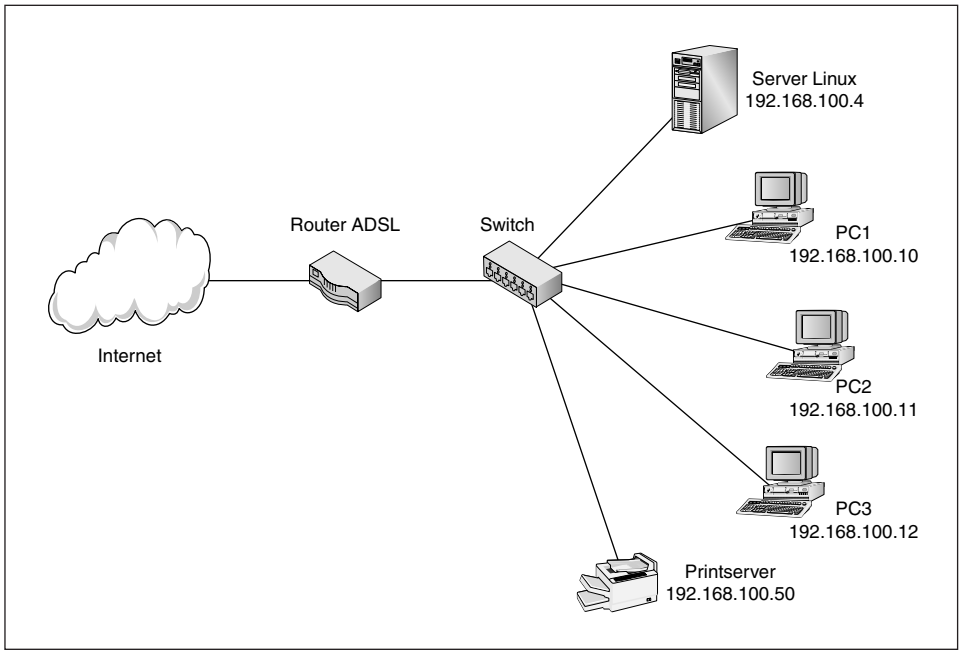

Figura 4.3
Schema della rete locale in oggetto.

Configurazione del DNS

Per personalizzare Bind bisogna innanzitutto cercare il file di configurazione principale, che a seconda delle distribuzioni può trovarsi dentro la directory `/etc` oppure in `/etc/bind` con il nome `named.conf`.
Il file contiene una configurazione generica di default applicata in fase di installazione della distribuzione. Si avrà probabilmente una situazione molto simile alla seguente:

```
## named.conf - configuration for bind
#
# Generated automatically by redhat-config-bind, alchemist et al.
# Any changes not supported by redhat-config-bind should be put
# in /etc/named.custom
#
controls {
        inet 127.0.0.1 allow { localhost; } keys { rndckey; };
};

include "/etc/named.custom";
include "/etc/rndc.key";


zone  "0.0.127.in-addr.arpa" {
  type master;
  file  "0.0.127.in-addr.arpa.zone";
};


zone  "localhost" {
  type master;
  file  "localhost.zone";
};
```

Il file `named.conf` può essere idealmente suddiviso in due parti: una sezione di specifiche di funzionamento e una serie di specifiche "zone". Le specifiche di funzionamento possono essere ignorate ai fini pratici in quanto servono per il funzionamento base del server DNS e sono già impostate di default. Quello che interessa, invece, per questa trattazione è il paragrafo sulle zone. Una zona è un insieme di computer raggruppati secondo un qualche criterio scelto dall'amministratore. Nelle piccole realtà si sceglie generalmente di identificare una zona con il dominio stesso. In pratica, se si ha un dominio per la propria azienda, si può creare una zona DNS con tutti i computer aziendali al suo interno.
Di default si hanno due zone: `localhost` e `0.0.127.in-addr.arpa`. Sono nomi standard che si riferiscono al sistema locale, il primo per le ricerche dirette e il secondo per le ricerche inverse. L'etichetta `in-addr.arpa` è una formula storica che rappresenta sempre un dominio di ricerca inversa. Una ricerca diretta è una interrogazione che punta a determinare l'indirizzo IP a

