

Red Hat Linux 8.0

**Official Red Hat Linux Reference
Guide**



Red Hat Linux 8.0: Official Red Hat Linux Reference Guide

Copyright © 2002 Red Hat, Inc.



Red Hat, Inc.

1801 Varsity Drive
Raleigh NC 27606-2072 USA
Telefono: +1 919 754 3700
Telefono: 888 733 4281
Fax: +1 919 754 3701
PO Box 13588
Research Triangle Park NC 27709 Stati Uniti

rhl-rg(IT)-8.0-Print-RHI(2002-08-14T22:29-0400)

Copyright © 2002 Red Hat, Inc. Questo materiale può essere distribuito solo secondo i termini e le condizioni della Open Publication License, V1.0 o successiva (l'ultima versione è disponibile all'indirizzo <http://www.opencontent.org/openpub/>).

La distribuzione di versioni modificate di questo documento è proibita senza esplicita autorizzazione del detentore del copyright.

La distribuzione per scopi commerciali del libro o di una parte di esso sotto forma di opera stampata è proibita se non autorizzata dal detentore del copyright.

Red Hat, Red Hat Network, il logo Red Hat "Shadow Man", RPM, Maximum RPM, il logo RPM, Linux Library, PowerTools, Linux Undercover, RHmember, RHmember More, Rough Cuts, Rawhide e tutti i logo e i marchi registrati di Red Hat sono marchi o marchi registrati di Red Hat, Inc. negli Stati Uniti e in altri paesi.

Linux è un marchio registrato di Linus Torvalds.

Motif e UNIX sono marchi registrati di The Open Group.

Intel e Pentium sono marchi registrati di Intel Corporation. Itanium e Celeron sono marchi di Intel Corporation.

AMD, AMD Athlon, AMD Duron e AMD K6 sono marchi di Advanced Micro Devices, Inc.

Netscape è un marchio registrato di Netscape Communications Corporation negli Stati Uniti e in altri paesi.

Windows è un marchio registrato di Microsoft Corporation.

SSH e Secure Shell sono marchi di SSH Communications Security, Inc.

FireWire è un marchio registrato di Apple Computer Corporation.

Tutti gli altri marchi e diritti sono di proprietà dei rispettivi proprietari.

Il codice GPG della chiave `security@redhat.com` è:

CA 20 86 86 2B D6 9D FC 65 F6 EC C4 21 91 80 CD DB 42 A6 0E

Sommario

Introduzione	ix
1. Modifiche al manuale.....	ix
2. Ricerca della documentazione adatta.....	x
2.1. Documentazione per gli utenti inesperti	x
2.2. Documentazione per i più esperti	xii
2.3. Documentazione per i guru di Linux	xii
3. Convenzioni del documento.....	xii
4. Utilizzo del mouse	xv
5. Copiare e incollare testi con X.....	xv
6. Prossimamente.....	xv
6.1. Inviatemi i vostri suggerimenti!	xv
7. Registrazione per ottenere l'assistenza	xvi
I. Sistema.....	xvii
1. Struttura del filesystem	19
1.1. Perché condividere una struttura comune?	19
1.2. Panoramica sull'FHS (Filesystem Hierarchy Standard)	19
1.3. Posizione dei file speciali.....	23
2. Il filesystem <code>proc</code>	25
2.1. Un filesystem virtuale	25
2.2. File di livello superiore del filesystem <code>proc</code>	26
2.3. Directory in <code>/proc</code>	40
2.4. Utilizzo di <code>sysctl</code>	56
2.5. Risorse aggiuntive.....	56
3. Processo di avvio, <code>init</code> e spegnimento	59
3.1. Il processo di avvio	59
3.2. Un esame dettagliato del processo di avvio	59
3.3. Esecuzione dei programmi all'avvio	64
3.4. Differenze nel processo di avvio di altre architetture	64
3.5. SysV <code>Init</code>	64
3.6. Runlevel <code>init</code>	65
3.7. La directory <code>/etc/sysconfig/</code>	67
3.8. Chiusura del sistema	79
4. Boot loader.....	81
4.1. Boot loader e architettura di sistema.....	81
4.2. GRUB.....	81
4.3. Installazione di GRUB	82
4.4. Terminologia	83
4.5. Interfacce di GRUB	85
4.6. Comandi	86
4.7. File di configurazione del menu GRUB.....	87
4.8. LILO	88
4.9. Opzioni di <code>/etc/lilo.conf</code>	89
4.10. Modifica dei runlevel all'avvio	91
4.11. Risorse aggiuntive.....	91
5. Utenti e gruppi	93
5.1. Strumenti per la creazione di utenti e gruppi	93
5.2. Utenti standard.....	93
5.3. Gruppi standard.....	95
5.4. Gruppi utente privati	96
5.5. Utility shadow	98
6. Il sistema X Window.....	101
6.1. La potenza di X.....	101
6.2. XFree86.....	101
6.3. Ambienti desktop e Window Manager	106

6.4. Runlevel	107
6.5. Font	109
6.6. Risorse aggiuntive	111
II. Sicurezza	113
7. Moduli di autenticazione PAM	115
7.1. I vantaggi dei PAM	115
7.2. File di configurazione PAM	115
7.3. Moduli PAM	116
7.4. Opzioni di controllo PAM	117
7.5. Percorsi dei moduli PAM	118
7.6. Argomenti PAM	118
7.7. Esempi di file di configurazione PAM	118
7.8. PAM e proprietà dei dispositivi	121
7.9. Risorse aggiuntive	121
8. Wrapper TCP e xinetd	123
8.1. Cosa sono i wrapper TCP?	123
8.2. Elenchi di controllo dell'accesso basati sugli host	123
8.3. Controllo dell'accesso tramite xinetd	126
8.4. Risorse aggiuntive	131
9. Protocollo SSH	133
9.1. Caratteristiche di SSH	133
9.2. Sequenza degli eventi di una connessione SSH	134
9.3. Livelli di sicurezza SSH	134
9.4. File di configurazione OpenSSH	136
9.5. Shell più che sicura	137
9.6. Richiesta di SSH per le connessioni remote	139
10. Kerberos	141
10.1. Vantaggi di Kerberos	141
10.2. Svantaggi di Kerberos	141
10.3. Terminologia di Kerberos	141
10.4. Funzionamento di Kerberos	143
10.5. Kerberos e PAM	144
10.6. Configurazione di un server Kerberos 5	144
10.7. Configurazione di un client Kerberos 5	146
10.8. Risorse aggiuntive	146
11. Tripwire	149
11.1. Come usare Tripwire	149
11.2. Installazione dell'RPM di Tripwire	151
11.3. Personalizzazione di Tripwire	152
11.4. Inizializzazione del database di Tripwire	154
11.5. Esecuzione del controllo dell'integrità	155
11.6. Esaminare i report di Tripwire	155
11.7. Aggiornamento del database	157
11.8. Aggiornamento del file di policy di Tripwire	158
11.9. Aggiornamento del file di configurazione di Tripwire	159
11.10. Riferimenti alla posizione del file di Tripwire	160
11.11. Risorse aggiuntive	161

III. Servizi di rete	163
12. Script di rete	165
12.1. File di configurazione per la rete	165
12.2. File di configurazione delle interfacce	165
12.3. Script di controllo delle interfacce	170
12.4. Funzioni di rete	171
12.5. Risorse aggiuntive	171
13. Firewall e iptables	173
13.1. Filtraggio dei pacchetti	173
13.2. Differenze tra iptables e ipchains	174
13.3. Opzioni utilizzate nei comandi di iptables	175
13.4. Memorizzare informazioni relative a iptables	182
13.5. Risorse aggiuntive	182
14. Apache HTTP Server	185
14.1. Apache HTTP Server 2.0	185
14.2. Migrazione dei file di configurazione di Apache HTTP Server 1.3	186
14.3. Dopo l'installazione	196
14.4. Avvio e chiusura di httpd	196
14.5. Direttive di configurazione in httpd.conf	197
14.6. Moduli predefiniti	214
14.7. Aggiungere moduli al server	215
14.8. Utilizzare gli host virtuali	215
14.9. Risorse aggiuntive	217
15. E-mail	219
15.1. Protocolli	219
15.2. Diversi tipi di programmi e-mail	221
15.3. Sendmail	222
15.4. Fetchmail	226
15.5. Procmail	230
15.6. Sicurezza	236
15.7. Risorse aggiuntive	237
16. BIND (Berkeley Internet Name Domain)	241
16.1. Introduzione a DNS e BIND	241
16.2. File di configurazione di BIND	242
16.3. Uso di rndc	252
16.4. BIND: caratteristiche avanzate	254
16.5. Errori comuni da evitare	256
16.6. Risorse aggiuntive	256
17. Network File System (NFS)	259
17.1. Metodologia	259
17.2. File di configurazione del server NFS	261
17.3. File di configurazione del client NFS	263
17.4. Sicurezza e NFS	266
17.5. Risorse aggiuntive	267
18. LDAP (Lightweight Directory Access Protocol)	269
18.1. Perché utilizzare LDAP?	269
18.2. Demoni e utility OpenLDAP	270
18.3. Terminologia LDAP	272
18.4. File di configurazione di OpenLDAP	273
18.5. Panoramica sulla configurazione di OpenLDAP	275
18.6. Aggiornamento alla versione 2.0 di OpenLDAP	275
18.7. Configurare il sistema per l'autenticazione mediante OpenLDAP	276
18.8. Risorse aggiuntive	277

IV. Appendici..... 279

- A. Parametri generali e moduli 281
 - A.1. Come specificare i parametri dei moduli 281
 - A.2. Parametri per i CD-ROM 282
 - A.3. Parametri SCSI..... 284
 - A.4. Parametri Ethernet..... 287

Indice..... 293

Colophon..... 305

Benvenuti nella *Official Red Hat Linux Reference Guide*.

La *Official Red Hat Linux Reference Guide* contiene informazioni utili relative al sistema Red Hat Linux. Dai concetti di base, come la struttura dei filesystem, ad argomenti più complessi, come la sicurezza del sistema e il controllo dell'autenticazione, ci auguriamo che questo libro possa rappresentare una risorsa preziosa.

Vi consigliamo di consultare questa guida se desiderate saperne un po' di più sul funzionamento del sistema Red Hat Linux. Gli argomenti trattati sono i seguenti:

- La struttura del filesystem
- Il processo di avvio
- Il sistema X Window
- Problemi di sicurezza
- Servizi di rete

1. Modifiche al manuale

Questo manuale è stato riorganizzato per chiarezza e ampliato con le nuove caratteristiche di Red Hat Linux 8.0. Tra le modifiche si trovano:

Un nuovo capitolo sui boot loader

Il capitolo su GRUB è stato ampliato per includere LILO.

Un capitolo aggiornato su Apache HTTP Server

È ora disponibile una guida per la migrazione dalla versione 1.3 alla versione 2.0 di Apache HTTP Server. È stato inoltre aggiornato l'elenco di opzioni di configurazione del server. Un ringraziamento speciale a **Gary Benson** e **Joe Orton** per il lavoro svolto per la guida sulla migrazione di Apache HTTP Server.

Un capitolo aggiornato su LDAP

Il capitolo sul protocollo LDAP è stato riorganizzato.

Un capitolo aggiornato sul comando iptables

Il capitolo sul comando iptables è stato riorganizzato.

Un capitolo aggiornato su Tripwire

Il capitolo su Tripwire è stato riorganizzato.

Prima di leggere questa guida, assicuratevi di conoscere i passi relativi all'installazione contenuti nella *Official Red Hat Linux Installation Guide*, i concetti fondamentali contenuti nella *Official Red Hat Linux Getting Started Guide* e le istruzioni relative alla personalizzazione che potete trovare nella *Official Red Hat Linux Customization Guide*. La *Official Red Hat Linux Reference Guide* contiene informazioni su argomenti piuttosto complessi che necessitano di una conoscenza abbastanza approfondita del sistema Red Hat Linux.

Tutti i manuali della versione ufficiale di Red Hat Linux sono disponibili in formato HTML e PDF all'indirizzo <http://www.redhat.com/docs>.



Nota Bene

Sebbene questo manuale contenga le informazioni più aggiornate, è consigliabile leggere le Release Note di Red Hat Linux per reperire eventuali informazioni aggiuntive che non sia stato possibile inserire in questa documentazione. Le Release Note sono disponibili nel CD 1 di Red Hat Linux nonché online all'indirizzo seguente:

<http://www.redhat.com/docs/manuals/linux>

2. Ricerca della documentazione adatta

È consigliabile consultare la documentazione più adatta al vostro livello di conoscenza, altrimenti rischiate di non trovare le informazioni che cercate. La *Official Red Hat Linux Reference Guide* tratta gli aspetti e le opzioni più tecniche del sistema Red Hat Linux. Questa sezione vi aiuterà a stabilire se questo manuale contiene le informazioni di cui avete bisogno oppure se consultare altri manuali Red Hat Linux o risorse online.

Gli utenti di Red Hat Linux possono essere suddivisi in tre gruppi, in base al livello di esperienza. Per ogni "categoria di appartenenza" è indicato il tipo di documentazione da consultare:

Nuovi utenti di Linux

Questi utenti non hanno mai utilizzato un sistema operativo Linux o simile oppure lo conoscono appena, ma potrebbero saper usare altri sistemi operativi (per esempio Windows). Se è il vostro caso consultate la Sezione 2.1.

Utenti con qualche nozione di Linux

Questi utenti hanno già installato e utilizzato Linux in precedenza (ma non Red Hat Linux) oppure hanno un po' di esperienza con altri sistemi operativi simili a Linux. Vi riconoscete in questo tipo di utente? Allora consultate la Sezione 2.2.

Utenti esperti di Linux

Questi utenti hanno installato e usato Red Hat Linux in precedenza. Se appartenete a questa categoria, consultate la Sezione 2.3.

2.1. Documentazione per gli utenti inesperti

Per chi non conosce Linux, la quantità di informazioni disponibili su qualsiasi argomento, come la stampa, l'avvio del sistema o il partizionamento del disco fisso, può sembrare enorme. All'inizio è opportuno raccogliere una base minima di informazioni sul funzionamento di Linux, prima di affrontare argomenti più complessi.

Innanzitutto dovete reperire della documentazione utile. Infatti, senza la documentazione adatta non sarete in grado di far funzionare il vostro sistema Red Hat Linux nel modo desiderato e ciò può divenire fonte di frustrazione.

Dovreste cercare i seguenti tipi di documentazione:

- *Breve storia di Linux* — molti aspetti di Linux sono legati alla sua storia. La cultura di Linux è basata su eventi, necessità e requisiti del passato. Una conoscenza basilare della storia di Linux può aiutarvi a capire come risolvere potenziali problemi, anche prima di incontrarli.

- *Spiegazione sul funzionamento di Linux* — anche se non è necessario investigare gli aspetti più arcani del kernel di Linux, può senz'altro essere utile capire come funziona il "cuore" del sistema. Ciò è particolarmente importante se avete sempre utilizzato altri sistemi operativi, infatti molte delle idee che vi siete fatti sul funzionamento dei computer potrebbero non essere applicabili a Linux.
- *Introduzione ai comandi (con esempi)* — si tratta forse della documentazione più importante per l'uso del sistema Linux, che si basa sulla filosofia secondo cui è meglio utilizzare tanti piccoli comandi collegati in diversi modi piuttosto che avere pochi comandi (complessi) che svolgono l'intero lavoro da soli. Senza esempi che illustrino questo approccio, l'elevato numero di comandi disponibili su Red Hat Linux potrebbe sicuramente intimidirvi.

Ricordatevi che non occorre imparare a memoria tutti i comandi. Esistono diversi modi per facilitare la ricerca del comando specifico di cui avete bisogno per l'esecuzione di un task. È importante conoscere solo il modo generale in cui Linux funziona, cioè il task che vi occorre eseguire e come accedere allo strumento che vi fornisce le istruzioni necessarie per eseguire il comando.

La *Official Red Hat Linux Installation Guide* costituisce un valido riferimento per installare e configurare correttamente il sistema Red Hat Linux. La *Official Red Hat Linux Getting Started Guide* ripercorre la storia di Linux, dei comandi fondamentali di sistema, di GNOME, KDE, RPM e di molti altri concetti fondamentali. Iniziate con questi due libri e utilizzateli come base su cui costituire le vostre conoscenze. Vi accorgete che molti concetti più complessi avranno per voi un senso, poiché avrete già interiorizzato le informazioni di base.

Oltre ai manuali di Red Hat Linux esistono molte altre fonti eccellenti dove trovare informazioni gratuite o poco costose.

2.1.1. Introduzione ai siti Web di Linux

- <http://www.redhat.com> — in questo sito Web è disponibile la documentazione LDP (Linux Documentation Project), le versioni online dei manuali Red Hat Linux, le FAQ (Frequently Asked Questions), un database per aiutarvi nella ricerca del gruppo di utenti Linux più vicino a voi, informazioni tecniche nel Support Knowledge Base di Red Hat e altro ancora.
- <http://www.linuxheadquarters.com> — il sito Web del "quartier generale" di Linux visualizza alcune guide che illustrano passo per passo i numerosi task di Linux.

2.1.2. Introduzione ai newsgroup di Linux

Potete entrare a far parte di un newsgroup leggendo gli interventi di altri, tentando di risolvere i problemi e ponendo domande. Gli esperti di Linux sono sempre disposti ad aiutare i nuovi utenti su varie problematiche di Linux — specialmente se si stanno ponendo le domande nella giusta sede. Se non potete accedere a un'applicazione che permette di leggere le news, visitate il sito <http://www.deja.com>. Esistono comunque decine di newsgroup correlati a Linux, tra cui:

- `linux.help` — un ottimo riferimento in cui ricevere aiuto da altri utenti di Linux.
- `linux.redhat` — questo newsgroup si occupa essenzialmente di tematiche legate a Red Hat Linux.
- `linux.redhat.install` — in questo newsgroup potete porre quesiti relativi all'installazione oppure trovare soluzioni già sperimentate da altri in relazione a problemi simili ai vostri.
- `linux.redhat.misc` — per chi ha domande o richieste che non rientrano nelle categorie tradizionali.
- `linux.redhat.rpm` — per chi ha problemi con l'uso di RPM.

2.1.3. Libri su Linux per inesperti

- *Red Hat Linux for Dummies, 2a edizione* di Jon "maddog" Hall, pubblicato da IDG
- *Special Edition Using Red Hat Linux* di Alan Simpson, John Ray e Neal Jamison, pubblicato da Que
- *Running Linux* di Matt Welsh e Lar Kaufman, pubblicato da O'Reilly & Associates
- *Red Hat Linux 7 Unleashed* di William Ball e David Pitts, pubblicato da Sams

I libri elencati sopra sono un'eccellente fonte di informazione primaria relativa alla conoscenza di base di un sistema Red Hat Linux. Per informazioni più dettagliate relative ai vari argomenti trattati in questo manuale, molti capitoli elencano libri di riferimento specifico, soprattutto le sezioni dedicate alle *Risorse aggiuntive*.

2.2. Documentazione per i più esperti

Se avete già utilizzato altre distribuzioni di Linux, probabilmente avete una conoscenza di base dei comandi più utilizzati. Potreste aver installato il vostro sistema Linux e magari aver scaricato e installato del software reperito su Internet. Tuttavia, dopo l'installazione, potreste avere delle difficoltà con la configurazione.

La *Official Red Hat Linux Customization Guide* è stata ideata per illustrarvi i vari modi in cui il sistema Red Hat Linux può essere configurato per soddisfare le vostre esigenze personali. Usate questo manuale per apprendere tutte le opzioni di configurazione possibili e il modo in cui applicarle.

Se la *Official Red Hat Linux Customization Guide* non contiene le informazioni che cercate, per esempio, su un software che state installando, consultate i documenti HOWTO (LDP), disponibili all'indirizzo <http://www.redhat.com/mirrors/LDP/HOWTO/HOWTO-INDEX/howtos.html>. In queste pagine sono contenute informazioni di tutti i tipi: a partire dal kernel di basso livello fino all'uso di Linux per stazioni radio amatoriali.

2.3. Documentazione per i guru di Linux

Se utilizzate Red Hat Linux da molto tempo, probabilmente saprete già che il modo migliore per capire un particolare programma è leggere il suo codice sorgente e/o i file di configurazione. Uno dei vantaggi di Red Hat Linux è proprio la facilità con cui è possibile leggere il suo codice sorgente.

Ovviamente non tutti sono programmatori in C, dunque il codice sorgente può non essere utile. Comunque se avete le conoscenze e le abilità necessarie per leggerlo, il codice sorgente contiene tutte le risposte alle vostre domande.

3. Convenzioni del documento

Consultando il presente manuale, vedrete alcune parole stampate con caratteri, dimensioni e stili differenti. Si tratta di un metodo sistematico per mettere in evidenza determinate parole; lo stesso stile grafico indica l'appartenenza a una specifica categoria. I tipi di parole rappresentate in questo modo possono essere:

comando

I comandi di Linux (e di altri sistemi operativi) vengono evidenziati così. Questo stile indica che potete digitare la parola o la frase nella linea di comando e premere [Invio] per eseguire

il comando. A volte un comando contiene parole che dovrebbero essere rappresentate con uno stile diverso (come i nomi dei file). In questi casi, tali parole vengono considerate come parte integrante del comando e, dunque, l'intera frase viene visualizzata con lo stile del comando. Per esempio:

Utilizzate il comando `cat testfile` per visualizzare il contenuto di un file chiamato `testfile`, nella directory corrente.

nome del file

I nomi dei file, delle directory, dei percorsi e dei pacchetti RPM vengono rappresentati con questo stile grafico. Ciò significa che un file o una directory particolari hanno questo nome nel sistema Red Hat Linux. Per esempio:

Il file `.bashrc` nella vostra directory home contiene le definizioni e gli alias della shell bash per uso personale.

Il file `/etc/fstab` contiene le informazioni relative ai diversi dispositivi e filesystem di sistema.

Installate il pacchetto RPM `webalizer` per utilizzare un programma di analisi per il file di log del server Web.

applicazione

Questo stile grafico indica che il programma citato è un'applicazione per l'utente finale (contrariamente al software di sistema). Per esempio:

Utilizzate **Mozilla** per navigare sul Web.

[tasto]

I tasti della tastiera sono rappresentati in questo modo. Per esempio:

Per utilizzare la funzionalità [Tab], inserite una lettera e poi premete il tasto [Tab]. Viene visualizzato l'elenco dei file che iniziano con quella lettera.