partire dal nome esteso, per esempio fornendo il nome `www.apogeonline.com`. Ogni volta che si scrive un indirizzo nel browser e si preme il tasto Invio della tastiera si genera un'interrogazione diretta al DNS per ottenere l'indirizzo IP del sistema che si vuole raggiungere.
La ricerca inversa si ha invece quando si vuole sapere quale nome è associato a un determinato IP.
Per convenzione il sistema locale è sempre associato all'IP 127.0.0.1. Questo valore, detto *indirizzo di loopback*, è stato introdotto per fare in modo che ogni macchina possieda sempre un indirizzo valido. L'utilizzo è rigorosamente interno. Non si troverà cioè mai un computer esterno con questo IP.
Un ping verso 127.0.0.1 genererà un "annello" locale. La richiesta sarà gestita dal sistema TCP/IP del proprio computer senza uscire dalla scheda di rete. Questa operazione è molto utile per verificare che lo stack TCP/IP stia funzionando correttamente o semplicemente per fare riferimento in maniera generica a se stessi. Molte applicazioni e servizi di rete usano questo indirizzo per funzionare. In questo modo non hanno bisogno di sapere l'indirizzo IP che è stato assegnato dall'amministratore di sistema alla macchina locale. Molte configurazioni di default risultano così funzionanti appena installate nel sistema.
La specifica di zona *localhost* del file di configurazione contiene un riferimento al file `localhost.zone` mentre la specifica di zona `0.0.127.in-addr.arpa` contiene un riferimento al file `0.0.127.in-addr.arpa.zone`. Questi file si trovano dentro `/etc/bind` e sono la parte centrale del sistema DNS.
Il file `localhost.zone` contiene la fisionomia seguente:

```
$TTL 86400
@  IN  SOA  @  root.localhost (
      1 ; serial
      28800 ; refresh
      7200 ; retry
      604800 ; expire
      86400 ; ttl
      )


   IN  NS  localhost.


@  IN  A  127.0.0.1
```

Si inizia con una chiave `$TTL 86400`, che non deve essere cambiata, e un preambolo contraddistinto da `SOA` *(Start Of Authority)*. Questo inizia con il simbolo `@`, finisce alla chiusura della prima parentesi graffa e contiene alcuni dettagli necessari al funzionamento del sistema DNS.
Il simbolo `@` è una abbreviazione per quella che si definisce origine. Il file `localhost.zone` che si sta esaminando è stato specificato in `named.conf`

nella zona localhost. L'origine è perciò `localhost`, il nome di zona specificato in `named.conf`.
La prima riga del preambolo viene quindi tradotta in questo modo:

```
localhost.   IN SOA localhost.  root.localhost (
```