[tasto]-[combinazione]

Una combinazione di tasti viene rappresentata in questo modo. Per esempio:

La combinazione [Ctrl]-[Alt]-[Backspace] chiude la sessione grafica e vi riporta alla schermata di login o nella console.

testo presente in un'interfaccia grafica

Un titolo, una parola o una frase di una schermata o di una finestra dell'interfaccia grafica, viene mostrato con questo stile: serve per identificare una particolare schermata o elemento dell'interfaccia grafica, per esempio il testo associato a una casella di controllo o a un campo). Qualche esempio:

Selezionate la casella di controllo **Richiedi password** se desiderate che lo screen saver richieda una password prima di scomparire.

livello superiore di un menu o di una finestra dell'interfaccia grafica

Quando vedete una parola scritta con questo stile grafico, si tratta della parola posta al livello superiore di un menu a tendina. Facendo clic sulla parola nella schermata dell'interfaccia grafica, dovrebbe comparire il resto del menu. Per esempio:

In corrispondenza di **File** in un terminale di GNOME è visualizzata l'opzione **Nuova scheda** che vi consente di aprire più prompt della shell nella stessa finestra.

Se dovete digitare una sequenza di comandi da un menu dell'interfaccia grafica, vi compare uno stile simile al seguente esempio:

Per avviare l'editor di testo **Emacs** fate clic sul **pulsante del menu principale** (sul pannello) => **Applicazioni** => **Emacs**.

pulsante di una schermata o una finestra dell'interfaccia grafica

Questo stile indica che il testo si trova su un pulsante in una schermata dell'interfaccia grafica e può essere selezionato con un clic del mouse. Per esempio:

Fate clic sul pulsante **Indietro** per tornare all'ultima pagina Web visualizzata.

output del computer

Quando trovate un testo scritto con questo stile grafico, si tratta del testo visualizzato dal computer sulla linea di comando: risposte a comandi che avete digitato, messaggi di errore e prompt interattivi di richiesta di input nel corso di script o programmi. Per esempio:

Utilizzate il comando `ls` per visualizzare il contenuto di una directory:

```
$ ls
Desktop          axhome          logs            paulwesterberg.gif
Mail             backupfiles     mail            reports
```

L'output restituito dal computer in risposta al comando (in questo caso, il contenuto della directory) viene mostrato con questo stile grafico.

prompt

Questo è lo stile con cui viene visualizzato un prompt, ovvero uno dei modi utilizzati dal computer per indicare che è pronto per ricevere un vostro input. Qualche esempio:

```
$
#
[stephen@maturin stephen]$
leopard login:
```

input dell'utente

Il testo che l'utente deve digitare sulla linea di comando o in un'area di testo di una schermata di un'interfaccia grafica è visualizzato con questo stile, come nell'esempio che segue:

Per avviare il programma di installazione in modalità testuale, dovete digitare il comando **text** al prompt `boot:`.

Inoltre, noi adottiamo diverse strategie per attirare la vostra attenzione su alcune informazioni particolari. In base all'importanza che tali informazioni hanno per il sistema, questi elementi verranno definiti "Nota Bene", "Suggerimento", "Importante", "Attenzione" o "Avvertenza". Per esempio:



Nota Bene

Ricordate che Linux distingue le minuscole dalle maiuscole. In altre parole, una rosa non è una ROSA né una rOSA.



Suggerimento

La directory `/usr/share/doc` contiene documentazione aggiuntiva per i pacchetti installati sul sistema.

**Importante**

Se modificate il file di configurazione DHCP, le modifiche non avranno effetto se non si riavvia il demone DHCP.

**Attenzione**

Non effettuate operazioni standard come utente root. Si consiglia di utilizzare sempre un account utente normale, a meno che non dobbiate amministrare il sistema.

**Avvertenza**

Se decidete di effettuare il partizionamento automatico, l'installazione server rimuove tutte le partizioni esistenti su tutti i dischi fissi installati. Non optate per questo tipo di installazione a meno che siate sicuri di non avere dati da salvare.

4. Utilizzo del mouse

Per Red Hat Linux è previsto l'utilizzo di un mouse a tre tasti. Se avete un mouse a due tasti, dovrete selezionare l'emulazione del terzo tasto durante il processo di installazione. Se avete attivato questa funzione, premendo contemporaneamente i due tasti del mouse ottenete lo stesso effetto dato dalla pressione del terzo tasto (quello centrale).

In questo documento, se vi viene richiesto di fare clic con il mouse su qualche elemento, significa che dovete utilizzare il tasto sinistro. Quando occorre usare il tasto destro o quello centrale, vi viene esplicitamente indicato (qualora il mouse sia stato configurato per un utente mancino le impostazioni sono, naturalmente, invertite).

La locuzione "drag-and-drop" può risultrarvi familiare. Se vi viene indicato di trascinare e lasciare un oggetto all'interno del desktop grafico, dovete fare clic su qualcosa e, tenendo premuto il tasto del mouse, trascinare l'oggetto spostando il mouse in una nuova posizione. Una volta raggiunto il punto desiderato, per depositare l'oggetto dovete semplicemente rilasciare il tasto del mouse.

5. Copiare e incollare testi con X

Copiare e incollare parti di testo è semplice, con il mouse e il sistema X Window. Per copiare un testo, evidenziatelo facendo clic e trascinando il mouse fin dove necessario. Per incollare poi la selezione in un altro punto, fate clic con il tasto centrale del mouse nel punto desiderato.

6. Prossimamente

La *Official Red Hat Linux Reference Guide* fa parte dell'impegno di Red Hat nel fornire un supporto utile e immediato agli utenti di Red Hat Linux. Le prossime edizioni conterranno informazioni più dettagliate inerenti all'amministrazione del sistema, ai tool e ad altre risorse per aiutarvi ad ampliare le potenzialità del vostro sistema Red Hat Linux e la vostra competenza nell'usarlo.

Ovviamente potete contribuire anche voi!

6.1. Inviateci i vostri suggerimenti!

Se individuate refusi di stampa nella *Official Red Hat Linux Reference Guide* o se avete qualche idea per migliorare il manuale, saremmo lieti di raccogliere le vostre informazioni. Inviare un report a Bugzilla (<http://bugzilla.redhat.com/bugzilla>) in merito al componente *rhl-rg*.

Assicuratevi di menzionare l'identificatore del manuale:

```
rhl-rg(IT)-8.0-Print-RHI (2002-08-14T22:29-0400)
```

In questo modo sapremo esattamente a quale manuale vi riferite.

Nel riportare un'imprecisione, cercate di essere il più specifici possibile: indicate il paragrafo e alcune righe di testo, in modo da agevolare la ricerca dell'errore.

7. Registrazione per ottenere l'assistenza

Se siete in possesso di un'edizione ufficiale di Red Hat Linux 8.0, ricordatevi di registrarvi per godere dei benefici che vi spettano in qualità di clienti di Red Hat.

A seconda del prodotto Red Hat Linux ufficiale che avete acquistato, avrete diritto a tutti o ad alcuni dei benefici seguenti:

- Supporto Red Hat ufficiale — il team di supporto vi fornirà assistenza in merito a questioni legate all'installazione.
- Red Hat Network — vi permette di aggiornare facilmente i pacchetti che avete installato e di ricevere avvisi relativi alla sicurezza specifici per il vostro sistema. Per maggiori dettagli, visitate il sito <http://rhn.redhat.com>.
- *Under the Brim: The Official Red Hat E-Newsletter* — ogni mese, riceverete, direttamente da Red Hat, le ultime novità e le informazioni più aggiornate.

Per registrarvi, dovete andare all'indirizzo <http://www.redhat.com/apps/activate/>. Troverete l'ID del prodotto su una scheda di colore nero, rosso e bianco all'interno della vostra confezione di Red Hat Linux ufficiale.

Per maggiori informazioni sull'assistenza tecnica per la versione ufficiale di Red Hat Linux, consultate l'appendice *Come ottenere assistenza tecnica* nella *Official Red Hat Linux Installation Guide*.

Buona fortuna e grazie per aver scelto Red Hat Linux!

Il team di documentazione di Red Hat

Sistema

Struttura del filesystem

1.1. Perché condividere una struttura comune?

La struttura del filesystem di un sistema operativo è il suo livello più elementare di organizzazione. Quasi tutti i modi in cui un sistema operativo interagisce con gli utenti, le applicazioni e i modelli di sicurezza dipendono dal modo in cui il sistema memorizza i suoi file su un dispositivo di memorizzazione. Per svariate ragioni è importante che gli utenti, così come i programmi, possano fare riferimento a una linea guida comune per sapere dove leggere e scrivere i file.

Un filesystem può essere visto come comprendente due diverse categorie logiche di file:

- I file condivisibili e i file non condivisibili
- I file variabili e i file statici

I file *condivisibili* sono file a cui vari host possono accedere, mentre i file *non condivisibili* non sono disponibili per altri host. I file *variabili* possono cambiare in qualsiasi momento senza interventi esterni. I file *statici*, come i file di documentazione di sola lettura o i file binari, non cambiano senza un'azione dell'amministratore del sistema o di chi è autorizzato a effettuare una tale operazione.

La ragione che ci porta a classificare i file è quella di aiutarvi a comprendere il tipo di autorizzazione dato alla directory che contiene i file. Il modo in cui il sistema operativo e i suoi utenti utilizzano i file determina la directory dove questi verranno inseriti, indipendentemente dal fatto che la directory sia montata in modalità di sola lettura o di lettura e scrittura e indipendentemente dal livello di accesso autorizzato a ogni file. Il livello massimo di questa organizzazione è fondamentale, poiché l'accesso alle directory sottostanti può essere limitato o possono insorgere problemi di sicurezza, se il livello massimo è disorganizzato o privo di struttura.

Tuttavia, il fatto di avere una struttura non significa molto a meno che non sia standard. La creazione di strutture rivali rischia di causare problemi anziché risolverli. Per questo motivo, Red Hat ha scelto la struttura di filesystem più comune estendendola leggermente per adattarla a file speciali utilizzati all'interno di Red Hat Linux.

1.2. Panoramica sull'FHS (Filesystem Hierarchy Standard)

Red Hat è fedele al *Filesystem Hierarchy Standard (FHS)*, documento che definisce i nomi e la posizione di molti file e directory. Continueremo a seguire lo standard perché Red Hat Linux sia conforme all'FHS.

L'FHS corrente è il documento di riferimento per qualsiasi filesystem conforme all'FHS, ma lo standard lascia molte zone indefinite ed estensibili. In questa sezione viene fornita una panoramica sullo standard e una descrizione delle parti del filesystem non coperte dallo standard.

Lo standard completo è disponibile all'indirizzo:

<http://www.pathname.com/fhs>

La conformità con lo standard è molto importante, ma i due fattori fondamentali sono la compatibilità con altri sistemi conformi e la capacità di montare una partizione `/usr` come partizione in sola lettura perché contiene file eseguibili comuni e non va modificata dagli utenti. Poiché `/usr` è in sola lettura, può essere montata dal CD-ROM o da un'altra macchina tramite NFS in sola lettura.

1.2.1. Organizzazione dell'FHS

Le directory e i file qui menzionati rappresentano un piccolo sottoinsieme di quelli specificati dal documento FHS. Per informazioni più dettagliate, consultate l'ultimo documento dell'FHS.

1.2.1.1. La directory `/dev`

La directory `/dev` contiene voci del filesystem che rappresentano dispositivi collegati al sistema. Questi file sono essenziali perché il sistema funzioni correttamente.

1.2.1.2. La directory `/etc`

La directory `/etc` è riservata ai file locali di configurazione presenti sul vostro computer. Nessun file binario deve essere inserito in `/etc`. Tutti i file binari che sono stati precedentemente inseriti in `/etc` devono essere trasferiti in `/sbin` oppure in `/bin`.

Le directory `x11` e `skel` sono sottodirectory di `/etc`:

```
/etc
|- X11
|- skel
```

La directory `x11` contiene i file di configurazione di X11, come per esempio `XF86Config`. La directory `skel` contiene i file di base per gli utenti, cioè i file che servono per popolare una directory home quando viene creato un nuovo utente.

1.2.1.3. La directory `/lib`

La directory `/lib` contiene solo le librerie necessarie all'esecuzione dei file binari presenti in `/bin` e `/sbin`. Queste immagini di librerie condivise sono particolarmente importanti per l'avvio del sistema e l'esecuzione di comandi all'interno del filesystem di root.

1.2.1.4. La directory `/mnt`

La directory `/mnt` è riservata ai filesystem montati temporaneamente, come i CD-ROM e i dischetti floppy.

1.2.1.5. La directory `/opt`

La directory `/opt` fornisce un'area per la memorizzazione di pacchetti applicativi statici di grandi dimensioni.

Per evitare di inserire i file di un pacchetto nel filesystem, `/opt` fornisce un sistema organizzativo logico e prevedibile sotto la directory del pacchetto. In questo modo l'amministratore del sistema può facilmente determinare il ruolo di ogni file all'interno di un determinato pacchetto.

Per esempio, se `sample` è il nome di un pacchetto particolare all'interno di `/opt`, allora tutti i suoi file dovrebbero essere inseriti in `/opt/sample`. Per esempio `/opt/sample/bin` per i binari e `/opt/sample/man` per le pagine man.

Anche i pacchetti che comprendono più sotto-pacchetti, ognuno con un compito particolare, vanno inseriti in `/opt` e avranno così un modo standardizzato di organizzarsi. Per esempio, il pacchetto `sample` può avere diversi tool appartenenti ognuno alla propria sottodirectory come `/opt/sample/tool1` e `/opt/sample/tool2`; ognuno di questi può avere la propria directory `bin`, `man` e altre directory simili.

1.2.1.6. La directory `/proc`

La directory `/proc` contiene "file" speciali che estraggono o inviano informazioni al kernel.

Data la svariata quantità di dati disponibili in `/proc` e i vari modi in cui questa directory può essere usata per comunicare con il kernel, è stato dedicato un intero capitolo all'argomento. Per maggiori informazioni, consultate il Capitolo 2.

1.2.1.7. La directory `/sbin`

La directory `/sbin` contiene gli eseguibili utilizzati unicamente dall'utente root. Gli eseguibili in `/sbin` servono solo ad avviare e a montare `/usr` e a eseguire operazioni di ripristino del sistema. L'FHS dice:

"`/sbin` contiene normalmente i file essenziali per l'avvio del calcolatore oltre a quelli presenti in `/bin`. Qualunque altro eseguibile di sistema utilizzato dopo il mount della directory `/usr` (quando non si sono verificati problemi) deve essere collocato in `/usr/sbin`. I file binari riservati all'amministrazione locale del sistema devono essere inseriti in `/usr/local/sbin`."

In `/sbin` trovate almeno i seguenti programmi:

```
arp, clock,
getty, halt,
init, fdisk,
fsck.*, grub,
ifconfig, lilo,
mkfs.*, mkswap,
reboot, route,
shutdown, swapoff,
swapon, update
```

1.2.1.8. La directory `/usr`

La directory `/usr` contiene tutti i file condivisi nello stesso calcolatore (o da una rete). La directory `/usr` ha solitamente una partizione dedicata e dovrebbe essere montata in sola lettura. Quelle riportate di seguito sono le sottodirectory che dovrebbero essere contenute in `/usr`:

```
/usr
|- bin
|- dict
|- doc
|- etc
|- games
|- include
|- kerberos
|- lib
|- libexec
|- local
|- sbin
|- share
|- src
|- tmp -> ../var/tmp
|- X11R6
```

La directory `bin` contiene gli eseguibili, `dict` contiene le pagine di documentazione non conformi a FHS, `etc` contiene i file di configurazione del sistema, `games` quelli per i giochi, `include` contiene i file header di C, `kerberos` contiene i binari e molte altre cose per Kerberos e `lib` contiene file oggetti e librerie che non sono stati concepiti per essere usati direttamente dagli utenti o dagli script della shell. La directory `libexec` contiene piccoli programmi di help richiamati da altri programmi, `sbin` è per i binari di amministrazione del sistema (quelli che non appartengono a `/sbin`), `share` contiene i file non specifici per l'architettura, `src` contiene i codici sorgenti e `X11R6` serve per il sistema X Window (**XFree86** in Red Hat Linux).

1.2.1.9. La directory `/usr/local`

L'FHS dice:

"La gerarchia `/usr/local` viene utilizzata dall'amministratore di sistema quando installa il software a livello locale. Prima che il software sia aggiornato deve essere effettuato un back up di questa directory. Può essere usato per i programmi e i dati che sono condivisibili con altri gruppi di host, ma che non si trovano in `/usr`."

La directory `/usr/local` ha una struttura simile alla directory `/usr`. Contiene le sottodirectory seguenti, che hanno uno scopo simile a quelle contenute nella directory `/usr`:

```
/usr/local
| - bin
| - doc
| - etc
| - games
| - include
| - lib
| - libexec
| - sbin
| - share
| - src
```

1.2.1.10. La directory `/var`

Poiché l'FHS richiede che siate in grado di montare `/usr` in sola lettura, tutti i programmi che scrivono file di log o necessitano delle directory `spool` o `lock` dovrebbero scriverli nella directory `/var`. L'FHS in riferimento a `/var` dice che è per:

"... file di dati variabili. Questa directory contiene i file di spool, di amministrazione, di log e i file temporanei."

Le seguenti directory sono tutte sottodirectory di `/var`:

```
/var
| - account
| - arpwatch
| - cache
| - crash
| - db
| - empty
| - ftp
| - gdm
| - kerberos
```

```

|- lib
|- local
|- lock
|- log
|- mail -> spool/mail
|- mailman
|- named
|- nis
|- opt
|- preserve
|- run
+- spool
    |- anacron
    |- at
    |- cron
    |- fax
    |- lpd
    |- mail
    |- mqueue
    |- news
    |- rwho
    |- samba
    |- slrnpull
    |- squid
    |- up2date
    |- uucp
    |- uucppublic
    |- vbox
    |- voice
|- tmp
|- tux
|- www
|- yp

```

I file di log del sistema, come messages e lastlog si trovano in `/var/log`. La directory `/var/lib/rpm` contiene anche i database del sistema RPM. I file lock si trovano in `/var/lock`, solitamente in directory particolari per il programma che usa il file. La directory `/var/spool` ha delle sottodirectory per vari sistemi che devono memorizzare file di dati.

1.2.2. `/usr/local` in Red Hat Linux

In Red Hat Linux, l'uso della directory `/usr/local` è leggermente diverso da quello specificato dall'FHS. L'FHS dice che `/usr/local` si dovrebbe trovare nel posto in cui è memorizzato il software che non deve subire aggiornamenti del sistema. Poiché gli aggiornamenti di Red Hat vengono effettuati in modo sicuro con **RPM e Package Management Tool**, non dovete proteggere i file mettendoli in `/usr/local`. Invece, vi raccomandiamo di usare `/usr/local` per il software locale del vostro computer.

Per esempio, supponiamo che avete montato `/usr` tramite l'NFS in sola lettura da un host chiamato `jake`. Se desiderate installare un pacchetto o un programma, ma non avete l'autorizzazione di scrivere su `jake`, installatelo in `/usr/local`. Più avanti, se siete riusciti a convincere l'amministratore di `jake` a installare il programma in `/usr`, potrete disinstallarlo da `/usr/local`.

1.3. Posizione dei file speciali

Red Hat ha esteso leggermente la struttura FHS per adattarsi ad alcuni file speciali utilizzati da Red Hat Linux.

Molti dei file presenti nel *Red Hat Package Manager* si trovano nella directory `/var/lib/rpm/`. Per maggiori informazioni in merito agli RPM, consultate il capitolo intitolato *Gestione di pacchetti con RPM* nella *Official Red Hat Linux Customization Guide*.

La directory `/var/spool/updates/` contiene i file utilizzati dal **Red Hat Update Agent**, contenente le informazioni relative agli header RPM per il sistema. Questa directory può anche essere utilizzata per memorizzare temporaneamente gli RPM scaricati durante l'aggiornamento del sistema. Per maggiori informazioni relative a Red Hat Network, consultate il relativo sito all'indirizzo <https://rhn.redhat.com/>.

Un'altra directory specifica di Red Hat Linux è `/etc/sysconfig`, dove vengono conservate le informazioni relative alla configurazione. Molti script eseguiti durante l'avvio usano i file contenuti in questa directory. Per maggiori informazioni sul contenuto di questa directory e sul ruolo che questi file svolgono nell'ambito del processo di avvio, consultate la Sezione 3.7.

Infine, un'altra directory degna di nota è `/initrd/`. Si tratta di una directory vuota, ma viene utilizzata come mount point cruciale durante il processo di avvio.



Avvertenza

Non eliminate per nessuna ragione la directory `/initrd/`, altrimenti non sarà più possibile avviare il sistema e comparirà un messaggio di errore relativo al kernel.

Il filesystem `proc`

Il kernel di Linux ha due funzioni principali: controllare l'accesso ai dispositivi del computer e stabilire quando e come i vari processi interagiscono con tali dispositivi. La directory `/proc` contiene una gerarchia di file speciali che rappresentano lo stato corrente del kernel e consentono alle applicazioni e agli utenti di esplorare il sistema attraverso il punto di vista del kernel.

All'interno della directory `/proc` potete trovare numerose informazioni sull'hardware di sistema e su qualsiasi processo attualmente in esecuzione. Inoltre alcuni file all'interno dell'albero di directory `/proc` possono essere manipolati dagli utenti e dalle applicazioni per comunicare al kernel eventuali modifiche di configurazione.

2.1. Un filesystem virtuale

In Linux tutto viene memorizzato sotto forma di file. La maggior parte degli utenti hanno familiarità con i due principali tipi di file, ovvero i file di testo e binari. La directory `/proc/`, tuttavia, contiene un altro tipo di file denominato *file virtuale*. Per questo motivo `/proc/` è spesso indicato come *filesystem virtuale*.

Questi file virtuali hanno qualità univoche e la maggior parte di essi ha dimensioni pari a zero byte, ma quando visualizzati, possono contenere una grande quantità di informazioni. Inoltre gran parte delle impostazioni dell'ora e della data dei file virtuali riflette la data e l'ora correnti, per indicare il continuo mutamento.

I file virtuali come `interrupts`, `/proc/meminfo`, `/proc/mounts` e `/proc/partitions` forniscono una visualizzazione corrente dell'hardware del sistema, mentre altri, come `/proc/filesystems` e la directory `/proc/sys/` forniscono informazioni sulla configurazione del sistema e interfacce.

A scopo organizzativo i file contenenti informazioni relative a un argomento simile sono raggruppati in directory virtuali e in sottodirectory. Per esempio, `/proc/ide/` contiene informazioni per tutti i dispositivi fisici IDE. In modo simile le directory dei processi contengono informazioni su ciascun processo in esecuzione nel sistema.