Si può notare un punto alla fine di `localhost`. Il punto indica che l'indirizzo è assoluto e letterale. Se non c'è il punto, si ha invece un indirizzo relativo. Questo non è più letterale ma composto automaticamente aggiungendo in fondo il nome di zona; quindi, `localhost.` significa semplicemente `localhost,` mentre `localhost` (senza punto) viene automaticamente espanso in `localhost.localhost` ogni volta che viene usato.
Questa scelta sintattica permette alcuni automatismi in Bind e serve per rendere più corti i file di configurazione, evitando di dover sempre scrivere la zona per esteso nel caso in cui i nomi siano molto lunghi. Nessuno infatti vieta di avere una zona chiamata `ufftec.bologna.italia.incipit.biz`. Sarebbe però scomodo riscrivere questa stringa di testo ogni volta che si volesse inserire un indirizzo nel DNS. Invece di scrivere per esempio `pc1.ufftec.bologna.italia.incipit.biz` è sufficiente scrivere `pc1` (senza il punto). Il sistema espanderà automaticamente il nome.
Le righe seguenti contengono indicazioni utili ai fini della sincronizzazione del server DNS locale con server DNS esterni. Questo è utile quando sono definite zone che sono in realtà specificate in altri server DNS esterni. In questo caso serve un numero di serie che venga incrementato a ogni modifica dei record DNS, un tempo di refresh che indichi ogni quanto tempo i DNS devono aggiornarsi, un tempo di retry per i tentativi seguenti in caso di mancata sincronizzazione e un tempo di expire al quale il DNS perde validità se non riesce più a connettersi con il DNS esterno. Subito prima dell'apertura della graffa compare `root.localhost`. Qui dovrebbe apparire l'indirizzo e-mail del gestore del DNS in una notazione priva di `@` (che sarebbe interpretata come nome di origine). Al suo posto compare un punto. Questo indirizzo rappresenta in realtà `root@localhost`.
A titolo di esempio l'indirizzo `admin@supersportshop.com` dovrebbe qui essere scritto come `admin.supersportshop.com`.
Se il DNS contiene solo indicazioni locali autonome, senza riferimenti ad altri DNS, si consiglia di lasciare questi campi del preambolo inalterati.
La specifica seguente è NS che sta per *nameserver*. Qui è indicato che il DNS si trova sul sistema locale localhost. Di seguito si ha la specifica A che sta per *Address*. Queste sono le associazioni nomi-indirizzo. Si associa qui che localhost (abbreviato come `@`) è presente all'indirizzo 127.0.0.1.
In ogni riga di specifica compare IN, una convenzione sintattica che deve essere rispettata.

Esempio pratico

Le configurazioni esaminate fino a qui sono estremamente generiche e regolano semplicemente il funzionamento del proprio sistema Linux dopo un'installazione del tutto standard. Per servire la rete locale è necessario specificare i nomi dei computer, i relativi indirizzi e creare una zona di pertinenza. Per fare questo si deve andare nel file di configurazione generale `named.conf` e inserire una nuova zona:

```
zone  "incipit.biz" {
  type master;
  file  "incipit.biz.zone";
};
```

In questo caso si sta specificando che esiste una nuova zona denominata `incipit.biz` dentro il file `incipit.biz.zone`; `incipit` è in questo esempio il nome della propria organizzazione.
Il tipo è definito master perché tutte le informazioni su quella zona sono presenti in maniera esaustiva dentro il file `incipit.biz.zone`. Si sta in pratica affermando che questo DNS ha la piena autorità su quella zona.
In alternativa esiste anche il tipo slave. In tal caso si afferma che la zona è specificata in un altro DNS e che il sistema locale deve trarre tutte le informazioni su quella zona da quel preciso DNS esterno. Viene in tal caso realizzato un file locale con le informazioni tratte in remoto. Questo file ha una scadenza dopo la quale sarà nuovamente aggiornato con un nuovo accesso al DNS di riferimento. Questo permette ai due sistemi DNS di rimanere sempre sincronizzati.
Il file `incipit.biz.zone` dovrà contenere riferimenti ai sistemi presenti nel proprio ufficio. Sono PC1, PC2, PC3 e il print server:

```
$TTL 86400
@  IN  SOA  @  info.incipit.biz (
      4 ; serial
      28800 ; refresh
      7200 ; retry
      604800 ; expire
      86400 ; ttl
      )

  IN  NS  127.0.0.1.

pc1    IN  A  192.168.100.10
pc2    IN  A  192.168.100.11
pc3    IN  A  192.168.100.12
printserver  IN  A  192.168.100.50
```

Si può notare che il preambolo contiene la forma standard vista in precedenza ma con l'indirizzo dell'amministratore locale `info.incipit.biz` (notazione per `info@incipit.biz`).
La specifica `NS` punta al sistema locale, come nel caso precedente mentre si hanno invece una serie di specifiche A (*address*), uno per ogni sistema. Qui diventa finalmente chiara l'associazione tra i nomi estesi e gli indirizzi IP numerici.
Quando un client scriverà per esempio `ping pc1`, il computer andrà in questo archivio DNS, scandirà la lista delle specifiche A, troverà la riga `pc1`, otterrà l'indirizzo IP numerico e potrà così eseguire effettivamente la richiesta con un ping a 192.168.100.10.
Lo stesso accade quanto si visita un qualunque sito. Quando si digita per esempio `www.linux.org` il proprio computer esamina un archivio DNS che fa riferimento a `linux.org` e scorre i record A per trovare la voce `www`. Una volta individuata, legge l'IP corrispondente e lo fornisce al browser per la connessione.
La configurazione precedente può essere arricchita con il riferimento per il server Linux e il gateway Internet:

```
server    IN  A  192.168.100.1
gateway   IN  A  192.168.100.5
```

A questo punto i file di configurazione risultano creati. Per renderli operativi bisogna attivare il demone DNS o procedere al riavvio nel caso il servizio sia già in funzione. Per attivare il demone bisogna digitare il seguente comando:

```
/etc/init.d/named start
```

Se il demone è stato lanciato in fase di boot bisogna semplicemente riavviare il servizio per attivare le modifiche appena inserite:

```
/etc/init.d/named restart
```

Non bisogna a questo punto dimenticare di fare in modo che il servizio DNS sia attivato automaticamente all'avvio del sistema. Generalmente viene inserito automaticamente in fase di installazione. Nel caso non lo fosse si deve procedere manualmente oppure si può usare qualche strumento più veloce, come per esempio il comando *ntsysv*, sui sistemi RedHat.
Il DNS è ora configurato e attivo. Bisogna a questo punto istruire il sistema a utilizzarlo. Prima di tutto bisogna ritornare nella directory `/etc` e aprire il file `host.conf`.
Questo file specifica l'ordine con cui viene eseguita la risoluzione dei nomi. Di default si ha la configurazione seguente:

```
order hosts, bind
```

Questo significa che viene prima esaminato il file `hosts`, presente anch'esso dentro `/etc` e poi il sistema DNS.
Il file `hosts` è un elenco che associa i nomi estesi agli indirizzi IP ed è molto simile al DNS ma estremamente più semplice e privo della strutturazione gerarchica del DNS. Di default contiene la riga seguente:

```
127.0.0.1 localhost.localdomain localhost
```

Si tratta dell'associazione dell'indirizzo di loopback locale al nome localhost, un alias comunemente usato dalle applicazioni Linux per fare riferimento al sistema locale. Un ping a localhost equivale a un ping a 127.0.0.1.
La configurazione di default di `host.conf` e di `hosts` può essere lasciata inalterata anche se alcuni utenti preferiscono invertire l'ordine di ricerca in `host.conf` per avere prima l'interrogazione sul DNS e poi quella nel file `hosts`:

```
order bind, hosts
```

Per concludere bisogna controllare un ulteriore file di sistema: `/etc/resolv.conf`. Questo file contiene gli indirizzi dei server DNS che il computer locale deve interrogare per risolvere i nomi. Bisogna semplicemente specificare che si vuole usare il DNS appena configurato:

```
nameserver 127.0.0.1
```

Questa operazione attiva il DNS per le operazioni locali. Un ping dalla shell di Linux a `pc2.incipit.biz` sarebbe correttamente risolto nell'indirizzo 192.168.100.11.
Sussiste però un problema. Sono stati specificati unicamente i sistemi locali. Se si tentasse di accedere con il browser a qualunque sito esterno o a qualunque altro tipo di servizio si verificherebbe un errore. Il DNS non contiene infatti riferimenti al mondo esterno. Il problema può essere risolto in maniera molto rapida aggiungendo nuove righe in `resolv.conf` con gli indirizzi di DNS del proprio provider:

```
nameserver 127.0.0.1
nameserver 151.99.125.2
nameserver 151.99.250.2
```