2.1.1. Visualizzazione di file virtuali

Combinando i comandi `cat`, `more` o `less` con i file presenti all'interno della directory `/proc`, potete accedere immediatamente a un'enorme quantità di informazioni inerenti al sistema. Per esempio, per vedere quale tipo di CPU è presente nel vostro computer, digitate `cat cpuintfo` e vedrete comparire quanto segue:

```
processor : 0
vendor_id : AuthenticAMD
cpu family : 5
model : 9
model name : AMD-K6(tm) 3D+ Processor
stepping : 1
cpu MHz : 400.919
cache size : 256 KB
fdiv_bug : no
hlt_bug : no
f00f_bug : no
coma_bug : no
fpu : yes
```

```
fpu_exception : yes
cpuid level : 1
wp : yes
flags : fpu vme de pse tsc msr mce cx8 pge mmx syscall 3dnow k6_mtrr
bogomips : 799.53
```

Quando visualizzate diversi file virtuali all'interno del filesystem `/proc`, noterete che alcune informazioni risultano immediatamente comprensibili mentre altre parti non sono leggibili. Questo è in parte il motivo per cui esistono utility specifiche, come per esempio `lspci`, `apm`, `free` e `top`, che permettono di estrarre i dati dai file virtuali e visualizzarli in modo comprensibile.



Nota Bene

Alcuni dei file virtuali contenuti nella directory `/proc` sono impostati per risultare leggibili solo all'utente root.

2.1.2. Come cambiare i file virtuali

In generale, la maggior parte dei file virtuali all'interno della directory `/proc` è in sola lettura. Tuttavia, è possibile utilizzare alcuni di essi operando qualche modifica nelle impostazioni del kernel. Questo vale soprattutto per i file contenuti nella sottodirectory `/proc/sys/`.

Per cambiare il valore di un file virtuale, utilizzate il comando `echo` con un simbolo `>` per reindirizzare il nuovo valore al file. Per esempio, per cambiare il vostro nome host potete digitare:

```
echo bob.subgenious.com > /proc/sys/kernel/hostname
```

Altri file si comportano come switch binari o booleani. Per esempio, digitando `cat /proc/sys/net/ipv4/ip_forward`, otterrete 0 oppure 1. 0 indica che il kernel non sta inoltrando pacchetti di rete. Utilizzando il comando `echo` per trasformare il valore del file `ip_forward` in 1, potete abilitare immediatamente l'inoltro dei pacchetti.



Suggerimento

Un altro comando utilizzato per modificare le impostazioni della sottodirectory `/proc/sys/` è `/sbin/sysctl`. Per ulteriori informazioni su questo comando, consultate la Sezione 2.4

Per ottenere un elenco di alcuni dei file di configurazione del kernel disponibili nella directory `/proc/sys/`, consultate la Sezione 2.3.9.

2.2. File di livello superiore del filesystem `proc`

L'elenco seguente riporta alcuni dei file virtuali di livello superiore più utili contenuti nella directory `/proc/`.

**Nota Bene**

Nella maggior parte dei casi il contenuto dei file elencati in questa sezione non sarà uguale a quello dei file del vostro computer, perché gran parte delle informazioni è relativa all'hardware con cui viene eseguito Red Hat Linux.

2.2.1. /proc/apm

Il file fornisce informazioni sullo stato e le opzioni dell'*Advanced Power Management (APM)* sul sistema. Tali informazioni vengono poi utilizzate dal comando `apm`. Se a un alimentatore CA è collegato un sistema non alimentato a batteria, il file virtuale ha più o meno questo aspetto:

```
1.16 1.2 0x07 0x01 0xff 0x80 -1% -1 ?
```

L'esecuzione del comando `apm -v` in tale sistema dà un output simile a questo:

```
APM BIOS 1.2 (kernel driver 1.16)
AC on-line, no system battery
```

Per i sistemi non alimentati a batteria, `apm` può svolgere altre operazioni oltre a mettere la macchina in modalità di standby. Il comando `apm` è molto più utile per i laptop. Per esempio, l'output riportato di seguito deriva dal comando `cat /proc/apm` di un portatile che esegue Red Hat Linux ed è collegato a una presa di corrente:

```
1.16 1.2 0x03 0x01 0x03 0x09 100% -1 ?
```

Quando lo stesso portatile non è più alimentata a corrente da alcuni minuti, il contenuto del file `apm` cambia:

```
1.16 1.2 0x03 0x00 0x00 0x01 99% 1792 min
```

Il comando `apm -v` produce informazioni più utili come quelli riportati di seguito:

```
APM BIOS 1.2 (kernel driver 1.16)
AC off-line, battery status high: 99% (1 day, 5:52)
```

2.2.2. /proc/cmdline

Il file mostra i parametri trasmessi al kernel al momento dell'avvio. Un file di esempio `/proc/cmdline` ha più o meno questo aspetto:

```
ro root=/dev/hda2
```

Ciò significa che il kernel è montato in modalità di sola lettura, come indicato da (`ro`), esterno alla seconda partizione del primo dispositivo IDE (`/dev/hda2`).

2.2.3. /proc/cpuinfo

Il file virtuale identifica il tipo di processore presente sul vostro sistema. Quello riportato di seguito è un esempio di output tipico derivante da `/proc/cpuinfo`:

```
processor      : 0
vendor_id     : AuthenticAMD
```

```

cpu family      : 5
model          : 9
model name     : AMD-K6(tm) 3D+ Processor
stepping      : 1
cpu MHz       : 400.919
cache size    : 256 KB
fdiv_bug     : no
hlt_bug      : no
f00f_bug     : no
coma_bug     : no
fpu          : yes
fpu_exception : yes
cpuid level   : 1
wp           : yes
flags        : fpu vme de pse tsc msr mce cx8 pge mmx syscall 3dnow k6_mtrr
bogomips     : 799.53

```

- `processor` — attribuisce un numero identificativo a ciascun processore. Nel caso ci sia un unico processore, viene visualizzato soltanto 0.
- `cpu family` — è in grado di dirvi che tipo di processore avete nel vostro sistema. Se si tratta di un sistema basato su un processore Intel, per determinare il valore è sufficiente anteporre il numero a "86". Questa operazione è particolarmente utile nel caso si vogliano delle informazioni sull'architettura di un vecchio sistema, come 586, 486 o 386. Poiché per determinate architetture a volte vengono compilati alcuni pacchetti RPM, questo valore vi consente di determinare quale pacchetto installare sul sistema.
- `model name` — fornisce il nome comune del processore, compreso il nome del suo progetto.
- `cpu MHz` — mostra l'esatta velocità in megahertz di quel particolare processore (nell'ordine delle migliaia).
- `cache size` — comunica la quantità di memoria della cache di livello 2 disponibile per il processore.
- `flags` — definisce svariate caratteristiche del processore, per esempio la presenza di una FPU (unità in virgola mobile) e la capacità di elaborare istruzioni MMX.

2.2.4. `/proc/devices`

Il file visualizza i diversi dispositivi a carattere e a blocchi attualmente configurati per l'utilizzo con il kernel. Non include i dispositivi i cui moduli non sono caricati nel kernel. Ecco un esempio dell'output prodotto da questo file:

```

Character devices:
 1 mem
 2 pty
 3 tty
 4 ttyS
 5 cua
 7 vcs
10 misc
14 sound
29 fb
36 netlink
128 ptm
129 ptm

```

```
136 pts
137 pts
162 raw
254 iscsictl
```

```
Block devices:
 1 ramdisk
 2 fd
 3 ide0
 9 md
22 ide1
```

L'output del file `/proc/devices` comprende il numero maggiore e il nome del dispositivo ed è suddiviso in due sezioni principali: `Character devices` e `Block devices`.

I *dispositivi a carattere* sono simili ai *dispositivi a blocchi*, a eccezione di due differenze sostanziali.

Anzitutto, i dispositivi a blocchi dispongono di un buffer per le richieste inviate, grazie al quale possono ordinare tali richieste prima di elaborarle. Ciò si rivela alquanto utile nel caso dei dispositivi creati per immagazzinare informazioni — per esempio i dischi fissi —, poiché ordinando le informazioni prima che vengano scritte sul dispositivo è possibile poi sistamarle in modo più efficiente. Nel caso dei dispositivi a carattere questa operazione preliminare non è necessaria.

In secondo luogo, i dispositivi a blocchi possono inviare e ricevere informazioni in blocchi di una certa dimensione, configurati a seconda del dispositivo. I dispositivi a carattere non inviano i dati in base a una dimensione predefinita.

Per maggiori informazioni in merito ai dispositivi, vedere `/usr/src/linux-2.4/Documentation/devices.txt`.

2.2.5. `/proc/dma`

Il file contiene un elenco dei canali DMA (accesso diretto alla memoria) per il canale ISA in uso nel sistema. Un file di esempio `/proc/dma` ha il seguente aspetto:

```
4: cascade
```

2.2.6. `/proc/execdomains`

Il file elenca quali sono i *formati di eseguibili* attualmente supportati dal kernel di Linux e la gamma di "personalità" che essi supportano.

```
0-0 Linux [kernel]
```

Si potrebbe dire che i formati degli eseguibili rappresentano in un certo senso la "personalità" di un determinato sistema operativo. Dato che altri formati binari, quali Solaris, UnixWare e FreeBSD, possono essere utilizzati con Linux, i programmatori possono modificare il modo in cui il sistema operativo gestisce particolari chiamate di sistema da questi binari cambiando la personalità dell'attività. A eccezione di `PER_LINUX`, i formati degli eseguibili possono essere implementati come moduli caricabili dinamicamente.

2.2.7. `/proc/fb`

Il file contiene un elenco di dispositivi frame buffer, con il numero del dispositivo e l'unità che lo controlla. Un tipico esempio di output di `/proc/fb` per sistemi che contengono dispositivi frame buffer ha il seguente aspetto:

```
0 VESA VGA
```

2.2.8. `/proc/filesystems`

Il file visualizza un elenco dei tipi di filesystem attualmente supportati dal kernel. Ecco un esempio di output prodotto da un file `/proc/filesystems` di un kernel generico:

```
nodev rootfs
nodev bdev
nodev proc
nodev sockfs
nodev tmpfs
nodev shm
nodev pipefs
  ext2
nodev ramfs
  iso9660
nodev devpts
  ext3
nodev autofs
nodev binfmt_misc
```

La prima colonna indica se il filesystem è stato montato su un dispositivo a blocchi; quelli che iniziano con `nodev` non sono montati su un dispositivo a blocchi. Nella seconda colonna sono elencati i nomi dei filesystem supportati.

Il comando `mount` scorre l'elenco dei filesystem disponibili quando non ne è stato specificato alcuno come argomento.

2.2.9. `/proc/interrupts`

Il file registra il numero di interrupt (IRQ) su architettura x86. Un file `/proc/interrupts` ha solitamente questo aspetto:

```
          CPU0
0:   80448940      XT-PIC timer
1:   174412      XT-PIC keyboard
2:         0      XT-PIC cascade
8:         1      XT-PIC rtc
10:   410964      XT-PIC eth0
12:   60330      XT-PIC PS/2 Mouse
14:  1314121      XT-PIC ide0
15:  5195422      XT-PIC ide1
NMI:         0
ERR:         0
```

Nel caso di macchine multiprocessore, questo file può avere un aspetto leggermente diverso:

```
          CPU0          CPU1
0: 1366814704          0      XT-PIC timer
```

```

1:      128      340      IO-APIC-edge keyboard
2:       0       0          XT-PIC cascade
8:       0       1      IO-APIC-edge rtc
12:     5323     5793     IO-APIC-edge PS/2 Mouse
13:      1       0          XT-PIC fpu
16:    11184294  15940594     IO-APIC-level Intel EtherExpress Pro 10/100 Ethernet
20:     8450043  11120093     IO-APIC-level megaraid
30:     10432     10722     IO-APIC-level aic7xxx
31:      23      22      IO-APIC-level aic7xxx
NMI:      0
ERR:      0

```

La prima colonna si riferisce al numero di IRQ. Ogni CPU presente nel sistema ha la propria colonna e il proprio numero di interrupt (IRQ). La colonna successiva indica il tipo di interrupt e l'ultima colonna contiene il nome del dispositivo interessato.

Ogni tipo di interrupt presente in questo file (gli interrupt sono specifici a seconda dell'architettura) ha un significato leggermente diverso. Per le macchine x86, i seguenti valori sono comuni:

- XT-PIC — gli interrupt del vecchio computer AT.
- IO-APIC-edge — il segnale di voltaggio su questo interrupt è in transizione dal basso verso l'alto, creando così una *margin*e dove si verifica l'interrupt, ed è segnalato una sola volta. Questo tipo di interrupt, così come l'interrupt IO-APIC-level, è possibile solo su sistemi con processori della famiglia 586 e successivi.
- IO-APIC-level — genera degli interrupt quando il segnale di voltaggio va verso l'alto, finché poi non torna di nuovo verso il basso.

2.2.10. `/proc/iomem`

Il file mostra la mappa corrente della memoria del sistema per i vari dispositivi:

```

00000000-0009fbff : System RAM
0009fc00-0009ffff : reserved
000a0000-000bffff : Video RAM area
000c0000-000c7fff : Video ROM
000f0000-000fffff : System ROM
00100000-07ffffff : System RAM
  00100000-00291ba8 : Kernel code
  00291ba9-002e09cb : Kernel data
e0000000-e3ffffff : VIA Technologies, Inc. VT82C597 [Apollo VP3]
e4000000-e7ffffff : PCI Bus #01
  e4000000-e4003fff : Matrox Graphics, Inc. MGA G200 AGP
  e5000000-e57ffffff : Matrox Graphics, Inc. MGA G200 AGP
e8000000-e8ffffff : PCI Bus #01
  e8000000-e8ffffff : Matrox Graphics, Inc. MGA G200 AGP
ea000000-ea00007f : Digital Equipment Corporation DECchip 21140 [FasterNet]
  ea000000-ea00007f : tulip
ffff0000-ffffffff : reserved

```

Nella prima colonna sono visualizzati i registri di memoria usati da ciascuno dei diversi tipi di memoria. La seconda colonna indica il tipo di memoria presente all'interno di tali registri e persino quali registri di memoria sono usati dal kernel nella memoria RAM di sistema o, in caso di NIC Ethernet multiporta, i registri di memoria assegnati per ogni porta.

2.2.11. `/proc/ioports`

L'output di `/proc/ioports` fornisce un elenco delle porte registrate che vengono utilizzate per comunicazioni in ingresso o in uscita con un dispositivo. Questo file può essere piuttosto lungo e iniziare in questo modo:

```
0000-001f : dma1
0020-003f : pic1
0040-005f : timer
0060-006f : keyboard
0070-007f : rtc
0080-008f : dma page reg
00a0-00bf : pic2
00c0-00df : dma2
00f0-00ff : fpu
0170-0177 : ide1
01f0-01f7 : ide0
02f8-02ff : serial(auto)
0376-0376 : ide1
03c0-03df : vga+
03f6-03f6 : ide0
03f8-03ff : serial(auto)
0cf8-0cff : PCI conf1
d000-dfff : PCI Bus #01
e000-e00f : VIA Technologies, Inc. Bus Master IDE
    e000-e007 : ide0
    e008-e00f : ide1
e800-e87f : Digital Equipment Corporation DECchip 21140 [FasterNet]
    e800-e87f : tulip
```

La prima colonna indica l'effettivo range dell'indirizzo della porta I/O riservato al dispositivo che si trova nell'elenco della seconda colonna.

2.2.12. `/proc/isapnp`

Il file elenca le schede *PnP (Plug and Play)* su slot ISA nel sistema. Riguarda principalmente le schede audio, ma può includere qualsiasi altro dispositivo. Un file `/proc/isapnp` contenente la voce Soundblaster ha il seguente aspetto:

```
Card 1 'CTL0070:Creative ViBRA16C PnP' PnP version 1.0 Product version 1.0
Logical device 0 'CTL0001:Audio'
Device is not active
Active port 0x220,0x330,0x388
Active IRQ 5 [0x2]
Active DMA 1,5
Resources 0
    Priority preferred
    Port 0x220-0x220, align 0x0, size 0x10, 16-bit address decoding
    Port 0x330-0x330, align 0x0, size 0x2, 16-bit address decoding
    Port 0x388-0x3f8, align 0x0, size 0x4, 16-bit address decoding
    IRQ 5 High-Edge
    DMA 1 8-bit byte-count compatible
    DMA 5 16-bit word-count compatible
    Alternate resources 0:1
        Priority acceptable
        Port 0x220-0x280, align 0x1f, size 0x10, 16-bit address decoding
        Port 0x300-0x330, align 0x2f, size 0x2, 16-bit address decoding
```

```
Port 0x388-0x3f8, align 0x0, size 0x4, 16-bit address decoding
IRQ 5,7,2/9,10 High-Edge
DMA 1,3 8-bit byte-count compatible
DMA 5,7 16-bit word-count compatible
```

Il file può essere piuttosto lungo, a seconda del numero di dispositivi elencati, dei loro requisiti o le loro richieste in relazione alle risorse.

Ogni scheda elenca il proprio nome, il numero della versione di PnP e quello della versione del prodotto. Se un certo dispositivo è attivo e configurato, questo file indicherà anche i suoi numeri di porta e di IRQ. Inoltre, per assicurare una migliore compatibilità la scheda specifica i valori `preferred` e `acceptable` per svariati parametri, allo scopo di consentire alle schede PnP di lavorare evitando conflitti tra porta e IRQ.

2.2.13. `/proc/kcore`

Il file rappresenta la memoria fisica del sistema ed è memorizzato in formato file core. A differenza di molti file `/proc`, `kcore` visualizza la dimensione. Questo valore viene fornito in byte ed equivale alla dimensione della memoria fisica (RAM) usata più 4 KB.

Il contenuto di questo file è progettato per essere esaminato da un debugger, come `gdb` e non è leggibile.



Avvertenza

Non cercate di visualizzare il file virtuale `kcore`. Il contenuto del file altererà l'output testuale sul terminale. Se vi dovesse capitare per errore di visualizzare il file, premete [Ctrl]-[C] per arrestare il processo e digitate poi `reset` per tornare a visualizzare il prompt della linea di comando.

2.2.14. `/proc/kmsg`

Il file è utilizzato per contenere messaggi generati dal kernel. Tali messaggi vengono poi raccolti da altri programmi, come per esempio `/sbin/klogd`.

2.2.15. `/proc/ksyms`

Il file contiene le definizioni dei simboli esportati dal kernel utilizzati dai tool dei moduli per collegare in modo dinamico moduli caricabili.

```
e003def4 speedo_debug [eeepro100]
e003b04c eeepro100_init [eeepro100]
e00390c0 st_template [st]
e002104c RDINDOOR [megaraid]
e00210a4 callDone [megaraid]
e00226cc megaraid_detect [megaraid]
```

La prima colonna elenca l'indirizzo di memoria della funzione del kernel, mentre la seconda colonna si riferisce al nome della funzione e l'ultima colonna indica il nome del modulo caricato.

2.2.16. `/proc/loadavg`

Il file permette di dare uno sguardo al carico medio del processore nel tempo e fornisce dati aggiuntivi utilizzati da `uptime` e da altri comandi. Ecco un esempio di file `loadavg`:

```
0.20 0.18 0.12 1/80 11206
```

Le prime tre colonne misurano il grado di utilizzo della CPU negli ultimi 1, 5 e 10 minuti. La quarta colonna mostra il numero di processi attualmente in esecuzione e il numero totale dei processi. L'ultima colonna visualizza l'ultimo identificativo di processo usato.

2.2.17. `/proc/locks`

Il file visualizza i file attualmente bloccati dal kernel. All'interno di questo file sono contenuti dati interni di debugging del kernel e il suo contenuto può variare notevolmente, a seconda dell'uso del sistema. Un tipico file `/proc/locks` di un sistema carico al minimo ha questo aspetto:

```
1: FLOCK ADVISORY WRITE 807 03:05:308731 0 EOF c2a260c0 c025aa48 c2a26120
2: POSIX ADVISORY WRITE 708 03:05:308720 0 EOF c2a2611c c2a260c4 c025aa48
```

A ciascun blocco viene attribuito un numero, posto all'inizio di ogni linea. La seconda colonna si riferisce alla classe di blocco utilizzata: `FLOCK` indica che il file è stato bloccato, secondo il vecchio stile UNIX, da una chiamata di sistema `flock`, mentre `POSIX` rappresenta il nuovo sistema di bloccaggio POSIX, che si serve della chiamata di sistema `lockf`.

La terza colonna può avere due valori. `ADVISORY` indica che il blocco non impedisce ad altre persone di accedere ai dati, ma si limita a impedire altri tentativi di bloccare gli stessi. `MANDATORY` segnala che non sono permessi altri accessi ai dati mentre il blocco è attivo. La quarta colonna indica se il blocco concede o meno al proprietario (holder) l'accesso `READ` o `WRITE` al file, mentre la quinta colonna mostra l'ID del processo proprietario. La sesta colonna mostra l'ID del file che viene bloccato nel seguente formato: `MAJOR-DEVICE: MINOR-DEVICE: INODE-NUMBER`. La settima colonna indica dove inizia e finisce l'area del file bloccato. Le colonne restanti riportano le strutture interne dei dati del kernel utilizzati per il debugging e possono essere ignorate.

2.2.18. `/proc/mdstat`

Il file contiene l'informazione corrente per configurazioni di dischi multipli (RAID). Se il vostro sistema non dispone di tale configurazione, allora il file `/proc/mdstat` avrà questo aspetto:

```
Personalities :
read_ahead not set
unused devices: <none>
```

Il file rimane nello stato precedente finché non viene creato un RAID software o un dispositivo `md`. In tal caso, potete usare `/proc/mdstat` per farvi un'idea dell'attuale situazione dei vostri dispositivi RAID `mdX`.