Gabbie chroot

Potrebbe succedere che le impostazioni sul DNS non abbiano effetto e che qualunque operazione porti a errori o a comportamenti del tutto inaspettati. Questo può succedere se il DNS è implementato sulla propria distribuzione all'interno di una gabbia chroot. Si tratta in sostanza di un filesystem in miniatura isolato dal resto del sistema per motivi di sicurezza. Se un utente malintenzionato riuscisse a violare il DNS non potrebbe in tal caso fare danni al

sistema in quanto si troverebbe all'interno della gabbia chroot, un'area con un numero estremamente limitato di comandi critici.
Se il DNS funziona all'interno di una gabbia chroot significa che il file di configurazione principale non si trova dentro `/etc` e che le specifiche di zona non si trovano in `/var/named`. La struttura operativa si trova in realtà altrove. Fedora Core, per esempio, crea una gabbia chroot per Bind e qualunque configurazione sul file in `/etc` non porterebbe ad alcun risultato. Bisogna piuttosto andare in `/var/named/chroot` (la gabbia chroot) e usare il file `named.conf` presente dentro `/etc` e configurare le zone dentro `/var`. È importante verificare questo aspetto per non perdere ore cercando di configurare i file sbagliati.

Configurazione dei client Windows

Per configurare i client si deve andare in *Pannello di Controllo/Rete* e aprire la configurazione del TCP/IP. In basso nella configurazione DNS bisogna specificare l'indirizzo del sistema Linux, 192.168.100.4. Come indirizzi secondari bisogna invece mettere i DNS del proprio provider. In questo modo si ha il proprio DNS per gli indirizzi interni e i DNS pubblici del provider per tutti gli altri indirizzi Internet (Figura 4.4).

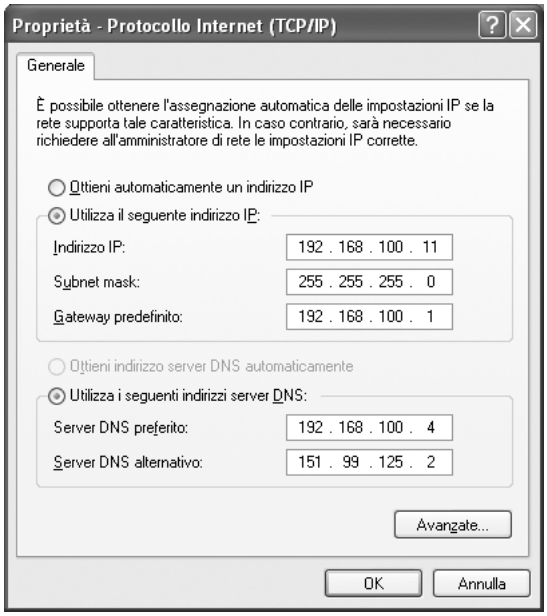

Figura 4.4
Inserimento dei parametri di rete e dei valori di DNS.

Per verificare che stia tutto funzionando a dovere si può aprire una shell dei comandi andando in *Start/Esegui*, digitare `cmd` e premere Invio.
Si deve inserire il comando *nslookup* e premere nuovamente Invio. Il programma dovrebbe partire indicando il server DNS di riferimento interno 192.168.100.4 (Figura 4.5).