Il file `/proc/mdstat` mostra un sistema che presenta `md0` configurato come dispositivo RAID 1. Al momento sta risincronizzando i dischi:

```
Personalities : [linear] [raid1]
read_ahead 1024 sectors
md0: active raid1 sda2[1] sdb2[0] 9940 blocks [2/2] [UU] resync=1% finish=12.3min
algorithm 2 [3/3] [UUU]
unused devices: <none>
```

2.2.19. `/proc/meminfo`

Questo è uno dei file `/proc` più comunemente usati: riporta, infatti, una grande quantità di preziose informazioni in merito all'attuale utilizzo della RAM nel sistema. Un sistema con 256 MB di RAM e 384 MB di spazio di swap potrebbe presentare un file `/proc/meminfo` simile a questo:

```

      total:      used:      free:  shared: buffers:  cached:
Mem:  261709824 253407232 8302592          0 120745984 48689152
Swap:  402997248      8192 402989056
MemTotal:      255576 kB
MemFree:       8108 kB
MemShared:     0 kB
Buffers:      117916 kB
Cached:       47548 kB
Active:       135300 kB
Inact_dirty:   29276 kB
Inact_clean:   888 kB
Inact_target: 0 kB
HighTotal:    0 kB
HighFree:     0 kB
LowTotal:     255576 kB
LowFree:      8108 kB
SwapTotal:    393552 kB
SwapFree:     393544 kB

```

Molte delle informazioni qui riportate sono utilizzate dai comandi `free`, `top` e `ps`. A dire il vero, l'output del comando `free` ha un aspetto simile a quello del contenuto e della struttura di `/proc/meminfo`. Guardando direttamente `/proc/meminfo`, si possono osservare ulteriori dettagli relativi alla memoria:

- `Mem` — visualizza lo stato attuale della RAM fisica presente nel sistema e fornisce un'analisi completa della quantità di memoria totale, usata, rimanente, condivisa, buffer e cache sotto forma di byte utilizzati.
- `Swap` — visualizza la quantità totale, usata e rimanente dello spazio di swap, misurata in byte.
- `MemTotal` — quantità totale di RAM fisica, misurata in kilobyte.
- `MemFree` — quantità di RAM fisica, misurata in kilobyte, ancora inutilizzata dal sistema.
- `MemShared` — con i kernel 2.4 e successivi non viene utilizzata, ma è stata lasciata per la compatibilità con i kernel delle precedenti versioni.
- `Buffers` — quantità di RAM fisica, misurata in kilobyte, utilizzata per i buffer dei file.
- `Cached` — quantità di RAM fisica, misurata in kilobyte, utilizzata come memoria cache.
- `Active` — quantità totale di memoria cache buffer o pagina, misurata in kilobyte, impiegata attivamente.
- `Inact_dirty` — quantità totale di memoria cache buffer o pagina, misurata in kilobyte, che può essere liberata e resa disponibile.
- `Inact_clean` — quantità totale di memoria cache buffer o pagina, misurata in kilobyte, che può decisamente essere liberata e resa disponibile.
- `Inact_target` — quantità netta di allocazioni al secondo, misurata in kilobyte, con media calcolata per un minuto.
- `HighTotal` e `HighFree` — quantità di memoria totale e rimanente che non è mappata direttamente allo spazio del kernel. Il valore `HighTotal` può variare a seconda del tipo di kernel utilizzato.

- `LowTotal` e `LowFree` — quantità di memoria totale e rimanente mappata direttamente allo spazio del kernel. Il valore `LowTotal` può variare a seconda del tipo di kernel utilizzato.
- `SwapTotal` — quantità totale di spazio di swap disponibile, misurata in kilobyte.
- `SwapFree` — quantità totale di spazio di swap rimanente, misurata in kilobyte.

2.2.20. `/proc/misc`

Il file elenca driver misti registrati sul dispositivo principale, il cui numero è 10:

```
135 rtc
    1 psaux
134 apm_bios
```

La prima colonna indica il numero minore di ciascun dispositivo, mentre la seconda colonna mostra il driver in uso.

2.2.21. `/proc/modules`

Il file mostra un elenco di tutti i moduli che sono stati caricati nel kernel. Il suo contenuto varia a seconda della configurazione e dell'uso del sistema, ma dovrebbe essere organizzato in modo analogo all'output di questo file `/proc/modules` di esempio:

```
ide-cd                27008  0 (autoclean)
cdrom                 28960  0 (autoclean) [ide-cd]
soundcore             4100   0 (autoclean)
agpgart               31072  0 (unused)
binfmt_misc           5956   1
iscsi                 32672  0 (unused)
scsi_mod              94424  1 [iscsi]
autofs                10628  0 (autoclean) (unused)
tulip                 48608  1
ext3                  60352  2
jbd                   39192  2 [ext3]
```

La prima colonna contiene il nome del modulo. La seconda si riferisce alla dimensione della memoria di questo modulo, misurata in byte. La terza colonna indica se il modulo è attualmente caricato (1) o non caricato (0). L'ultima colonna indica se il modulo può essere scaricato automaticamente dopo un certo periodo di inattività (`autoclean`) o se non lo si sta usando (`unused`). Un modulo che presenta una linea contenente un nome riportato tra parentesi ([o]) indica che esso dipende da un altro modulo che deve essere presente per poter funzionare.

2.2.22. `/proc/mounts`

Questo file fornisce un elenco rapido di tutti i mount utilizzati dal sistema:

```
rootfs / rootfs rw 0 0
/dev/hda2 / ext3 rw 0 0
/proc /proc proc rw 0 0
/dev/hdal /boot ext3 rw 0 0
none /dev/pts devpts rw 0 0
none /dev/shm tmpfs rw 0 0
none /proc/sys/fs/binfmt_misc binfmt_misc rw 0 0
```

L'output è simile al contenuto di `/etc/mstab`, con la differenza che `/proc/mount` può essere più attuale.

La prima colonna specifica il dispositivo montato e la seconda indica il mountpoint. Nella terza colonna è indicato il tipo di filesystem, mentre la quarta specifica se è montato in modalità di sola lettura (`ro`) oppure lettura-scrittura (`rw`). La quinta e la sesta colonna riportano dei valori fittizi creati in modo da corrispondere al formato in uso in `/etc/mstab`.

2.2.23. `/proc/mtrr`

Il file si riferisce agli MTRR (Memory Type Range Registers) in uso con il sistema. Se l'architettura del vostro sistema ha il supporto per gli MTRR, il file `/proc/mtrr` avrà all'incirca questo aspetto:

```
reg00: base=0x00000000 ( 0MB), size= 64MB: write-back, count=1
```

Gli MTRR vengono utilizzati con i processori Intel della famiglia P6 (Pentium II e successivi) per controllare l'accesso del processore ai range di memoria. Usando una scheda video su bus PCI o AGP, un file `/proc/mtrr` configurato correttamente può accrescere le prestazioni oltre il 150%.

Il valore è quasi sempre già configurato in modo corretto. Per maggiori informazioni sugli MTRR e la configurazione manuale di questo file, consultate la pagina Web <http://web1.linuxhq.com/kernel/v2.3/doc/mtrr.txt.html>.

2.2.24. `/proc/partitions`

Molte delle informazioni qui riportate sono poco importanti per gran parte degli utenti, a eccezione delle linee che seguono:

- `major` — il numero maggiore del dispositivo con questa partizione. Il numero maggiore nel nostro esempio (3) corrisponde al dispositivo `ide0` in `/proc/devices`.
- `minor` — il numero minore del dispositivo con questa partizione. Serve a separare le partizioni in dispositivi fisici differenti e si riferisce al numero posto alla fine del nome della partizione.
- `#blocks` — elenca il numero dei blocchi fisici del disco contenuti in una determinata partizione.
- `name` — nome della partizione.

2.2.25. `/proc/pci`

Questo file contiene un elenco completo di tutti i dispositivi PCI presenti sul sistema. A seconda del numero di tali dispositivi, `/proc/pci` può raggiungere una discreta lunghezza. Ecco qui un esempio dell'aspetto di questo file su un sistema di base:

```
Bus 0, device 0, function 0:
  Host bridge: Intel Corporation 440BX/ZX - 82443BX/ZX Host bridge (rev 3).
  Master Capable. Latency=64.
  Prefetchable 32 bit memory at 0xe4000000 [0xe7fffffff].
Bus 0, device 1, function 0:
  PCI bridge: Intel Corporation 440BX/ZX - 82443BX/ZX AGP bridge (rev 3).
  Master Capable. Latency=64. Min Gnt=128.
Bus 0, device 4, function 0:
  ISA bridge: Intel Corporation 82371AB PIIX4 ISA (rev 2).
Bus 0, device 4, function 1:
  IDE interface: Intel Corporation 82371AB PIIX4 IDE (rev 1).
  Master Capable. Latency=32.
```

```

I/O at 0xd800 [0xd80f].
Bus 0, device 4, function 2:
  USB Controller: Intel Corporation 82371AB PIIX4 USB (rev 1).
  IRQ 5.
  Master Capable. Latency=32.
  I/O at 0xd400 [0xd41f].
Bus 0, device 4, function 3:
  Bridge: Intel Corporation 82371AB PIIX4 ACPI (rev 2).
  IRQ 9.
Bus 0, device 9, function 0:
  Ethernet controller: Lite-On Communications Inc LNE100TX (rev 33).
  IRQ 5.
  Master Capable. Latency=32.
  I/O at 0xd000 [0xd0ff].
  Non-prefetchable 32 bit memory at 0xe3000000 [0xe30000ff].
Bus 0, device 12, function 0:
  VGA compatible controller: S3 Inc. ViRGE/DX or /GX (rev 1).
  IRQ 11.
  Master Capable. Latency=32. Min Gnt=4.Max Lat=255.
  Non-prefetchable 32 bit memory at 0xdc000000 [0xdfffffff].

```

L'output mostra un elenco di tutti i dispositivi PCI nell'ordine bus, dispositivo e funzione. Oltre a riportare il nome e la versione del dispositivo questo elenco fornisce anche informazioni dettagliate sull'IRQ in modo da snellire la ricerca di conflitti.



Suggerimento

Per ottenere una versione più leggibile di queste informazioni, digitate:

```
lspci -vb
```

2.2.26. `/proc/slabinfo`

Il file fornisce informazioni sull'uso della memoria a livello dello slab. I kernel di Linux di versione superiore a 2.2 utilizzano i *pool di slab* per gestire la memoria oltre il livello di pagina. Gli oggetti comunemente utilizzati dispongono di pool di slab propri. Quella riportata di seguito è una porzione di un tipico file virtuale `/proc/slabinfo/`:

```

slabinfo - version: 1.1
kmem_cache      64      68    112     2     2     1
nfs_write_data   0         0   384     0     0     1
nfs_read_data    0    160   384     0    16     1
nfs_page         0    200    96     0     5     1
ip_fib_hash     10    113    32     1     1     1
journal_head    51   7020    48     2    90     1
revoke_table     2    253    12     1     1     1
revoke_record    0         0    32     0     0     1
clip_arp_cache   0         0   128     0     0     1
ip_mrt_cache     0         0    96     0     0     1

```

I valori contenuti nel file seguono l'ordine nome cache, numero di oggetti attivi, numero totale degli oggetti, dimensioni dell'oggetto, numero degli slab (blocchi) attivi degli oggetti, numero totale degli slab degli oggetti e numero delle pagine per ogni slab.

Va sottolineato che, in questo caso, *attivo* significa in uso. Un oggetto attivo è un oggetto in uso e uno slab attivo è uno slab che contiene qualsiasi oggetto utilizzato.

2.2.27. `/proc/stat`

Il file tiene traccia di svariate statistiche relative al sistema dal momento dell'ultimo riavvio. Il contenuto di `/proc/stat`, che può raggiungere una discreta lunghezza, inizia all'incirca in questo modo:

```
cpu 1139111 3689 234449 84378914
cpu0 1139111 3689 234449 84378914
page 2675248 8567956
swap 10022 19226
intr 93326523 85756163 174412 0 3 3 0 6 0 1 0 428620 0 60330 0 1368304 5538681
disk_io: (3,0):(1408049,445601,5349480,962448,17135856)
ctxt 27269477
btime 886490134
processes 206458
```

Tra le statistiche più diffuse trovate:

- `cpu` — calcola il numero di *istanti* (centesimi di secondo) in cui il sistema è stato in user mode, user mode con priorità bassa (*nice*), system mode e idle task, rispettivamente. Il totale per tutte le CPU viene riportato all'inizio e sotto vengono mostrate le singole CPU con le relative statistiche.
- `page` — numero di pagine di memoria che il sistema ha utilizzato all'interno e all'esterno del disco.
- `swap` — numero di pagine di swap raccolte e liberate dal sistema.
- `intr` — numero degli interrupt verificatisi nel sistema.
- `btime` — tempo di avvio, misurato in numero di secondi, a partire dal 1 gennaio 1970 (noto anche come *epoca*).

2.2.28. `/proc/swaps`

Il file misura lo spazio di swap e il suo utilizzo. Per sistemi che hanno un'unica partizione di swap, l'output del file `/proc/swaps` ha all'incirca questo aspetto:

```
Filename  Type  Size Used Priority
/dev/hda6                partition 136512 20024 -1
```

Queste informazioni si possono reperire anche in altri file disponibili nella directory `/proc`, ma `/proc/swaps` fornisce una rapida istantanea del nome di ogni file di swap, del tipo di spazio di swap e delle dimensioni totali usate in kilobyte. La colonna della priorità (*Priority*) è utile quando vengono utilizzati file di swap multipli. Quanto più bassa è la priorità, tanto maggiore è la probabilità che il file di swap venga utilizzato.

2.2.29. `/proc/uptime`

Il file indica da quanto tempo il computer è acceso dal momento dell'ultimo riavvio. L'output di `/proc/uptime` è piuttosto ridotto:

```
350735.47 234388.90
```

Il primo numero indica il numero totale dei secondi trascorsi dall'accensione del sistema, mentre l'altro indica per quanti di quei secondi la macchina è rimasta inattiva.

2.2.30. `/proc/version`

Il file indica le versioni del kernel di Linux, di `gcc` e di Red Hat Linux installate sul sistema:

```
Linux version 2.4.18-0.4 (user@foo.redhat.com) (gcc version 2.96 20000731
(Red Hat Linux 7.2 2.96-106)) #1 Tue May 28 04:28:05 EDT 2002
```

Queste informazioni servono per diversi scopi, tra cui quello di fornire i dati relativi alla versione al prompt di login.

2.3. Directory in `/proc`

Gruppi comuni di informazioni relative al kernel vengono raccolti in directory e sottodirectory all'interno di `/proc`.

2.3.1. Directory di processo

Ciascuna directory di `/proc` contiene alcune directory identificate con un numero. Un elenco di queste directory inizia nel modo seguente:

```
dr-xr-xr-x 3 root root 0 Feb 13 01:28 1
dr-xr-xr-x 3 root root 0 Feb 13 01:28 1010
dr-xr-xr-x 3 xfs xfs 0 Feb 13 01:28 1087
dr-xr-xr-x 3 daemon daemon 0 Feb 13 01:28 1123
dr-xr-xr-x 3 root root 0 Feb 13 01:28 11307
dr-xr-xr-x 3 apache apache 0 Feb 13 01:28 13660
dr-xr-xr-x 3 rpc rpc 0 Feb 13 01:28 637
dr-xr-xr-x 3 rpcuser rpcuser 0 Feb 13 01:28 666
```

Queste directory vengono chiamate *directory di processo*, poiché si riferiscono all'ID di un processo e contengono informazioni specifiche relative a quel processo. Il proprietario e il gruppo di ciascuna di queste directory è impostato per l'utente che sta eseguendo quel dato processo. Una volta terminato, la sua directory `/proc` scompare.

Ciascuna directory di processo contiene i file seguenti:

- `cmdline` — contiene gli argomenti della linea di comando che hanno dato inizio al processo.
- `cpu` — fornisce informazioni specifiche in merito all'utilizzo di ciascuna delle CPU del sistema. Un processo in esecuzione su un sistema DualCPU produce un output simile a questo:


```
cpu 11 3
cpu0 0 0
cpu1 11 3
```
- `cwd` — collegamento con la directory attualmente in funzione per il processo.
- `environ` — fornisce un elenco delle variabili di ambiente per il processo. La variabile di ambiente viene data in caratteri maiuscoli e il valore in caratteri minuscoli.
- `exe` — collegamento con l'eseguibile di questo processo.
- `fd` — directory contenente tutti i descrittori dei file per un particolare processo. Vengono forniti sotto forma di collegamenti numerati:

```
total 0
lrwx----- 1 root    root      64 May  8 11:31 0 -> /dev/null
lrwx----- 1 root    root      64 May  8 11:31 1 -> /dev/null
lrwx----- 1 root    root      64 May  8 11:31 2 -> /dev/null
lrwx----- 1 root    root      64 May  8 11:31 3 -> /dev/ptmx
lrwx----- 1 root    root      64 May  8 11:31 4 -> socket:[7774817]
lrwx----- 1 root    root      64 May  8 11:31 5 -> /dev/ptmx
lrwx----- 1 root    root      64 May  8 11:31 6 -> socket:[7774829]
lrwx----- 1 root    root      64 May  8 11:31 7 -> /dev/ptmx
```

- `maps` — contiene mappe di memoria per i vari eseguibili e per i file di libreria associati a questo processo. Il file può essere piuttosto lungo, a seconda della complessità del processo. In ogni caso, un esempio di output tratto dal processo `sshd` inizia in questo modo:

```
08048000-08086000 r-xp 00000000 03:03 391479    /usr/sbin/sshd
08086000-08088000 rw-p 0003e000 03:03 391479    /usr/sbin/sshd
08088000-08095000 rwxp 00000000 00:00 0
40000000-40013000 r-xp 00000000 03:03 293205    /lib/ld-2.2.5.so
40013000-40014000 rw-p 00013000 03:03 293205    /lib/ld-2.2.5.so
40031000-40038000 r-xp 00000000 03:03 293282    /lib/libpam.so.0.75
40038000-40039000 rw-p 00006000 03:03 293282    /lib/libpam.so.0.75
40039000-4003a000 rw-p 00000000 00:00 0
4003a000-4003c000 r-xp 00000000 03:03 293218    /lib/libdl-2.2.5.so
4003c000-4003d000 rw-p 00001000 03:03 293218    /lib/libdl-2.2.5.so
```

- `mem` — memoria occupata dal processo. Questo file non può essere letto dall'utente.
- `root` — collegamento con la directory `root` del processo.
- `stat` — stato del processo.
- `statm` — stato della memoria utilizzata dal processo. I file `statm` hanno all'incirca questo aspetto:
263 210 210 5 0 205 0

Le sette colonne si riferiscono a differenti statistiche di memoria per il processo. Nell'ordine in cui sono disposte, da sinistra a destra, riportano diversi aspetti della memoria utilizzata:

1. Dimensioni totali del programma, in kilobyte
2. Dimensioni di porzioni di memoria, in kilobyte
3. Numero di pagine condivise
4. Numero di pagine di codice
5. Numero di pagine di dati/stack
6. Numero di pagine di libreria
7. Numero di pagine marcate dirty

- `status` — fornisce lo stato del processo in forma molto più leggibile rispetto a `stat` o `statm`. L'output per `sshd` ha un aspetto simile a questo:

```
Name: sshd
State: S (sleeping)
Tgid: 797
Pid: 797
PPid: 1
TracerPid: 0
Uid: 0 0 0 0
Gid: 0 0 0 0
FDSize: 32
Groups:
VmSize:      3072 kB
VmLck:       0 kB
```

```

VmRSS:      840 kB
VmData:     104 kB
VmStk:       12 kB
VmExe:       300 kB
VmLib:      2528 kB
SigPnd: 0000000000000000
SigBlk: 0000000000000000
SigIgn: 8000000000001000
SigCgt: 000000000014005
CapInh: 0000000000000000
CapPrm: 00000000ffffffeff
CapEff: 00000000ffffffeff

```

Oltre al nome e all'ID del processo, sono disponibili anche lo stato, come per esempio `S` (`sleeping`) o `R` (`running`), e l'ID dell'utente/gruppo che sta eseguendo il processo. Si possono inoltre reperire dati molto più dettagliati relativi all'utilizzo della memoria.

2.3.1.1. `/proc/self/`

La directory `/proc/self` è un collegamento al processo attualmente in esecuzione. Consente al processo di autoesaminarsi senza dover conoscere il proprio ID.

All'interno di un ambiente di shell, un elenco della directory `/proc/self` produce lo stesso contenuto di un elenco della directory di processo per quel processo.

2.3.2. `/proc/bus/`

La directory contiene informazioni specifiche per i vari bus disponibili sul sistema. Dunque, per esempio, su un sistema standard contenente bus ISA, PCI e USB, i dati correnti su ciascuno di questi bus sono disponibili nella directory corrispondente sotto `/proc/bus`.

I contenuti delle sottodirectory e dei file disponibili variano notevolmente sulla configurazione del sistema. Tuttavia, ognuna delle directory per ciascun tipo di bus contiene almeno una directory per ogni bus di quel tipo. Queste directory di bus singoli, generalmente identificate con dei numeri, come per esempio `00`, contengono file binari che si riferiscono ai vari dispositivi disponibili su quel bus.

Dunque, per esempio, un sistema con bus USB ma senza alcun dispositivo USB collegato ha una directory `/proc/bus/usb` contenente diversi file:

```

total 0
dr-xr-xr-x   1 root   root           0 May  3 16:25 001
-r--r--r--   1 root   root           0 May  3 16:25 devices
-r--r--r--   1 root   root           0 May  3 16:25 drivers
[root@thoth /]# ls -l /proc/bus/usb/001
total 1
-rw-r--r--   1 root   root          18 May  3 16:25 001

```

La directory `/proc/bus/usb` contiene dei file che registrano i vari dispositivi su qualunque bus USB e i driver necessari per utilizzarli. La directory `/proc/bus/usb/001` contiene tutti i dispositivi presenti sul primo (e unico) bus USB. Guardando il contenuto del file `devices`, si può osservare che si tratta della periferica USB root hub sulla scheda madre:

```

T: Bus=01 Lev=00 Prnt=00 Port=00 Cnt=00 Dev#= 1 Spd=12 MxCh= 2
B: Alloc= 0/900 us ( 0%), #Int= 0, #Iso= 0
D: Ver= 1.00 Cls=09(hub ) Sub=00 Prot=00 MxPS= 8 #Cfgs= 1
P: Vendor=0000 ProdID=0000 Rev= 0.00
S: Product=USB UHCI Root Hub

```

```
S:  SerialNumber=d400
C:* #Ifs= 1 Cfg#= 1 Atr=40 MxPwr= 0mA
I:  If#= 0 Alt= 0 #EPs= 1 Cls=09(hub ) Sub=00 Prot=00 Driver=hub
E:  Ad=81(I) Atr=03(Int.) MxPS= 8 Iv1=255ms
```

2.3.3. `/proc/driver/`

Questa directory contiene le informazioni per driver specifici usati dal kernel.

Un file comune di questa directory è `rtc`, che fornisce l'output del driver per l'*orologio tempo reale (RTC)* del sistema, il dispositivo che segna il tempo quando il sistema è spento. L'output prodotto da `/proc/driver/rtc` ha un aspetto simile a questo:

```
rtc_time : 01:38:43
rtc_date : 1998-02-13
rtc_epoch : 1900
alarm    : 00:00:00
DST_enable : no
BCD      : yes
24hr     : yes
square_wave : no
alarm_IRQ : no
update_IRQ : no
periodic_IRQ : no
periodic_freq : 1024
batt_status : okay
```

Per maggiori informazioni circa l'RTC, consultate il file `/usr/src/linux-2.4/Documentation/rtc.txt`.

2.3.4. `/proc/fs`

Questa directory indica quali filesystem vengono esportati. Se è disponibile un server NFS, potete digitare `cat /proc/fs/nfs/exports` per visualizzare i filesystem condivisi e le autorizzazioni concesse a tali filesystem. Per ulteriori informazioni sulla condivisione dei filesystem con NFS, consultate il Capitolo 17.

2.3.5. `/proc/ide/`

Questa directory contiene informazioni sui dispositivi IDE presenti sul sistema. Ogni canale IDE viene rappresentato come directory separata, per esempio `/proc/ide/ide0` e `/proc/ide/ide1`. È inoltre disponibile un file `drivers`, che fornisce il numero della versione dei vari driver utilizzati sui canali IDE:

```
ide-cdrom version 4.59
ide-floppy version 0.97
ide-disk version 1.10
```

Molti chipset forniscono anche un file informativo in questa directory che riporta dati aggiuntivi sulle unità connesse tramite i vari canali. Per esempio un chipset generico Intel PIIX4 Ultra 33 produce un file `/proc/ide/piix` che indicherà se i protocolli DMA o UDMA sono attivati per i dispositivi sui canali IDE:

```

                                Intel PIIX4 Ultra 33 Chipset.
----- Primary Channel ----- Secondary Channel -----
                                enabled
----- drive0 ----- drive1 ----- drive0 ----- drive1 -----
DMA enabled:   yes           no           yes           no
UDMA enabled:  yes           no           no            no
UDMA enabled:  2             X           X             X
UDMA
DMA
PIO

```

Esplorando la directory in cerca di un canale IDE, come `ide0`, si possono ottenere informazioni aggiuntive. Il file `channel` fornisce il numero del canale, mentre il file `model` indica il tipo di bus per il canale (per esempio `pci`).

2.3.5.1. Directory dei dispositivi

Nella directory del canale IDE è presente una directory dei dispositivi, il cui nome corrisponde alla lettera dell'unità nella directory `/dev` directory. Per esempio, la prima unità IDE su `ide0` sarà `hda`.



Nota Bene

Esiste un collegamento a ciascuna di queste directory dei dispositivi all'interno della directory `/proc/ide/`.

Ciascuna directory dei dispositivi contiene una serie di informazioni e statistiche. Il contenuto delle directory varia a seconda del tipo di dispositivo collegato. Tra i file più utili comuni a diversi dispositivi si trovano:

- `cache` — cache del dispositivo.
- `capacity` — capacità del dispositivo, in blocchi di 512 byte.
- `driver` — l'unità e la versione usate per controllare il dispositivo.
- `geometry` — geometria fisica e logica del dispositivo.
- `media` — tipo di dispositivo, per esempio `disk`.
- `model` — numero o nome del modello del dispositivo.
- `settings` — raccolta dei parametri correnti del dispositivo. Di norma questo file contiene una discreta quantità di informazioni tecniche piuttosto utili. Un file `settings` per un disco fisso IDE standard potrebbe avere un aspetto simile al seguente:

```

name           value           min           max           mode
----           -
bios_cyl       784             0             65535         rw
bios_head     255             0             255           rw
bios_sect     63              0             63            rw
breada_readahead 4              0             127           rw
bswap         0               0             1             r
current_speed 66              0             69            rw
file_readahead 0               0             2097151       rw
ide_scsi      0               0             1             rw
init_speed    66              0             69            rw

```

<code>io_32bit</code>	0	0	3	<code>rw</code>
<code>keepsettings</code>	0	0	1	<code>rw</code>
<code>lun</code>	0	0	7	<code>rw</code>
<code>max_kb_per_request</code>	64	1	127	<code>rw</code>
<code>multcount</code>	8	0	8	<code>rw</code>
<code>nicel</code>	1	0	1	<code>rw</code>
<code>nowerr</code>	0	0	1	<code>rw</code>
<code>number</code>	0	0	3	<code>rw</code>
<code>pio_mode</code>	<code>write-only</code>	0	255	<code>w</code>
<code>slow</code>	0	0	1	<code>rw</code>
<code>unmaskirq</code>	0	0	1	<code>rw</code>
<code>using_dma</code>	1	0	1	<code>rw</code>

2.3.6. `/proc/irq/`

La directory è utilizzata per impostare l'affinità tra IRQ e CPU, che consente di connettere un particolare IRQ a un'unica CPU. Oppure è possibile escludere una CPU dalla gestione degli IRQ.

Ciascun IRQ ha la propria directory, il che gli consente di essere configurato in modo diverso da tutti gli altri. Il file `/proc/irq/prof_cpu_mask` è un bitmask contenente i valori di default per il file `smp_affinity` all'interno della directory IRQ. I valori contenuti in `smp_affinity` specificano quali CPU gestiscono quel particolare IRQ.

Per maggiori informazioni, consultate il file `/usr/src/linux-2.4/Documentation/filesystems/proc.txt`.

2.3.7. `/proc/net/`

Questa directory permette di osservare in modo completo i vari parametri e le varie statistiche della rete. Ciascuno dei suoi file copre una specifica gamma di informazioni relative alla rete del sistema. Di seguito è riportato un elenco parziale di questi file virtuali:

- `arp` — contiene la tabella ARP del kernel. Questo file è particolarmente utile per collegare l'indirizzo hardware a un indirizzo IP sul sistema.
- `atm` — directory contenente file con varie impostazioni *ATM (Asynchronous Transfer Model, modalità di trasferimento asincrona)* e statistiche. Questa directory è usata soprattutto con networking ATM e schede ADSL.
- `dev` — elenca i vari dispositivi di rete configurati sul sistema, corredati di statistiche di trasmissione e ricezione. Questo file indica in modo rapido il numero di byte inviati e ricevuti da ciascuna interfaccia, il numero di pacchetti inbound e outbound, il numero di errori rilevati, il numero dei pacchetti persi e molto altro ancora.
- `dev_mcast` — visualizza i vari gruppi multicast Layer2 su cui ogni dispositivo è in attesa.
- `igmp` — elenca gli indirizzi IP multicast a cui è collegato il sistema.
- `ip_fwchains` — se `ipchains` è in uso, questo file virtuale rivela le regole correnti.
- `ip_fwnames` — se `ipchains` è in uso, questo file virtuale fornisce un elenco di tutti i nomi delle catene del firewall.
- `ip_masquerade` — fornisce una tabella di informazioni sul masquerading in `ipchains`.
- `ip_mr_cache` — elenco della cache del routing multicast.
- `ip_mr_vif` — elenco delle interfacce virtuali del protocollo multicast.

- `netstat` — contiene una raccolta molto dettagliata di statistiche di networking, insieme ai servizi di temporizzazione TCP, ai SYN cookie inviati e ricevuti e molto altro.
- `psched` — elenco di parametri dello schedatore di pacchetti.
- `raw` — elenco di statistiche del dispositivo a carattere.
- `route` — visualizza la tabella di routing del kernel.
- `rt_cache` — contiene la cache di routing corrente.
- `snmp` — elenco di dati relativi al protocollo SNMP (Simple Network Management Protocol) per i vari protocolli di networking in uso.
- `sockstat` — fornisce statistiche per il socket.
- `tcp` — contiene informazioni dettagliate riguardo al socket TCP.
- `tr_rif` — tabella di routing per il RIF token ring.
- `udp` — contiene informazioni dettagliate riguardo al socket UDP.
- `unix` — elenca i socket di dominio UNIX attualmente in uso.
- `wireless` — elenca i dati relativi all'interfaccia wireless.

2.3.8. `/proc/scsi/`

Questa directory è analoga alla directory `/proc/ide/`, ma serve per i dispositivi SCSI collegati.

Il file principale contenuto in questa directory è `/proc/scsi/scsi` che contiene un elenco di tutti i dispositivi SCSI riconosciuti. In questo elenco è disponibile il tipo di dispositivo, il nome del modello, il fornitore, il canale SCSI e i dati ID.

Per esempio, se un sistema contiene un CD-ROM SCSI, un'unità a nastro, dischi fissi e un controller RAID, il file avrà un aspetto simile a questo:

```
Attached devices:
Host: scsi1 Channel: 00 Id: 05 Lun: 00
  Vendor: NEC          Model: CD-ROM DRIVE:466 Rev: 1.06
  Type:   CD-ROM              ANSI SCSI revision: 02
Host: scsi1 Channel: 00 Id: 06 Lun: 00
  Vendor: ARCHIVE      Model: Python 04106-XXX Rev: 7350
  Type:   Sequential-Access ANSI SCSI revision: 02
Host: scsi2 Channel: 00 Id: 06 Lun: 00
  Vendor: DELL         Model: 1x6 U2W SCSI BP  Rev: 5.35
  Type:   Processor              ANSI SCSI revision: 02
Host: scsi2 Channel: 02 Id: 00 Lun: 00
  Vendor: MegaRAID     Model: LD0 RAID5 34556R Rev: 1.01
  Type:   Direct-Access          ANSI SCSI revision: 02
```

Ciascun dispositivo SCSI utilizzato dal sistema ha la propria directory all'interno di `/proc/scsi`, contenente file specifici di ogni controller SCSI che utilizza quel driver. Dunque, in riferimento al sistema citato come esempio prima, sono presenti le directory `aic7xxx` e `megaraid` poiché vengono utilizzati i due driver. I file di ciascuna delle directory contengono, di norma, il range dell'indirizzo IO, l'IRQ e le statistiche relative al controller SCSI che utilizza quel driver. Ogni controller può riportare un diverso tipo e una diversa quantità di informazioni. Il file dell'adattatore per host Adaptec AIC-7880 Ultra SCSI per il sistema preso come esempio produce il seguente output:

```
Adaptec AIC7xxx driver version: 5.1.20/3.2.4
Compile Options:
  TCQ Enabled By Default : Disabled
  AIC7XXX_PROC_STATS     : Enabled
```

AIC7XXX_RESET_DELAY : 5

Adapter Configuration:

```

SCSI Adapter: Adaptec AIC-7880 Ultra SCSI host adapter
               Ultra Narrow Controller
PCI MMAPed I/O Base: 0xfcffe000
Adapter SEEPROM Config: SEEPROM found and used.
Adaptec SCSI BIOS: Enabled
IRQ: 30
SCBs: Active 0, Max Active 1,
      Allocated 15, HW 16, Page 255
Interrupts: 33726
BIOS Control Word: 0x18a6
Adapter Control Word: 0x1c5f
Extended Translation: Enabled
Disconnect Enable Flags: 0x00ff
Ultra Enable Flags: 0x0020
Tag Queue Enable Flags: 0x0000
Ordered Queue Tag Flags: 0x0000
Default Tag Queue Depth: 8
Tagged Queue By Device array for aic7xxx host instance 1:
{255,255,255,255,255,255,255,255,255,255,255,255,255,255,255}
Actual queue depth per device for aic7xxx host instance 1:
{1,1,1,1,1,1,1,1,1,1,1,1,1,1,1}
    
```

Statistics:

(scsil:0:5:0)

```

Device using Narrow/Sync transfers at 20.0 MByte/sec, offset 15
Transinfo settings: current(12/15/0/0), goal(12/15/0/0), user(12/15/0/0)
Total transfers 0 (0 reads and 0 writes)
  < 2K   2K+   4K+   8K+   16K+   32K+   64K+   128K+
Reads:   0     0     0     0     0     0     0     0
Writes:  0     0     0     0     0     0     0     0
    
```

(scsil:0:6:0)

```

Device using Narrow/Sync transfers at 10.0 MByte/sec, offset 15
Transinfo settings: current(25/15/0/0), goal(12/15/0/0), user(12/15/0/0)
Total transfers 132 (0 reads and 132 writes)
  < 2K   2K+   4K+   8K+   16K+   32K+   64K+   128K+
Reads:   0     0     0     0     0     0     0     0
Writes:  0     0     0     1    131     0     0     0
    
```

In questa schermata viene visualizzata la velocità di trasferimento ai vari dispositivi SCSI connessi al controller sulla base del canale ID, oltre a delle statistiche dettagliate inerenti alla quantità e alle dimensioni dei file letti o scritti da quel dispositivo. Per esempio, in relazione all'output precedente, si può notare che questo controller sta comunicando con il CD-ROM a 20 megabyte al secondo, mentre l'unità a nastro è connessa a soli 10 megabyte al secondo.

2.3.9. `/proc/sys/`

La directory `/proc/sys/` è una directory speciale, diversa dalle altre directory presenti in `/proc/`. Infatti, non solo fornisce numerose informazioni relative al sistema, ma consente anche di modificare la configurazione di un kernel. Ciò consente all'amministratore della macchina di abilitare e disabilitare immediatamente le caratteristiche del kernel.



Avvertenza

Fate attenzione quando modificate le impostazioni di un sistema di produzione utilizzando i vari file contenuti nella directory `/proc/sys/`. In seguito alla modifica di un'impostazione errata il kernel può diventare instabile e può dunque essere necessario riavviare il sistema.

Prima di tentare di cambiare un valore nella directory `/proc/sys`, assicuratevi di conoscere le opzioni corrette per quel file e il risultato previsto.

Un buon modo per determinare se un particolare file può essere usato per configurare il kernel oppure se è stato concepito solo per fornire informazioni è quello di estrarne un elenco con il flag `-l/` nel terminale. Se il file può essere scritto, è possibile utilizzarlo per configurare il kernel. Per esempio, un elenco parziale del file `/proc/sys/fs` ha il seguente aspetto:

```
-r--r--r-- 1 root root 0 May 10 16:14 dentry-state
-rw-r--r-- 1 root root 0 May 10 16:14 dir-notify-enable
-r--r--r-- 1 root root 0 May 10 16:14 dquot-nr
-rw-r--r-- 1 root root 0 May 10 16:14 file-max
-r--r--r-- 1 root root 0 May 10 16:14 file-nr
```

In questo elenco i file `dir-notify-enable` e `file-max` possono essere scritti e pertanto è possibile utilizzarli per configurare il kernel. Gli altri file forniscono solamente un feedback in relazione alle attuali impostazioni.

La modifica di un valore all'interno di un file `/proc/sys` viene effettuata ripetendo il nuovo valore nel file. Per esempio, per abilitare il tasto `SysRq` su un kernel in funzione, digitate il comando:

```
echo 1 > /proc/sys/kernel/sysrq
```

In questo modo il valore del file `sysrq` passerà da 0 (off) a 1 (on).

Lo scopo del tasto `SysRq` è quello di consentire all'utente di ordinare al kernel l'immediata esecuzione di numerose importanti attività (come per esempio la chiusura o il riavvio del sistema, la sincronizzazione di tutti i filesystem montati o lo scaricamento di informazioni importanti sulla propria console) digitando una semplice combinazione di tasti. Questa funzionalità risulta particolarmente utile quando si sta utilizzando un kernel in sviluppo o se si verificano "congelamenti" del sistema. Tuttavia, è considerato un rischio di sicurezza per una console imprevista ed è pertanto disattivato di default in Red Hat Linux.

Per maggiori informazioni sul tasto `SysRq`, consultate il file `/usr/src/linux-2.4/Documentation/sysrq.txt`.

Alcuni file di configurazione `/proc/sys` contengono più di un valore. Per inviare a questi file nuovi valori in modo corretto, inserite uno spazio tra ogni valore trasmesso con il comando `echo`, come mostrato in questo esempio:

```
echo 4 2 45 > /proc/sys/kernel/acct
```

**Nota Bene**

Qualsiasi modifica di configurazione effettuata tramite il comando `echo` scomparirà nel momento in cui il sistema verrà riavviato. Per sapere come rendere effettive le modifiche al momento del riavvio del sistema, consultate la Sezione 2.4.

La directory `/proc/sys` contiene svariate sottodirectory che controllano aspetti diversi di un kernel in funzione.

2.3.9.1. `/proc/sys/dev/`

La directory fornisce parametri per particolari dispositivi sul sistema. Molti sistemi hanno almeno due directory, `cdrom` e `raid`, ma i kernel personalizzati ne possono avere altre, come `parport`, che fornisce la capacità di condividere una porta parallela tra diversi driver di dispositivi.

La directory `cdrom` contiene un file chiamato `info`, che mostra una serie di importanti parametri del CD-ROM:

```
CD-ROM information, Id: cdrom.c 3.12 2000/10/18
```

```
drive name: hdc
drive speed: 32
drive # of slots: 1
Can close tray: 1
Can open tray: 1
Can lock tray: 1
Can change speed: 1
Can select disk: 0
Can read multisession: 1
Can read MCN: 1
Reports media changed: 1
Can play audio: 1
Can write CD-R: 0
Can write CD-RW: 0
Can read DVD: 0
Can write DVD-R: 0
Can write DVD-RAM: 0
```

Esaminando rapidamente questo file è possibile scoprire le caratteristiche di un CD-ROM ignoto, almeno agli occhi del kernel. Se si dispone di più CD-ROM su uno stesso sistema, ciascun dispositivo avrà la propria colonna di informazioni.

Svariate file contenuti in `/proc/sys/dev/cdrom`, come `autoclose` e `checkmedia`, possono essere utilizzati per controllare il CD-ROM del sistema. Utilizzate il comando `echo` per abilitare o disabilitare queste caratteristiche.

Se nel kernel è stato compilato il supporto RAID, sarà disponibile la directory `/proc/sys/dev/raid`, che conterrà almeno due file: `speed_limit_min` e `speed_limit_max`. Queste impostazioni determinano in quale misura va aumentata la velocità con cui viene utilizzato il dispositivo RAID per task di I/O particolarmente intensivi, come la risincronizzazione dei dischi.

2.3.9.2. `/proc/sys/fs/`

Questa directory contiene una serie di opzioni e informazioni relative a vari aspetti del filesystem, tra cui informazioni su quota, file handle, inode e dentry.

La directory `binfmt_misc` viene utilizzata per fornire al kernel il supporto per formati binari misti.

I file importanti di `/proc/sys/fs` comprendono:

- `dentry-state` — fornisce lo stato della directory della cache. L'aspetto del file è simile a questo:
57411 52939 45 0 0 0
Il primo numero mostra il numero totale delle voci presenti nella directory della cache, mentre il secondo visualizza il numero delle voci non utilizzate. Il terzo numero indica i secondi che trascorrono tra il momento in cui una directory viene liberata e il momento in cui è possibile "reclamarla", mentre il quarto misura le pagine richieste attualmente dal sistema. Gli ultimi due numeri non sono in uso e al momento visualizzano solo il numero 0.
- `dquot-nr` — mostra il numero massimo di voci di quota disco memorizzato nella cache.
- `file-max` — consente di cambiare il numero massimo di file handle che il kernel può allocare. Aumentando il valore di questo file si possono risolvere eventuali errori derivanti da una carenza di file handle disponibili.
- `file-nr` — visualizza il numero di file handle allocati, il numero di file handle utilizzati e il numero massimo di file handle.
- `overflowgid` e `overflowuid` — definiscono rispettivamente l'ID di gruppo e l'ID utente da utilizzare con filesystem che supportano solo ID utente e di gruppo a 16 bit.
- `super-max` — controlla il numero massimo di superblocchi disponibili.
- `super-nr` — visualizza il numero di superblocchi attualmente in uso.

2.3.9.3. `/proc/sys/kernel/`

Questa directory contiene una serie di file di configurazione che interessano direttamente l'operato del kernel. Tra i file più importanti trovate:

- `acct` — controlla la sospensione della contabilità relativa a un processo sulla base della percentuale di spazio libero disponibile sul filesystem contenente il log. Per default, il file ha un aspetto simile a questo:
4 2 30
Il secondo valore definisce la soglia percentuale di spazio libero alla sospensione del logging, mentre il primo valore stabilisce la percentuale di spazio libero necessaria per riprendere il logging. Il terzo valore definisce l'intervallo in secondi in cui il kernel interroga il filesystem per vedere se il logging deve essere sospeso o ripreso.
- `cap-bound` — controlla le impostazioni del *limite di capacità*. Fornisce un elenco delle azioni che qualsiasi processo sul sistema è in grado di compiere. Se un'azione non è presente in questo elenco, allora nessun processo è in grado di compierla, a prescindere dalla quantità di privilegi di cui dispone. L'idea di fondo, in questo caso, è quella di rendere più sicuro il sistema facendo in modo che determinate situazioni non si verificino durante il processo di avvio, per lo meno a partire da un dato momento.
I diversi possibili valori sono al di fuori dell'ambito di questo manuale, dunque vi consigliamo di consultare la documentazione relativa al kernel qualora desideraste maggiori informazioni.
- `ctrl-alt-del` — controlla se la combinazione di tasti [Ctrl]-[Alt]-[Canc] riavvierà il computer in modo corretto tramite `init` (valore 0) o se provocherà, piuttosto, un riavvio repentino senza sincronizzare i buffer "dirty" con il disco (valore 1).

- `domainname` — consente di configurare il nome di dominio del sistema, come `subgenius.com`.
- `hostname` — consente di configurare il nome host del sistema, per esempio `bob.subgenius.com`.
- `hotplug` — configura l'utility da utilizzare quando il sistema rileva una modifica nella configurazione. Viene utilizzato principalmente con USB e Cardbus PCI. Si consiglia di non modificare il valore predefinito di `/sbin/hotplug`, a meno che non si stia provando un nuovo programma che svolga questo ruolo.
- `modprobe` — definisce la posizione del programma da utilizzare per caricare, quando necessario, i moduli del kernel. Il valore predefinito di `/sbin/modprobe` indica che `kmod` lo chiamerà proprio per caricare il modulo quando un kernel lo richiede (con `kmod`).
- `msgmax` — definisce la dimensione massima dei messaggi inviati da un processo all'altro, che di default è impostata su 8192 byte. Si consiglia di evitare di aumentare questo valore, poiché i messaggi in coda tra i vari processi vengono immagazzinati nella memoria "non swappable" del kernel e un qualsiasi aumento in `msgmax` comporterebbe un aumento della quantità di RAM necessaria per il sistema.
- `msgmnb` — stabilisce il numero massimo di byte consentiti per una singola coda di messaggi. Il numero predefinito è 16384.
- `msgmni` — definisce il numero massimo di identificatori consentiti per una coda di messaggi. Il valore predefinito è 16.
- `osrelease` — elenca il numero di release del kernel di Linux. Questo file può essere modificato solo cambiando il sorgente del kernel e ricompilando.
- `ostype` — visualizza il tipo di sistema operativo. Di default, questo file è impostato su Linux e questo valore può essere modificato solo cambiando il sorgente del kernel e ricompilando.
- `overflowgid` e `overflowuid` — definiscono rispettivamente l'ID di gruppo e l'ID utente da utilizzare con le chiamate di sistema su architetture che supportano soltanto ID utente e di gruppo a 16 bit.
- `panic` — determina di quanti secondi il kernel posticiperà il riavvio del sistema qualora si verificassero dei problemi. Il valore predefinito, che è impostato su 0, disattiva il riavvio automatico in seguito a una crisi del kernel.
- `printk` — questo file controlla una serie di impostazioni relative alla stampa o alla registrazione di messaggi di errore. Ciascun messaggio di errore riportato dal kernel è associato a un *livello di log* che determina l'importanza del messaggio stesso. I valori del livello di log si articolano come segue:
 - 0 — emergenza kernel: il sistema è inutilizzabile.
 - 1 — allarme kernel: è necessario un intervento immediato.
 - 2 — il kernel è in condizioni critiche.
 - 3 — errore generale del kernel.
 - 4 — avvertimento sulle condizioni generali del kernel.
 - 5 — condizioni del kernel normali ma significative.
 - 6 — messaggio informativo riguardo al kernel.
 - 7 — messaggi a livello di debug riguardanti il kernel.