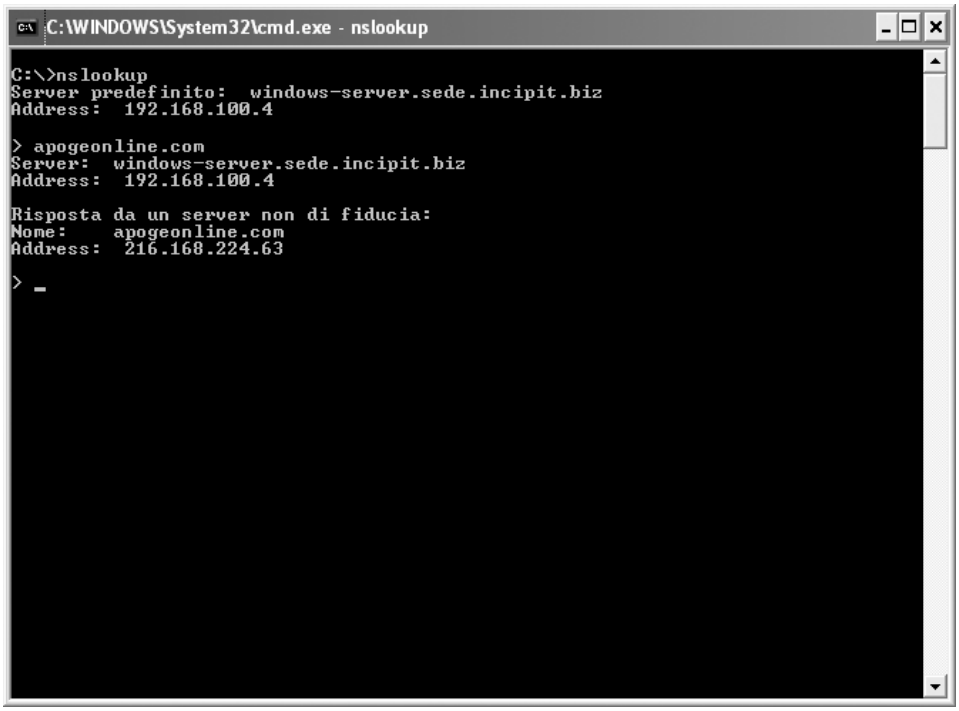

Figura 4.5
Esecuzione del comando nslookup.

Digitando i nomi estesi delle macchine interne si dovrebbe ottenere automaticamente l'indirizzo. Scrivendo per esempio PC2 si dovrebbe ottenere come risposta 192.168.100.11. Inserendo invece un indirizzo esterno come `www.sun.com` si dovrebbe ottenere il corretto indirizzo IP prelevato dal sistema DNS che gestisce il nome.
Un punto chiave da tenere in considerazione è il suffisso che viene aggiunto ai nomi. I nomi Internet sono infatti composti da un host e da un dominio. `PC2` è, per esempio, il nome dell'host mentre `incipit.biz` è quello del dominio. Windows 2000 e Windows XP aggiungono automaticamente il suffisso ai nomi digitati. Digitando PC2 su nslookup dovrebbe venire automaticamente aggiunto il dominio per completare in maniera corretta l'indirizzo.
Per verificare il suffisso di default bisogna trascinare il mouse sopra *Risorse del computer* di Windows XP, fare clic sul tasto destro del mouse, scegliere

dal menu la voce *Proprietà* e selezionare la scheda *Nome computer*. A circa metà della finestra si trova la voce *Dominio*, in cui è indicato il suffisso di default. Quest'ultimo dovrà risultare equivalente al nome della zona del DNS, per esempio `incipit.biz`.
Per uscire da nslookup basta digitare `exit` e poi scrivere nuovamente `exit` per chiudere la shell di comandi.

Checklist

1. Installare il pacchetto `bind` dai CD-ROM di installazione del proprio sistema operativo.
2. Verificare se il sistema DNS è installato in una gabbia chroot. In caso affermativo, i file di configurazione indicati di seguito saranno memorizzati nella directory `/var/named/chroot`, anziché in `/etc/bind`.
3. Aprire il file `named.conf`, che si può trovare all'interno della directory `/etc` oppure in `/etc/bind`.
4. Creare le specifiche di ricerca necessarie per la propria installazione, prevedendo una zona di ricerca diretta ed una di ricerca inversa.
5. Creare, all'interno della directory `/etc/bind`, i file per le zone specificate in `named.conf`.
6. Riavviare il servizio con il comando `/etc/init.d/named restart`.
7. Aprire il file `/etc/host.conf` e modificare l'ordine con cui il sistema risolve gli indirizzi, inserendo prima `bind` e poi `hosts`.
8. Aprire il file `/etc/resolv.conf` e indicare l'indirizzo IP del DNS appena configurato: e' sufficiente l'indirizzo di loopback locale `127.0.0.1`.
9. Configurare i client, inserendo l'indirizzo IP del sistema DNS appena creato. Come indirizzi secondari si possono specificare gli indirizzi IP dei DNS del provider.