Nel file `printk` sono presenti quattro valori:

```
6 4 1 7
```

Ciascuno di questi valori definisce una regola distinta per la gestione dei messaggi di errore. Il primo valore, che si chiama *livello di log della console*, indica la priorità più bassa dei messaggi che verranno visualizzati sulla console (più è bassa la priorità, più è alto il numero del livello di log). Il secondo valore definisce il valore predefinito del livello di log per i messaggi cui non è

associato un livello di log specificato. Il terzo valore indica la configurazione più bassa per il livello di log della console. Infine, l'ultimo numero indica il valore predefinito per il livello di log della console.

- `rtsig-max` — configura il numero massimo di segnali POSIX realtime che il sistema può tenere in coda alla volta. Il valore predefinito è 1024.
- `rtsig-nr` — il numero attuale di segnali POSIX realtime tenuti in coda dal kernel.
- `sem` — questo file configura impostazioni "semaforiche" all'interno del kernel. Per *semaforo* si intende un oggetto IPC di System V utilizzato per controllare l'uso di un particolare processo.
- `shmall` — stabilisce la quantità totale di memoria condivisa (misurata in byte) che può essere utilizzata sul sistema ogni singola volta. Il valore predefinito è 2097152.
- `shmmax` — stabilisce la dimensione massima (misurata in byte) del segmento di memoria condivisa consentita dal kernel. Il valore predefinito è 33554432, ma il kernel può supportare valori molto più alti di questo.
- `shmni` — stabilisce il numero massimo di segmenti di memoria condivisa per l'intero sistema. Il valore predefinito è 4096.
- `sysrq` — attiva il tasto SysRq se questo valore è impostato su un numero diverso da quello di default, ovvero 0. Per informazioni sul tasto SysRq, consultate la Sezione 2.3.9.
- `threads-max` — stabilisce il numero massimo di thread che il kernel può utilizzare, con un valore di default pari a 2048.
- `version` — visualizza la data e l'ora dell'ultima compilazione del kernel. Il primo campo di questo file, che può essere per esempio #3, si riferisce al numero di volte in cui il kernel è stato costruito dalla base sorgente.

La directory `random` memorizza numerosi valori relativi alla generazione di numeri casuali per il kernel.

2.3.9.4. `/proc/sys/net/`

La directory contiene un assortimento di directory inerenti a vari aspetti di networking. Molte configurazioni, durante la compilazione del kernel, rendono disponibili diverse directory, come `appletalk`, `ethernet`, `ipv4`, `ipx` e `ipv6`. All'interno di queste directory si possono sistemare i diversi valori di networking in funzione di quella configurazione su un sistema in funzione.

Data l'ampia gamma di possibili opzioni di networking disponibile con Linux, si renderebbe necessaria una quantità di spazio troppo grande per presentarle tutte. Di conseguenza verranno presentate solo le directory `/proc/sys/net` più comuni.

La directory `core` contiene una serie di impostazioni che controllano l'interazione tra il kernel e i livelli di networking. I file più importanti sono i seguenti:

- `message_burst` — i decimi di secondo necessari per scrivere un messaggio di avvertimento. Il file viene utilizzato per impedire attacchi *Dos* (*Denial of Service*). Il valore predefinito è 50.
- `message_cost` — usato anch'esso per impedire attacchi DoS, ponendo un costo su ogni messaggio di avvertimento. Più alto è il valore di questo file (di default è impostato su 5), maggiore è la probabilità che il messaggio venga ignorato.

Un hacker potrebbe infatti bombardare il sistema di richieste che generano errori e riempiono le partizioni del disco di log o richiedono a tutte le risorse del sistema di gestire logging di errore. Le impostazioni in `message_burst` e `message_cost` possono essere modificate in funzione del fattore di rischio accettabile per il sistema contro la necessità di un logging di vasta portata.

- `netdev_max_backlog` — stabilisce il numero massimo di pacchetti che possono restare in coda quando la velocità con cui una particolare interfaccia riceve i pacchetti è superiore a quella con cui il kernel è in grado di elaborarli. Il valore predefinito per questo file è 300.
- `optmem_max` — configura la dimensione massima del buffer ausiliario consentito per ogni socket.
- `rmem_default` — stabilisce la dimensione predefinita (misurata in byte) del buffer del socket per la ricezione.
- `rmem_max` — stabilisce la dimensione massima (misurata in byte) del buffer del socket per la ricezione.
- `wmem_default` — stabilisce la dimensione predefinita (misurata in byte) del buffer del socket per la trasmissione.
- `wmem_max` — stabilisce la dimensione massima (misurata in byte) del socket per la trasmissione.

La directory `/ipv4` contiene impostazioni di rete aggiuntive. Molte di queste impostazioni, correttamente associate tra loro, sono davvero utili per impedire attacchi al sistema o utilizzare il sistema in modo che funga da router.



Attenzione

L'errata modifica di questi file può avere ripercussioni sulla connettività remota al vostro sistema.

Qui di seguito sono riportati alcuni dei file più importanti contenuti nella directory `/proc/sys/net/ipv4/`:

- `icmp_destunreach_rate`, `icmp_echoreply_rate`, `icmp_paramprob_rate` e `icmp_timeexceed_rate` — impostano la velocità massima (in centesimi di secondo) di invio dei pacchetti ICMP a degli host a diverse condizioni. Con impostazione 0 vengono rimossi tutti i ritardi ed è, dunque, sconsigliabile.
- `icmp_echo_ignore_all` e `icmp_echo_ignore_broadcasts` — consentono, rispettivamente, al kernel di ignorare i pacchetti ICMP ECHO provenienti da qualsiasi host o solo quelli provenienti da indirizzi broadcast e multicast. Se è impostato il valore 0 il kernel risponderà positivamente, mentre con valore 1 i pacchetti saranno ignorati.
- `ip_default_ttl` — imposta il *TTL* (*Time To Live*) predefinito, ossia il valore che limita il numero di salti che un pacchetto può compiere prima di arrivare a destinazione. Aumentare questo valore può portare a una riduzione nelle prestazioni del sistema.
- `ip_forward` — consente alle interfacce del sistema di inoltrarsi pacchetti a vicenda. Il valore predefinito di questo file è 0 e ciò significa che la funzione di inoltrare non è abilitata. Per attivarla occorre impostare il file su 1.
- `ip_local_port_range` — specifica il range di porte che TCP o UDP devono utilizzare quando è richiesta una porta locale. Il primo numero rappresenta la porta più bassa da utilizzare, mentre il secondo numero indica quella più alta. Per i sistemi su cui si prevede di dover utilizzare un numero più elevato di porte rispetto a quello predefinito (da 1024 a 4999) occorre impostare, in questo file, il range 32768 a 61000.
- `tcp_syn_retries` — permette di impostare un limite al numero di tentativi da parte del sistema di trasmettere un pacchetto SYN quando si sta cercando di effettuare una connessione.
- `tcp_retries1` — imposta il numero di tentativi di trasmissione consentiti quando si sta cercando di rispondere a una connessione in ingresso. Il numero predefinito è 3.
- `tcp_retries2` — imposta il numero di tentativi consentiti di trasmissione dei pacchetti TCP. Il valore predefinito è 15.

Se desiderate un elenco completo dei file e delle opzioni disponibili nella directory `/proc/sys/net/ipv4/`, consultate il file `/usr/src/linux-2.4/Documentation/networking/ip-sysctl.txt`.

All'interno della directory `/proc/sys/net/ipv4/` si trovano svariate altre directory relative ad argomenti specifici. La directory `conf` consente di configurare ciascuna delle interfacce del sistema in modo diverso. Si trovano, tra l'altro, impostazioni predefinite da utilizzare per i dispositivi non configurati (sottodirectory `default`) e impostazioni che annullano qualsiasi configurazione speciale (sottodirectory `all`).

Per controllare le connessioni tra `neighbor` diretti, cioè con qualsiasi altro sistema direttamente connesso al vostro, potete effettuare configurazioni mirate per ciascuna interfaccia grazie alla directory `neigh`. Questo vi permette di interagire in modo diverso con i sistemi che ritenete più fidati in base alla loro prossimità o al rapporto che hanno con il vostro sistema e, allo stesso tempo, vi mette in condizione di stabilire regole ferree per i sistemi più distanti.

Il routing IPv4 ha la propria directory (`route`). A differenza di `conf` e `neigh`, la directory `route` contiene delle specifiche di routing applicabili a tutte le interfacce presenti sul sistema. Molte di queste impostazioni, tra cui `max_size`, `max_delay` e `min_delay`, riguardano il controllo delle dimensioni della cache di routing. Per vuotare la cache di routing è sufficiente scrivere qualsiasi valore sul file `flush`.

Per reperire ulteriori informazioni riguardo a queste directory e ai possibili valori per i relativi file di configurazione consultate il file `/usr/src/linux-2.4/Documentation/filesystems/proc.txt`.

2.3.9.5. `/proc/sys/vm/`

Questa directory facilita la configurazione del sottosistema VM (virtual memory) del kernel di Linux. Il kernel fa un uso estensivo ed efficiente della memoria virtuale, comunemente nota come spazio di swap.

I seguenti file si trovano normalmente nella directory `/proc/sys/vm`:

- `bdflush` — imposta i valori relativi al demone `bdflush`.
- `buffermem` — consente di controllare la percentuale di memoria totale del sistema da utilizzare per la memoria buffer. L'output di questo file ha, di norma, il seguente aspetto:

```
2      10      60
```

Il primo e l'ultimo valore definiscono rispettivamente la percentuale minima e massima da utilizzare come memoria buffer. Il valore centrale stabilisce la percentuale di memoria di sistema assegnata alla memoria buffer dove il sottosistema di gestione della memoria inizierà a vuotare la cache di buffer, più che altri tipi di memoria, per compensare una generale carenza di memoria libera.

- `kswapd` — imposta i valori relativi al demone `kswapd` del kernel per le operazioni di swap out. Questo file presenta tre valori

```
512 32 8
```

Il primo valore stabilisce il numero massimo di pagine che `kswapd` cercherà di liberare in un singolo tentativo. Più alto è questo numero, maggiore è la determinazione con cui il kernel si sposta verso le pagine libere. Il secondo valore stabilisce il numero minimo di tentativi che `kswapd` può compiere per liberare una pagina. Il terzo valore stabilisce quante pagine alla volta `kswapd` cerca di scrivere. Impostando in modo adeguato quest'ultimo valore è possibile migliorare le prestazioni di un sistema che usa molto spazio di swap istruendo il kernel affinché scriva pagine con blocchi di dati piuttosto grandi, riducendo al minimo il numero di ricerche su disco.

- `max_map_count` — configura il numero massimo di zone di memoria map di cui può disporre un processo. Di norma, il valore predefinito `65536` risulta appropriato.

- `overcommit_memory` — quando impostato al valore predefinito 0, il kernel calcola la quantità di memoria disponibile e interrompe richieste chiaramente non valide. Sfortunatamente, dato che la memoria è allocata mediante un algoritmo euristico invece di uno preciso, il sistema può risultare sovraccarico.

Se `overcommit_memory` è impostato a 1, viene aumentato il potenziale di sovraccarico del sistema, così come le prestazioni per attività che richiedono molta memoria, come quelle utilizzate da alcune applicazioni scientifiche.

Per coloro che necessitano meno rischi di sovraccarico di memoria sono state aggiunte le due opzioni riportate di seguito. L'impostazione di `overcommit_memory` a 2 porta a un'interruzione se una richiesta di memoria aggiunge oltre la metà di RAM fisica, oltre allo swap. L'impostazione a 3 porta a un'interruzione se la richiesta di memoria aggiunge un valore superiore a quello che lo swap da solo può contenere.

- `pagecache` — controlla la quantità di memoria utilizzata dalla cache della pagina. I valori contenuti in `pagecache` sono percentuali e funzionano in modo analogo a `buffermem` per stabilire quantità massime e minime di memoria di pagina della cache disponibile.
- `page-cluster` — stabilisce il numero di pagine lette in una volta. Il valore predefinito (4), che di fatto si riferisce a 16 pagine, risulta adeguato per la maggior parte dei sistemi.
- `pagetable_cache` — controlla il numero di page table che vengono memorizzate in cache per processore. Il primo e il secondo valore si riferiscono rispettivamente al numero minimo e al numero massimo di page table da memorizzare.

Per maggiori informazioni su questi file consultate il file `/usr/src/linux-2.4/Documentation/sysctl/vm.txt`.

2.3.10. `/proc/sysvipc/`

La directory contiene informazioni sulle risorse di System V IPC. I file contenuti in questa directory riguardano le chiamate di System V IPC per i messaggi (`msg`), i semafori (`sem`) e la memoria condivisa (`shm`).

2.3.11. `/proc/tty/`

La directory contiene informazioni circa i dispositivi tty disponibili e attualmente in uso sul sistema. I terminali a carattere, che in origine si chiamavano *dispositivi teletype* (telescriventi), vengono ora chiamati *dispositivi tty*.

In Linux esistono tre diversi tipi di dispositivi tty. I *dispositivi seriali* vengono utilizzati con le connessioni seriali (per esempio tramite un modem o un cavo seriale). I *terminali virtuali* creano la comune connessione di console (per esempio le console virtuali disponibili quando viene digitata la combinazione di tasti `[Alt]-[<F-key>]` nella console di sistema). Gli *pseudo terminali* creano una comunicazione bidirezionale (two-way) usata da alcune applicazioni di alto livello (per esempio **X11**).

Il file `drivers` è un elenco dei dispositivi tty attualmente in uso:

```
serial          /dev/cua        5  64-127 serial:callout
serial          /dev/ttyS       4  64-127 serial
pty_slave      /dev/pts       136 0-255 pty:slave
pty_master     /dev/ptm       128 0-255 pty:master
pty_slave      /dev/ttyp      3   0-255 pty:slave
pty_master     /dev/pty       2   0-255 pty:master
/dev/vc/0      /dev/vc/0      4   0 system:vtmaster
/dev/ptmx      /dev/ptmx      5   2 system
/dev/console   /dev/console   5   1 system:console
```

```

/dev/tty          /dev/tty          5          0 system:/dev/tty
unknown          /dev/vc/%d        4          1-63 console

```

Il file `/proc/tty/driver/serial` elenca le statistiche di utilizzo e lo stato di ciascuna delle linee tty seriali.

Per poter utilizzare i dispositivi tty in modo analogo ai dispositivi di rete, il kernel di Linux applica al dispositivo una *disciplina di linea*. Questo consente al driver di inserire uno specifico tipo di in-testazione in ogni blocco di dati trasmesso tramite il dispositivo, permettendo così, all'estremo della connessione, di vedere quel blocco di dati come singola parte di un flusso di blocchi di dati. SLIP e PPP sono comuni discipline di linea utilizzate per connettere tra loro vari sistemi mediante collegamento seriale.

Le discipline di linea registrate sono memorizzate nel file `ldiscs` e informazioni dettagliate sono disponibili nella directory `ldisc`.

2.4. Utilizzo di `sysctl`

Il `sysctl` è utilizzato per visualizzare, impostare e automatizzare parametri speciali del kernel nella directory `/proc/sys/`.

Se desiderate una rapida panoramica di tutte le impostazioni configurabili nella directory `/proc/sys/`, digitate il comando `sysctl-a` come root per ottenere un elenco molto dettagliato. Una piccola parte di tale elenco ha all'incirca il seguente aspetto:

```

net.ipv4.route.min_delay = 2
kernel.sysrq = 0
kernel.sem = 250      32000      32      128

```

In questo modo si ottengono le stesse informazioni fornite dalla visualizzazione di un singolo file alla volta. L'unica differenza è data dalla posizione del file. `/proc/sys/net/ipv4/route/min_delay` è segnalato da `net.ipv4.route.min_delay`: gli slash della directory sono sostituiti da punti e la parte `proc.sys` è presunta.

Il comando `sysctl` può essere utilizzato al posto di `echo` per assegnare dei valori a file scrivibili nella directory `/proc/sys/`. Per esempio, invece di utilizzare questo comando:

```
echo 1 > /proc/sys/kernel/sysrq
```

Potete usare il comando `sysctl`:

```
sysctl -w kernel.sysrq="1"
kernel.sysrq = 1

```

L'impostazione rapida di singoli valori come questo all'interno di `/proc/sys` risulta utile quando si sta effettuando una prova, ma dà risultati altrettanto positivi su un sistema di produzione, poiché tutte le impostazioni speciali presenti in `/proc/sys` vanno perdute al riavvio della macchina. Per conservare le impostazioni da assegnare al kernel in modo permanente occorre aggiungerle al file `/etc/sysctl.conf`.

A ogni riavvio del sistema, `init` esegue lo script `/etc/rc.d/rc.sysinit`, contenente un comando per l'esecuzione di `sysctl`, utilizzando `/etc/sysctl.conf` per stabilire i valori passati al kernel. I valori aggiunti a `/etc/sysctl.conf` avranno effetto subito dopo l'avvio del sistema.

2.5. Risorse aggiuntive

Di seguito è riportato un elenco di fonti aggiuntive relative a `/proc/`.

2.5.1. Documentazione installata

La maggior parte delle informazioni più utili e complete riguardo a `/proc/` è disponibile sul vostro sistema.

- `/usr/src/linux-2.4/Documentation/filesystems/proc.txt` — contiene alcune limitate informazioni di varia natura relative a tutti gli aspetti della directory `/proc/`.
- `/usr/src/linux-2.4/Documentation/sysrq.txt` — una panoramica sulle opzioni del tasto SysRq.
- `/usr/src/linux-2.4/Documentation/sysctl` — una directory contenente numerosi suggerimenti riguardo a `sysctl`, per esempio la modifica dei valori relativi al kernel (`kernel.txt`), l'accesso ai filesystem (`fs.txt`) e l'utilizzo della memoria virtuale (`vm.txt`).
- `/usr/src/linux-2.4/Documentation/networking/ip-sysctl.txt` — presenta varie opzioni di networking IP.
- `/usr/src/linux-2.4` — probabilmente, le informazioni più autorevoli che possiate ottenere riguardo a `/proc/` si trovano nel codice sorgente del kernel. Accertatevi che l'RPM `kernel-source` sia installato sul sistema e cercate il sorgente nella directory `/usr/src/linux-2.4`.

2.5.2. Siti Web utili

- <http://www.linuxhq.com> — in questo sito si trova un database completo contenente i sorgenti, le patch e la documentazione relativa a diverse versioni del kernel di Linux.

Processo di avvio, init e spegnimento

Uno degli aspetti più importanti e potenti di Red Hat Linux è il metodo utilizzato per avviare e arrestare il sistema operativo. Al momento dell'avvio Red Hat Linux carica i programmi in ordine specifico e configurabile e, successivamente, potete liberamente cambiare i file di configurazione che controllano il processo di avvio oltre ai file di configurazione per i programmi eseguiti in fase di avvio. In modo simile, l'arresto del sistema interrompe i processi in modo organizzato e configurabile, anche se la personalizzazione di questo processo è raramente necessaria.

La comprensione del funzionamento dei processi di avvio e di arresto non solo vi consente di personalizzare in modo semplice Red Hat Linux, ma vi aiuta a determinare la fonte esatta della maggior parte dei problemi legati all'avvio o allo spegnimento del sistema.

3.1. Il processo di avvio

Di seguito sono riportate le fasi principali del processo di avvio per un sistema x86:

1. Il sistema BIOS (Basic Input/Output System) controlla il sistema e avvia il boot loader della prima fase nel file MBR del disco fisso primario.
2. Il boot loader della prima fase viene caricato in memoria e consente di avviare il boot loader della seconda fase dalla partizione `/boot/`.
3. Il kernel viene caricato in memoria e carica tutti i moduli e monta la partizione root di sola lettura.
4. Il kernel gestisce il controllo del processo di avvio per il programma `/sbin/init`.
5. Il programma `/sbin/init` carica tutti i servizi e gli strumenti user-space e monta tutte le partizioni elencate in `/etc/fstab`.

Dato che la configurazione del processo di avvio è più comune della personalizzazione del processo di arresto, la parte restante di questo capitolo sarà dedicata a presentare in modo dettagliato il funzionamento del processo di avvio e come può essere personalizzato in base alle esigenze.

3.2. Un esame dettagliato del processo di avvio

L'inizio del processo di avvio varia in base alla piattaforma in cui viene avviato Red Hat Linux. Tuttavia, quando il kernel viene rilevato e caricato dal sistema, il processo di avvio di default è identico per tutte le architetture. Nell'esempio riportato di seguito, si tratterà di un computer con sistema x86.

3.2.1. Il BIOS

Quando un computer x86 viene avviato, il processore controlla il *BIOS* o *Basic Input/Output System* alla fine della memoria di sistema e lo avvia. Il BIOS controlla non solo la prima fase del processo di avvio, ma fornisce l'interfaccia di livello inferiore alle periferiche. Per questo motivo è scritto in una memoria permanente in sola lettura e può sempre essere utilizzato.

Altre piattaforme utilizzano programmi diversi per eseguire attività di livello inferiore in minima parte equivalenti a quelle del BIOS di un sistema x86. Per esempio i computer con processore Itanium utilizzano la *shell EFI* (*Extensible Firmware Interface*, mentre i sistemi Alpha utilizzano la *console SRM*.

Dopo il caricamento il BIOS testa il sistema, cerca e controlla le periferiche e cerca un dispositivo valido per avviare il sistema. Di solito controlla le unità floppy e i CD-ROM presenti alla ricerca di supporti avviabili e verifica il disco fisso. Nella maggior parte dei casi, la sequenza delle unità utilizzate per l'avvio è controllata da una particolare configurazione del BIOS. Spesso il primo disco fisso impostato per l'avvio è il disco C o il dispositivo IDE master del bus IDE primario. Il BIOS carica in memoria qualsiasi programma si trovi nel primo settore di questo dispositivo, denominato *MBR* o *Master Boot Record*. L'MBR ha dimensioni pari a soli 512 byte e contiene le istruzioni in codice macchina per l'avvio del computer oltre alla tabella delle partizioni. Al termine dell'operazione il BIOS passa il controllo a qualsiasi programma si trova nell'MBR.

3.2.2. Il boot loader

Questa sezione si dedica in particolare al processo di avvio per la piattaforma x86, che può variare leggermente in base all'architettura del sistema. Per ulteriori informazioni sui processi di avvio diversi da x86, consultate la Sezione 3.4.

I boot loader di Linux per la piattaforma x86 sono caratterizzati da almeno due fasi, la prima delle quali è rappresentata da una piccola porzione del codice macchina binario dell'MBR. L'unico obiettivo di questa fase è quello di rilevare il boot loader secondario e caricare la prima parte in memoria. In Red Hat Linux potete installare uno dei due boot loader *GRUB* o *LILLO*. GRUB è il boot loader di default e LILLO è disponibile per coloro che lo richiedono per la propria configurazione hardware o per chi lo preferisce. Per ulteriori informazioni sulla configurazione e sull'utilizzo di GRUB o LILLO, consultate il Capitolo 4.

Se utilizzate LILLO in Red Hat Linux, il boot loader secondario impiega informazioni dell'MBR per determinare quali opzioni del boot loader sono disponibili. Ciò significa che ogni volta che viene effettuata una modifica alla configurazione o si aggiorna il kernel manualmente, è necessario eseguire il comando `/sbin/lilo -v -v` per scrivere le informazioni appropriate nell'MBR. Per ulteriori informazioni su questa operazione, consultate la Sezione 4.8.

GRUB, d'altro canto, può leggere le partizioni ext2 e caricare semplicemente il file di configurazione `/boot/grub/grub.conf` quando viene chiamato il boot loader secondario. Per informazioni su come modificare questo file, consultate la Sezione 4.7.



Suggerimento

Se aggiornate il kernel mediante **Red Hat Update Agent**, l'MBR o il file `/boot/grub/grub.conf` verrà aggiornato automaticamente. Per ulteriori informazioni su RHN, fate riferimento all'URL riportato di seguito, <https://rhn.redhat.com>

Dopo avere memorizzato il boot loader secondario, potrete visualizzare la schermata iniziale di Red Hat Linux che mostra i diversi sistemi operativi o kernel che sono stati configurati per l'avvio. Se è installato solo Red Hat Linux e nulla è stato modificato nel file `/etc/lilo.conf` o `/boot/grub/grub.conf`, sarà disponibile una sola opzione di avvio.

Se avete configurato il boot loader per l'avvio di altri sistemi operativi, questa schermata vi consentirà di selezionarli. Utilizzare i tasti direzionali per evidenziare il sistema operativo desiderato e premere [Invio]. Se non effettuate alcuna operazione, il boot loader caricherà la selezione predefinita.



Nota Bene

Se avete installato il supporto del kernel SMP (Symmetric Multi-Processor), potrete visualizzare più opzioni al primo avvio del sistema. In LILLO potrete visualizzare `linux` e `linux-up`. In GRUB Red Hat Linux (*versione kernel*) e Red Hat Linux (*smp versione-kernel*). L'opzione `linux`

O Red Hat Linux (*smp-versione kernel*) è il kernel SMP. Se si verificassero problemi con il kernel SMP, selezionate `linux-up` o il kernel non SMP dopo il riavvio.

Se dovete modificare gli argomenti della linea di comando del kernel, consultate il Capitolo 4. Per informazioni sulla modifica del runlevel al prompt di GRUB o LILO, consultate la Sezione 3.5.

Dopo che il boot loader secondario ha determinato quale kernel avviare, rileva il kernel binario corrispondente nella directory `/boot/`, cioè il file `/boot/vmlinuz-2.4.x-xx` che corrisponde alle impostazioni del boot loader. Colloca quindi in memoria l'immagine *RAM disk iniziale* appropriata, denominata *initrd*. *initrd* è utilizzata dal kernel per caricare tutti i driver non compilati e che sono necessari per avviare il sistema. Questa operazione è particolarmente importante se disponete di unità SCSI o state utilizzando il filesystem `ext3`¹.



Avvertenza

Non rimuovete per nessun motivo la directory `/initrd/` dal filesystem o il sistema non sarà in grado di avviarsi e verrà visualizzato un messaggio di errore relativo al kernel.

Dopo avere caricato in memoria il kernel e l'immagine *initrd*, il boot loader gestisce il controllo del processo di avvio per il kernel.

3.2.3. Il kernel

Quando il kernel viene caricato, inizializza e configura immediatamente la memoria del computer. Configura quindi i vari elementi hardware collegati al sistema, inclusi tutti i processori e i sottosistemi I/O, oltre a tutti i dispositivi di memorizzazione. Cerca quindi l'immagine *initrd* compressa in un percorso predeterminato della memoria, la decomprime, la monta e carica tutti i driver necessari. Successivamente inizializza i dispositivi virtuali del sistema, come LVM o il software RAID prima di smontare l'immagine disco *initrd* e liberare tutta la memoria.

Dopo l'inizializzazione di tutti i dispositivi del sistema da parte del kernel, viene creato un dispositivo root, montata la partizione root di sola lettura e liberata la memoria non utilizzata.

Il kernel risulta così caricato in memoria e operativo. Tuttavia, senza alcuna applicazione che consenta all'utente di fornire input significativo al sistema, il kernel non è molto utile.

Per configurare l'ambiente utente, il kernel esegue il comando `/sbin/init`.

3.2.4. Il programma `/sbin/init`

Il programma `init` coordina la fase restante del processo di avvio e configura l'ambiente per l'utente.

Quando `init` viene eseguito, diventa il padre di tutti i processi che si avviano automaticamente sul sistema Red Hat Linux. Innanzitutto esegue lo script `/etc/rc.d/rc.sysinit` che imposta il percorso, attiva lo swapping, controlla i filesystem e così via. In sostanza, `rc.sysinit` si occupa di tutti i processi che vanno eseguiti all'inizializzazione del sistema. La maggior parte dei sistemi utilizza un orologio e `rc.sysinit` usa il file di configurazione `/etc/sysconfig/clock` per inizializzare l'orologio. Se dovete inizializzare processi speciali per le porte seriali, `rc.sysinit` può eseguire anche `rc.serial`.

1. Per dettagli su come creare un *initrd*, consultate la sezione sulla conversione di un filesystem `ext3` nella *Official Red Hat Linux Customization Guide*.

In seguito *init* esegue lo script `/etc/inittab`, che descrive il modo in cui il sistema va configurato per ogni *runlevel* SysV². Questo file specifica, tra le altre cose, che `/etc/inittab` imposta il runlevel predefinito e che `/sbin/update` va eseguito a ogni avvio dei runlevel³.

Successivamente *init* configura la libreria delle funzioni sorgenti `/etc/rc.d/init.d/functions` per il sistema, che stabilisce come avviare o terminare un programma e come trovarne il PID.

A questo punto *init* avvia tutti i processi di background cercando nella relativa directory `rc` il runlevel specificato come predefinito in `/etc/inittab`. Le directory `rc` sono numerate per corrispondere ai runlevel che rappresentano. Per esempio `/etc/rc.d/rc5.d/` è la directory per il runlevel cinque.

Quando si esegue l'avvio dal runlevel 5, il programma *init* cerca nella directory `/etc/rc.d/rc5.d/` per determinare quali processi iniziare e interrompere. Di seguito è riportato un esempio che illustra un runlevel 5, la directory `/etc/rc.d/rc5.d/`:

```
K01pppoe -> ../init.d/pppoe
K05innd -> ../init.d/innd
K10ntpd -> ../init.d/ntpd
K15httpd -> ../init.d/httpd
K15mysqld -> ../init.d/mysqld
K15pvmd -> ../init.d/pvmd
K16rarpd -> ../init.d/rarpd
K20bootparamd -> ../init.d/bootparamd
K20nfs -> ../init.d/nfs
K20rstatd -> ../init.d/rstatd
K20rusersd -> ../init.d/rusersd
K20rwalld -> ../init.d/rwalld
K20rwhod -> ../init.d/rwhod
K25squid -> ../init.d/squid
K28amd -> ../init.d/amd
K30mcsvr -> ../init.d/mcserve
K34yppasswdd -> ../init.d/yppasswdd
K35dhcpcd -> ../init.d/dhcpcd
K35smb -> ../init.d/smb
K35vncserver -> ../init.d/vncserver
K45arpwatch -> ../init.d/arpwatch
K45named -> ../init.d/named
K50snmpd -> ../init.d/snmpd
K54pxe -> ../init.d/pxe
K55routed -> ../init.d/routed
K60mars-nwe -> ../init.d/mars-nwe
K61ldap -> ../init.d/ldap
K65kadmin -> ../init.d/kadmin
K65kprop -> ../init.d/kprop
K65krb524 -> ../init.d/krb524
K65krb5kdc -> ../init.d/krb5kdc
K75gated -> ../init.d/gated
K80nscd -> ../init.d/nscd
K84ypserv -> ../init.d/ypserv
K90ups -> ../init.d/ups
K96irda -> ../init.d/irda
S05kudzu -> ../init.d/kudzu
S06reconfig -> ../init.d/reconfig
S08ipchains -> ../init.d/ipchains
S10network -> ../init.d/network
```

2. Per maggiori informazioni su SysV *init*, consultate la Sezione 3.5. Per maggiori informazioni sulla configurazione dei runlevel, consultate la Sezione 3.6)

3. Il comando `update` viene utilizzato per ripulire i buffer difettosi su disco.

```

S12syslog -> ../init.d/syslog
S13portmap -> ../init.d/portmap
S14nfslock -> ../init.d/nfslock
S18autofs -> ../init.d/autofs
S20random -> ../init.d/random
S25netfs -> ../init.d/netfs
S26apmd -> ../init.d/apmd
S35identd -> ../init.d/identd
S40atd -> ../init.d/atd
S45pcmcia -> ../init.d/pcmcia
S55sshd -> ../init.d/sshd
S56rawdevices -> ../init.d/rawdevices
S56xinetd -> ../init.d/xinetd
S60lpd -> ../init.d/lpd
S75keytable -> ../init.d/keytable
S80isdn -> ../init.d/isdn
S80sendmail -> ../init.d/sendmail
S85gpm -> ../init.d/gpm
S90canna -> ../init.d/canna
S90crond -> ../init.d/crond
S90FreeWnn -> ../init.d/FreeWnn
S90xfs -> ../init.d/xfs
S95anacron -> ../init.d/anacron
S95firstboot -> ../init.d/firstboot
S97rhnscd -> ../init.d/rhnscd
S99local -> ../rc.local
S99mdmonitor -> ../init.d/mdmonitor

```

Nessuno degli script che avvia e interrompe realmente i servizi si trova nella directory `/etc/rc.d/rc5.d/`. Tutti i file in `/etc/rc.d/rc5.d/` sono *link simbolici* diretti a script che si trovano nella directory `/etc/rc.d/init.d/`. I link simbolici sono utilizzati in ciascuna delle directory `rc` per fare in modo che i runlevel possano essere riconfigurati creando, modificando ed eliminando i link simbolici senza influire sugli script a cui fanno riferimento.

Il nome di ciascun link simbolico inizia con `K` o `S`. I link `K` sono processi che vengono terminati, mentre quelli che iniziano con `S` vengono avviati.

Il comando `init` interrompe innanzi tutto i link simbolici `K` della directory eseguendo il comando `/etc/rc.d/init.d/<comando> stop`, in cui `<comando>` è il processo da terminare. Avvia quindi tutti i link simbolici `S` eseguendo il comando `/etc/rc.d/init.d/<comando> start`.



Suggerimento

Al termine dell'avvio del sistema, potete accedere come `root` ed eseguire gli stessi script per avviare e interrompere i servizi. Per esempio `/etc/rc.d/init.d/httpd stop` interromperà il server Web Apache.

Ciascuno dei link simbolici è numerato per stabilire l'ordine di avvio. Potete modificare l'ordine in cui i servizi vengono avviati o interrotti cambiando questo numero. I link simbolici con lo stesso numero sono avviati in base a un ordine alfabetico.

Nella directory di esempio `/etc/rc.d/rc5.d/` precedente `init` interrompe `pppoe`, `innd`, `ntpd`, `httpd`, `mysqld`, `pvmd`, `rarpd`, `bootparamd`, `nfs`, `rstatd`, `rusersd`, `rwall`, `rwhod`, `squid`, `amd`, `mcserv`, `yppasswdd`, `dhcpd`, `smb`, `vncserver`, `arpwatch`, `named`, `snmpd`, `pxe`, `routed`, `mars-nwe`, `ldap`, `kadmin`, `kprop`, `krb524`, `krb5kdc`, `gated`, `nscd`, `ypserv`, `ups` e `irda`.

Dopo avere terminato tutti i processi, `init` cerca nella stessa directory e trova script di avvio per `ku-dzu`, `reconfig`, `ipchains`, `portmap`, `nfslock`, `autofs`, `random`, `netfs`, `apmd`, `identd`, `atd`, `pcmcia`, `sshd`, `rawdevices`, `xinetd`, `lpd`, `keytable`, `isdn`, `sendmail`, `gpm`, `canna`, `crond`, `FreeWnn`, `xf`, `anacron`, `firstboot`, `rhn` e `mdmonitor`.

L'ultima azione di `init` è l'avvio di tutti gli script di `/etc/rc.d/rc.local` (per ulteriori informazioni sulla personalizzazione del file `rc.local`, consultate la Sezione 3.3). A questo punto il sistema funziona al runlevel 5.

Dopo che `init` ha percorso la directory appropriata `rc` alla ricerca del runlevel, lo script `/etc/inittab` crea un processo `getty` per ciascuna console virtuale (prompt di login) di ogni runlevel. I runlevel da 2 a 5 dispongono di sei console, mentre il runlevel 1, in modalità a utente singolo, dispone di un'unica console e i runlevel 0 e 6 non ne ottengono nessuna. In sostanza, `getty` apre delle linee ai dispositivi `tty`⁴, ne imposta la modalità, visualizza il prompt di login, riceve il nome dell'utente e inizializza il processo di login per quell'utente.

Nel runlevel 5 `/etc/inittab` esegue uno script denominato `/etc/X11/prefdm`. Lo script `prefdm` esegue il display manager X preferito, `gdm` se state utilizzando GNOME o `kdm` se state eseguendo KDE, in base al contenuto della directory `/etc/sysconfig/desktop/`.

Infine il comando `init` esegue lo script `/etc/rc.d/rc.local`.

A questo punto dovrebbe comparire il prompt di login.

3.3. Esecuzione dei programmi all'avvio

Lo script `/etc/rc.d/rc.local` viene eseguito dal comando `init` in fase di avvio, al termine dell'inizializzazione e ogni volta che modificate i runlevel. Potete aggiungere ulteriori comandi di inizializzazione. Per esempio potete avviare ulteriori demoni o inizializzare una stampante.

Se è necessaria la configurazione di una porta seriale, potete creare e modificare `/etc/rc.serial` per eseguire l'operazione automaticamente in fase di avvio. Questo script può eseguire numerosi comandi `setserial` per configurare in modo specifico le porte seriali del sistema. Per ulteriori informazioni, consultate la pagina `man setserial`.

3.4. Differenze nel processo di avvio di altre architetture

Dopo il caricamento del kernel di Red Hat Linux e la gestione del processo relativo con `init`, gli stessi eventi si verificano in ciascuna architettura nello stesso modo. L'unica differenza tra le due architetture sta nell'applicazione utilizzata per trovare e caricare il kernel.

Per esempio, l'architettura Alpha utilizza il boot loader `aboot`, mentre l'architettura Itanium impiega il boot loader `ELILO`.

Per informazioni dettagliate sui diversi metodi di avvio, consultate la Official Red Hat Linux Installation Guide specifica di queste piattaforme.

3.5. SysV Init

`SysV init` è il processo standard utilizzato da Red Hat Linux per controllare quale software viene avviato per un runlevel particolare dal comando `init`. È stato scelto perché è più semplice da utilizzare e più flessibile dello stile BSD tradizionale.

4. Per maggiori informazioni sui dispositivi `tty`, consultate la Sezione 2.3.11.

I file di configurazione di SysV init si trovano in `/etc/rc.d`. In questa directory troverete gli script `rc`, `rc.local`, `rc.sysinit` e le seguenti directory:

```
init.d
rc0.d
rc1.d
rc2.d
rc3.d
rc4.d
rc5.d
rc6.d
```

La directory `init.d` contiene gli script utilizzati dal comando `init` per il controllo dei servizi. Ciascuna delle directory numerate rappresenta i sei runlevel predefiniti configurati per default in Red Hat Linux. Per ulteriori informazioni sui runlevel, consultate la Sezione 3.6.

Il runlevel di default è definito in `/etc/inittab`. Per rilevare il runlevel per il vostro sistema, cercate la linea simile a quella riportata di seguito nella parte superiore di `/etc/inittab`:

```
id:3:initdefault:
```

In questo esempio il runlevel di default è il 3, ossia il numero dopo la prima colonna. Se desiderate modificarlo, potete digitare `/etc/inittab` come root.



Avvertenza

Prestate molta attenzione nel modificare il file `/etc/inittab`, perché una digitazione errata potrebbe impedire al sistema di avviarsi se non tramite un dischetto di avvio o accedendo alla modalità `rescue`.

Per ulteriori informazioni sulla modalità `rescue`, consultate il capitolo *Rescue Mode* nella *Official Red Hat Linux Customization Guide*.

3.6. Runlevel init

Il concetto di attivare servizi diversi in runlevel differenti si basa sul fatto che possono essere utilizzati sistemi diversi in modi differenti. Alcuni servizi non possono essere utilizzati finché il sistema è in uno stato o *modalità*, particolare, come nel caso in cui sia pronto per più utenti o sia collegato alla rete.

Talvolta potete utilizzare il sistema in una modalità inferiore, come la risoluzione dei problemi di corruzione del disco in runlevel 1, quando nessun altro utente può trovarsi nel sistema oppure lasciare un server in runlevel 3 senza una sessione X in esecuzione. In questi casi l'esecuzione dei servizi che dipendono da una modalità di sistema superiore non ha senso perché non funzionerebbero comunque in modo corretto. L'assegnazione di ogni servizio per l'avvio al raggiungimento del runlevel specifico consente un processo di avvio regolare e può modificare rapidamente la modalità del computer senza preoccuparsi di quali servizi dovranno essere avviati o interrotti manualmente.

In genere Red Hat Linux funziona nel runlevel 3 o 5, entrambe modalità multiutente complete. In Red Hat Linux sono definiti i runlevel riportati di seguito:

- 0 — arresto
- 1 — modalità a utente singolo
- 2 — non utilizzato (definito dall'utente)
- 3 — modalità multiutente completa

- 4 — non utilizzato (definito dall'utente)
- 5 — modalità multiutente completa (con schermata di login basata su X)
- 6 — riavvio

Il runlevel predefinito, utile perché un sistema venga avviato e interrotto, è configurato in `/etc/inittab`. Per ulteriori informazioni su `/etc/inittab`, consultate la Sezione 3.5.

Potete liberamente configurare i runlevel 2 e 4 come meglio credete. Molti utenti li configurano nel modo più opportuno lasciando liberi i runlevel 3 e 5 standard. In questo modo possono uscire ed entrare dalla configurazione personalizzata senza influire sulla serie di caratteristiche comuni dei runlevel standard.

Se il computer è in uno stato per cui non potrà essere avviato a causa di un file `/etc/inittab` corrotto o non vi consentirà l'accesso perché il vostro file `/etc/passwd` è danneggiato o semplicemente avete dimenticato la password, potete eseguire l'avvio in modalità a utente singolo.

Se utilizzate LILO, potete attivare la modalità a utente singolo digitando **linux single** al prompt `boot:` di LILO.

Se utilizzate GRUB come boot loader, potete attivare la modalità a utente singolo seguendo le indicazioni riportate di seguito.

- Nella schermata grafica di GRUB selezionate l'etichetta di avvio **Red Hat Linux** e premete [e] per modificarla.
- Utilizzate il tasto freccia giù per passare alla linea del kernel e premete [e] per modificarla.
- Al prompt digitate **single** e premete [Invio].
- Tornerete alla schermata di GRUB con le informazioni sul kernel. Premete il tasto [b] per avviare il sistema in modalità a utente singolo.

Verrà avviato un sistema molto semplice e sarà disponibile una shell di comando con cui risolvere eventuali problemi.

Se questa operazione non funziona, sarà necessario digitare **linux init=/bin/bash** al prompt `boot:` di LILO. In questo modo giungerete a un prompt della shell. Nessun filesystem diverso da quello root è montato e tale filesystem è in modalità di sola lettura. Per montarlo in modalità di lettura e scrittura e consentire, per esempio la modifica di un file `/etc/inittab` interrotto, eseguite quanto segue:

```
mount -n /proc
mount -o rw,remount /
```

3.6.1. Utility di initscript

L'utility `/sbin/chkconfig` fornisce un semplice strumento a linea di comando per la manutenzione della gerarchia delle directory `/etc/rc.d/init.d`. Solleva gli amministratori di sistema dall'incombenza di manipolare i numerosi link simbolici nelle directory di `/etc/rc.d`.

Inoltre `/sbin/ntsysv` fornisce un'interfaccia testuale che può risultare più semplice da utilizzare rispetto all'interfaccia a linea di comando di `chkconfig`.

Se preferite un'interfaccia grafica, utilizzate il programma **Services Configuration Tool**, che può essere chiamato mediante il comando `redhat-config-services`.

Tutte queste utility devono essere eseguite come root.

Per ulteriori informazioni su questi strumenti, consultate il capitolo *Controllo dell'accesso ai servizi* nella *Official Red Hat Linux Customization Guide*.

Verranno ora esaminate informazioni sui file all'interno della directory `/etc/sysconfig/` che definiscono i parametri utilizzati da alcuni servizi al momento dell'avvio.

3.7. La directory `/etc/sysconfig/`

Le seguenti informazioni delineano alcuni dei numerosi file contenuti in `/etc/sysconfig`, spiegandone funzione e contenuto. Ovviamente queste informazioni non sono complete, poiché molti dei file hanno numerose opzioni utilizzate solo in casi specifici e rari.

Il file `/usr/share/doc/initscripts-<numero-versione>/sysconfig.txt` contiene un elenco più esauriente dei file disponibili nella directory `/etc/sysconfig` e le opzioni di configurazione disponibili.

3.7.1. File contenuti in `/etc/sysconfig`

Nella directory `/etc/sysconfig` si trovano di solito i file seguenti:

- `amd`
- `apmd`
- `arpwatch`
- `authconfig`
- `cipe`
- `clock`
- `desktop`
- `dhcpcd`
- `firstboot`
- `gpm`
- `harddisks`
- `hwconf`
- `i18n`
- `identd`
- `init`
- `ipchains`
- `iptables`
- `irda`
- `keyboard`
- `kudzu`
- `mouse`
- `named`
- `netdump`
- `network`
- `ntpd`

- pcmcia
- radvd
- rawdevices
- redhat-config-users
- redhat-logviewer
- samba
- sendmail
- soundcard
- squid
- tux
- ups
- vncservers
- xinetd

È possibile che nel vostro sistema ne manchi qualcuno, se non è installato il programma corrispondente.

Questi file verranno ora esaminati in dettaglio:

3.7.1.1. `/etc/sysconfig/amd`

Il file `/etc/sysconfig/amd` contiene diversi parametri utilizzati da `amd` che consentono di montare o smontare automaticamente i filesystem.

3.7.1.2. `/etc/sysconfig/apmd`

Il file `/etc/sysconfig/apmd` è utilizzato da `apmd` per sapere quali processi avviare/terminare/modificare in caso di sospensione o ripristino. Questo file è configurato per attivare o disattivare `apmd` durante l'avvio, a seconda del fatto che il vostro hardware supporti l'*Advanced Power Management (APM)* oppure no. `apm` è un demone con funzioni di controllo che utilizza un codice APM nel kernel di Linux. Se usate Red Hat Linux su un portatile, `apmd` vi segnala anche lo stato della batteria.

3.7.1.3. `/etc/sysconfig/arpwatch`

Il file `/etc/sysconfig/arpwatch` serve per inviare argomenti al demone `arpwatch` durante l'avvio. Il demone contiene una tabella di indirizzi Ethernet MAC e i corrispondenti indirizzi IP. Per maggiori informazioni sui parametri utilizzabili in questo file, digitate `man arpwatch`. Per default, questo file imposta come proprietario del processo `arpwatch` l'utente `pcap`.

3.7.1.4. `/etc/sysconfig/authconfig`

Il file `/etc/sysconfig/authconfig` stabilisce i tipi di autorizzazione da utilizzare su un host. Contiene una o più delle righe seguenti:

- `USEMD5=<valore>`, dove `<valore>` può essere sostituito con uno dei seguenti valori:
 - `yes` — se volete usare MD5 per l'autenticazione.
 - `no` — se non volete usare MD5 per l'autenticazione.

- USEKERBEROS=<valore>, dove <valore> può essere sostituito con uno dei seguenti valori:
 - yes — se volete utilizzare Kerberos per l'autenticazione.
 - no — se non volete utilizzare Kerberos per l'autenticazione.
- USELDAPAUTH=<valore>, dove <valore> può essere sostituito con uno dei seguenti valori:
 - yes — se volete utilizzare LDAP per l'autenticazione.
 - no — se non volete utilizzare LDAP per l'autenticazione.

3.7.1.5. /etc/sysconfig/clock

Il file `/etc/sysconfig/clock` controlla l'interpretazione dei valori letti dall'orologio del sistema. Nelle prime versioni di Red Hat Linux venivano usati i seguenti valori (non utilizzateli in questa versione):

- CLOCKMODE=<valore>, dove <valore> può essere sostituito con uno dei seguenti valori:
 - GMT — indica che l'orologio è impostato secondo l'ora del meridiano di Greenwich.
 - ARC — indica che è attivo il time offset di 42 anni della console ARC (solo su sistema Alpha).

Attualmente i valori corretti sono:

- UTC=<valore>, dove <valore> può essere sostituito con uno dei seguenti valori:
 - true — indica che l'orologio è impostato secondo l'ora del meridiano di Greenwich. Qualsiasi altro valore indica che è configurato con l'ora locale.
- ARC=<valore>, dove <valore> può essere sostituito con uno dei seguenti valori:
 - true — indica che il time offset di 42 anni è attivo. Qualsiasi altro valore indica che viene utilizzato il metodo normale per la gestione del tempo di UNIX (solo su sistemi Alpha).
- ZONE=<nome del file> — indica il file del fuso orario sotto `/usr/share/zoneinfo` dove `/etc/localtime` ne rappresenta una copia, per esempio:
ZONE="America/New York"

3.7.1.6. /etc/sysconfig/desktop

Il file `/etc/sysconfig/desktop` specifica quale manager del desktop eseguire, per esempio:

```
DESKTOP="GNOME"
```

3.7.1.7. /etc/sysconfig/dhcpd

Il file `/etc/sysconfig/dhcpd` serve per inviare argomenti al demone `dhcpd` durante l'avvio. Il demone implementa i protocolli DHCP (Dynamic Host Configuration Protocol) e BOOTP (il protocollo Internet Bootstrap). DHCP e BOOTP assegnano i nomi host alle macchine presenti in rete. Per maggiori informazioni su quali parametri potete utilizzare in questo file, digitate `man dhcpd`.

3.7.1.8. /etc/sysconfig/firstboot

A partire dalla versione Red Hat Linux 8.0, al primo avvio del sistema, il programma `/sbin/init` chiama lo script `etc/rc.d/init.d/firstboot`. In questo modo all'utente può installare ulteriori applicazioni e documentazioni prima del completamento del processo di avvio.

Il file `/etc/sysconfig/firstboot` indica semplicemente al comando `firstboot` di non avviarsi. Se desiderate che `firstboot` venga eseguito, al successivo avvio del sistema, rimuovete semplicemente `/etc/sysconfig/firstboot` ed eseguite `chkconfig --level 5 firstboot on`.

3.7.1.9. /etc/sysconfig/gpm

Il file `/etc/sysconfig/gpm` serve per inviare argomenti al demone `gpm` durante l'avvio. Il demone è il mouse server che permette l'accelerazione del mouse e la pressione del tasto centrale. Per maggiori informazioni su quali parametri potete utilizzare in questo file, digitate `man gpm`. Per default, imposta il dispositivo mouse su `/dev/mouse`.

3.7.1.10. /etc/sysconfig/harddisks

Il file `/etc/sysconfig/harddisks` vi consente di configurare il disco fisso. Potete anche utilizzare il file `/etc/sysconfig/hardiskhd[a-h]` per impostare parametri per driver specifici.



Avvertenza

Non effettuate modifiche a questo file, a meno che non sia strettamente necessario. Se modificate i valori di default memorizzati nel file, potreste danneggiare tutti i dati presenti sul disco fisso.

Il file `/etc/sysconfig/harddisks` può contenere i campi seguenti:

- `USE_DMA=1`: impostando il valore 1 viene abilitato il DMA. Tuttavia, con alcuni chipset e combinazioni del disco fisso, il DMA può provocare il danneggiamento dei dati. *Prima di abilitare il DMA, controllate la documentazione del disco fisso.*
- `Multiple_IO=16`: se impostato su 16 abilita diversi settori per ogni interrupt di I/O. Se abilitata, questa caratteristica riduce del 30-50% le informazioni aggiuntive del sistema operativo. *Utilizzate con cautela.*
- `EIDE_32BIT=3`: abilita il supporto (E)IDE 32-bit I/O per una scheda di interfaccia.
- `LOOKAHEAD=1` abilita l'unità.
- `EXTRA_PARAMS=`: specifica dove si possono aggiungere altri parametri.

3.7.1.11. /etc/sysconfig/hwconf

Il file `/etc/sysconfig/hwconf` elenca tutti i componenti hardware rilevati da `kudzu` sul sistema e tutti i driver usati, l'ID del rivenditore e del dispositivo. Il programma `kudzu` rileva e configura componenti hardware nuovi e/o modificati. Il file `/etc/sysconfig/hwconf` non è stato ideato per essere modificato manualmente. Se lo fate, i dispositivi potrebbero risultare aggiunti o rimossi.

3.7.1.12. /etc/sysconfig/i18n

Il file `/etc/sysconfig/i18n` imposta la lingua di default, per esempio:

```
LANG="en_US"
```

3.7.1.13. /etc/sysconfig/identd

Il file `/etc/sysconfig/identd` serve per inviare argomenti al demone `identd` durante l'avvio. Il demone restituisce il nome utente dei processi con connessioni TCP/IP aperte. Alcuni servizi di rete, come i server FTP e IRC, accuseranno difficoltà e rallentamenti nelle risposte se `identd` non è in esecuzione. In generale, comunque, `identd` non è indispensabile, quindi se vi preoccupa la questione della sicurezza, non dovrete eseguirlo. Per maggiori informazioni su quali parametri potete utilizzare in questo file, digitate `man identd`. Per default, il file non contiene alcun parametro.

3.7.1.14. /etc/sysconfig/init

Il file `/etc/sysconfig/init` controlla il funzionamento del sistema durante l'avvio.

Possono essere utilizzati i valori seguenti:

- `BOOTUP=<valore>`, dove `<valore>` può essere sostituito con uno dei seguenti valori:
 - `BOOTUP=color`: richiama una schermata standard di avvio a colori, in cui il successo o il fallimento dei dispositivi e dei servizi è visualizzato con colori diversi.
 - `BOOTUP=verbose`: richiama una schermata con stile antiquato, che visualizza soprattutto informazioni piuttosto che messaggi di successo o fallimento.
 - Qualsiasi altro elemento richiama una schermata nuova, ma senza formattazione ANSI.
- `RES_COL=<valore>`, dove `<valore>` è il numero di colonne della schermata dove vengono avviate le etichette dello stato. Il numero predefinito è 60.
- `MOVE_TO_COL=<valore>`, dove `<valore>` muove il cursore alla riga `RES_COL`. Per default vengono utilizzate le sequenze ANSI visualizzate tramite `-e`.
- `SETCOLOR_SUCCESS=<valore>`, dove `<valore>` determina il colore per la visualizzazione di un'operazione riuscita. Per default vengono usate le sequenze ANSI visualizzate da `-e`. Il colore impostato è il verde.
- `SETCOLOR_FAILURE=<valore>`, dove `<valore>` determina il colore per la visualizzazione di un'operazione fallita. Per default vengono usate le sequenze ANSI visualizzate da `-e`. Il colore impostato è il rosso.
- `SETCOLOR_WARNING=<valore>`, dove `<valore>` determina il colore per la visualizzazione di un avvertimento. Per default vengono usate le sequenze ANSI visualizzate da `-e`. Il colore impostato è il giallo.

- `SETCOLOR_NORMAL=<valore>`, dove `<valore>` imposta il colore al valore "normale". Per default vengono usate le sequenze ANSI visualizzate da -e.
- `LOGLEVEL=<valore>`, dove `<valore>` indica il livello di registrazione per il kernel della console iniziale. Il livello di default è 7, mentre 8 significa "tutto" (incluso il debugging) e 1 significa "nulla" a eccezione dei kernel panic. `syslogd` si sovrappone a questo una volta avviato.
- `PROMPT=<valore>`, dove `<valore>` può essere sostituito con uno dei seguenti valori:
 - `yes` — abilita il controllo della chiave per la modalità interattiva.
 - `no` — disabilita il controllo della chiave per la modalità interattiva.

3.7.1.15. /etc/sysconfig/ipchains

Il file `/etc/sysconfig/ipchains` contiene informazioni necessarie al kernel per impostare le regole di `ipchains` riguardanti il filtraggio dei pacchetti.

Il file viene modificato digitando il comando `/sbin/service ipchains save` quando vi sono regole di `ipchains` valide. Non potete modificare manualmente questo file. Utilizzate invece il comando `/sbin/ipchains` per impostare le regole di filtraggio dei pacchetti e salvarle in questo file mediante `/sbin/service ipchains save`.

L'utilizzo di `ipchains` per impostare le regole del firewall non è consigliabile e potrebbe non essere più presente nelle versioni future Red Hat Linux. Se dovete utilizzare un firewall, impiegate `iptables` instead.

3.7.1.16. /etc/sysconfig/iptables

Come `/etc/sysconfig/ipchains`, il file `/etc/sysconfig/iptables` contiene speciali informazioni utilizzate dal kernel per servizi particolari di filtraggio di pacchetti in fase di avvio o ogni volta che il servizio viene avviato.

Non dovete modificare questo file manualmente se non siete avvezzi all'uso di metodologie per creare le regole di `iptables`. Il modo più semplice per aggiungere le regole è quello di utilizzare il comando `/usr/sbin/lokkit` o l'applicazione grafica **gnome-lokkit** per creare il firewall. Mediante questa applicazione il file verrà automaticamente modificato alla fine del processo.

Se lo desiderate, potete creare manualmente le regole mediante `/sbin/iptables` e digitando `/sbin/service iptables save` per aggiungere le regole al file `/etc/sysconfig/iptables`.

Se il file esiste, tutte le regole dei firewall salvate in questo contesto continueranno a esistere dopo il riavvio del sistema.

Per ulteriori informazioni su `iptables` consultate il Capitolo 13.

3.7.1.17. /etc/sysconfig/irda

Il file `/etc/sysconfig/irda` controlla la configurazione dei dispositivi a infrarossi all'avvio del sistema.

È possibile utilizzare i valori seguenti:

- `IRDA=<valore>`, dove `<valore>` può essere sostituito con uno dei seguenti valori:
 - `yes` — viene eseguito `irattach` che controlla periodicamente la porta di connessione per i dispositivi a infrarossi, per verificare se tali dispositivi, come un altro portatile, cercano di creare

una connessione di rete. Se desiderate che dispositivi a infrarossi funzionino sul vostro sistema, è necessario impostare questa riga su *yes*.

- *no* — non viene eseguito *irattach*. Si impedisce così la comunicazione con dispositivi a infrarossi.
- *DEVICE=<valore>*, dove *<valore>* va sostituito con il dispositivo (di solito una porta seriale) che gestisce le connessioni ai dispositivi a infrarossi.
- *DONGLE=<valore>*, dove *<valore>* specifica il tipo di adattatore utilizzato per la comunicazione con dispositivi a infrarossi. Questa impostazione esiste per le persone che utilizzano adattatori seriali piuttosto che vere porte a infrarossi. Un adattatore è un dispositivo che, collegato a una porta seriale tradizionale, permette di comunicare tramite infrarossi. Questa riga è commentata di default, perché i portatili con porte a infrarossi effettive sono molto più diffusi di quelli con adattatori aggiunti.
- *DISCOVERY=<valore>*, dove *<valore>* può essere sostituito con uno dei seguenti valori:
 - *yes* — avvia *irattach* nella modalità Discovery, ciò significa che vengono controllati altri dispositivi a infrarossi. È necessario attivare questo comando per cercare un collegamento a infrarossi (altrimenti non viene inizializzata la connessione).
 - *no* — non avvia *irattach* nella modalità Discovery.

3.7.1.18. */etc/sysconfig/keyboard*

Il file */etc/sysconfig/keyboard* controlla il funzionamento della tastiera. È possibile utilizzare i seguenti valori:

- *KEYBOARDTYPE=sun|pc*, usata solo su SPARCs. La voce *sun* indica che una tastiera Sun è collegata a */dev/kbd* e *pc* indica che una tastiera PS/2 è connessa a una porta PS/2.
- *KEYTABLE=<file>*, dove *<file>* rappresenta il nome di un file *keytable*.

Per esempio, *KEYTABLE="us"*. I file che possono essere utilizzati come *keytable* partono da */usr/lib/kbd/keymaps/i386* e da qui si suddividono in differenti layout di tastiera tutti etichettati come *<file>.kmap.gz*. Viene usato il primo file individuato in */usr/lib/kbd/keymaps/i386* che coincide con le impostazioni di *KEYTABLE*.

3.7.1.19. */etc/sysconfig/kudzu*

Il file */etc/sysconfig/kudzu* consente all'avvio un controllo sicuro del vostro hardware tramite *kudzu*. Per "controllo sicuro" si intende un controllo che disabilita la porta seriale.

- *SAFE=<valore>*, dove *<valore>* può essere sostituito con uno dei seguenti valori:
 - *yes* — *kudzu* esegue un controllo sicuro.
 - *no* — *kudzu* esegue un controllo normale.

3.7.1.20. /etc/sysconfig/mouse

Il file `/etc/sysconfig/mouse` viene utilizzato per indicare le informazioni relative al mouse disponibile. Utilizzate i valori seguenti:

- `FULLNAME=<valore>`, dove `<valore>` va sostituito con il nome del tipo di mouse utilizzato.
- `MOUSETYPE=<valore>`, dove `<valore>` indica:
 - `microsoft` — un mouse Microsoft™.
 - `mouseman` — un mouse MouseMan™.
 - `mousesystems` — un mouse Mouse Systems™.
 - `ps/2` — un mouse PS/2.
 - `msbm` — un mouse bus Microsoft™.
 - `logibm` — un mouse bus Logitech™.
 - `atibm` — un mouse bus ATI™.
 - `logitech` — un mouse Logitech™.
 - `mmseries` — un mouse MouseMan™ datato.
 - `mmhittab` — un mouse mmhittab.
- `XEMU3=<valore>`, dove `<valore>` può essere sostituito con uno dei seguenti valori:
 - `yes` — il mouse ha solo due tasti, ma il terzo tasto viene emulato.
 - `no` — il mouse ha già tre tasti.
- `XMOUSETYPE=<valore>`, dove `<valore>` indica il tipo di mouse utilizzato con X. Le opzioni sono le stesse di `MOUSETYPE`.
- `DEVICE=<valore>`, dove `<valore>` è il dispositivo mouse.

Inoltre `/dev/mouse` è un link simbolico diretto al dispositivo mouse in uso.

3.7.1.21. /etc/sysconfig/named

Il file `/etc/sysconfig/named` serve per inviare argomenti al demone `named` durante l'avvio. Il demone è un server *Domain Name System (DNS)* che implementa la distribuzione *Berkeley Internet Name Domain (BIND)* versione 9. Il server presenta una tabella i cui nomi host sono associati a indirizzi IP sulla rete.

Al momento, si possono usare solo i valori seguenti:

- `ROOTDIR="</some/where>"`, dove `</some/where>` si riferisce al percorso completo della directory di un ambiente `chroot` configurato sotto il quale `named` verrà eseguito. Tale ambiente `chroot` deve prima essere configurato. Digitate `info chroot` per sapere come farlo.
- `OPTIONS="<valore>"`, dove `<valore>` è qualsiasi opzione elencata nella pagina `man` di `named`, eccetto `-t`. Al posto di `-t`, utilizzate la linea `ROOTDIR`.

Per maggiori informazioni sui parametri che potete utilizzare nel file, digitate `man named`, mentre se desiderate informazioni dettagliate su come configurare un server `BIND DNS`, consultate il Capitolo 16. Per default, questo file non contiene alcun parametro.

3.7.1.22. `/etc/sysconfig/netdump`

`/etc/sysconfig/netdump` è il file di configurazione per il servizio `/etc/init.d/netdump`. Il servizio `netdump` invia dati oops e dump di memoria attraverso la rete. In generale, `netdump` non è indispensabile, dunque, a meno che non sia strettamente necessario, non dovrete utilizzarlo. Per maggiori informazioni su quali parametri potete usare nel file, digitate `man netdump`.

3.7.1.23. `/etc/sysconfig/network`

Il file `/etc/sysconfig/network` è utilizzato per specificare le informazioni relative alla configurazione di rete desiderata. È possibile usare i seguenti parametri:

- `NETWORKING=<valore>`, dove `<valore>` può essere sostituito con uno dei seguenti valori:
 - `yes` — la rete deve essere configurata.
 - `no` — la rete non deve essere configurata.
- `HOSTNAME=<valore>`, dove `<valore>` dovrebbe essere sostituito dall'*FQDN (Fully Qualified Domain Name)*, per esempio `hostname.domain.com`, ma può anche essere un nome host di vostra scelta.



Nota Bene

Per questioni di compatibilità con il vecchio software che si desidera installare (per esempio `trn`), il file `/etc/HOSTNAME` deve contenere questi valori.

- `GATEWAY=<valore>`, dove `<valore>` rappresenta l'indirizzo IP del gateway della rete.
- `GATEWAYDEV=<valore>`, dove `<valore>` rappresenta il dispositivo per accedere al gateway, per esempio `eth0`.
- `NISDOMAIN=<valore>`, dove `<valore>` rappresenta il nome del dominio NIS.

3.7.1.24. `/etc/sysconfig/ntpd`

Il file `/etc/sysconfig/ntpd` serve per inviare argomenti al demone `ntpd` durante l'avvio. Il demone `ntpd` imposta l'orologio di sistema in modo che sia sincronizzato con un time server standard di Internet. Implementa la versione 4 del protocollo NTP (Network Time Protocol). Per maggiori informazioni su quali parametri potete utilizzare in questo file, consultate il file `/usr/share/doc/ntp-<versione>/ntpd.htm` (dove `<versione>` è il numero di versione di `ntpd`). Per default, questo file imposta il proprietario del processo `ntpd` come utente `ntp`.

3.7.1.25. `/etc/sysconfig/pcmcia`

Il file `/etc/sysconfig/pcmcia` viene usato per specificare le informazioni di configurazione PCMCIA. È possibile utilizzare i seguenti valori:

- `PCMCIA=<valore>`, dove `<valore>` indica:
 - `yes` — il supporto PCMCIA va abilitato.
 - `no` — il supporto PCMCIA non va abilitato.

- `PCIC=<valore>`, dove `<valore>` può essere sostituito con uno dei seguenti valori:
 - `i82365` — il computer ha un chipset socket PCMCIA di tipo `i82365`.
 - `tcic` — Il computer ha un chipset socket PCMCIA di tipo `tcic`.
- `PCIC_OPTS=<valore>`, dove `<valore>` rappresenta i driver socket (`i82365` o `tcic`).
- `CORE_OPTS=<valore>`, dove `<valore>` rappresenta l'elenco delle opzioni `pcmcia_core`.
- `CARDMGR_OPTS=<valore>`, dove `<valore>` rappresenta l'elenco delle opzioni per il `cardmgr` PCMCIA (come `-q` per la modalità silenziosa e `-m` per la ricerca dei moduli del kernel caricabili nella directory specificata). Per maggiori informazioni, consultate la pagina `man` relativa a `cardmgr`.

3.7.1.26. `/etc/sysconfig/radvd`

Il file `/etc/sysconfig/radvd` serve per inviare argomenti al demone `radvd` durante l'avvio. Il demone resta in attesa di richieste del router e invia messaggi del router per la versione 6 del protocollo IP. Questo servizio consente agli host di una rete di cambiare in modo dinamico i propri router predefiniti sulla base di tali messaggi. Per maggiori informazioni su quali parametri potete utilizzare in questo file, digitate `man radvd`. Per default, questo file imposta il proprietario del processo `radvd` come utente `radvd`.

3.7.1.27. `/etc/sysconfig/rawdevices`

Il file `/etc/sysconfig/rawdevices` viene utilizzato per configurare i collegamenti raw device:

```
/dev/raw/raw1 /dev/sd1
/dev/raw/raw2 8 5
```

3.7.1.28. `/etc/sysconfig/redhat-config-users`

`/etc/sysconfig/redhat-config-users` è il file di configurazione dell'applicazione grafica, **redhat-config-users**. In Red Hat Linux 8.0 questo file serve solo per filtrare gli utenti di sistema come `root`, `daemon` o `lp`. Questo file può essere modificato tramite il menu a tendina **Preferences => Filter system users and groups** dell'applicazione **redhat-config-users** e non deve essere modificato manualmente. Per maggiori informazioni su come utilizzare tale applicazione, consultate il capitolo intitolato *Configurazione di utenti e gruppi* all'interno della *Official Red Hat Linux Customization Guide*.

3.7.1.29. `/etc/sysconfig/redhat-logviewer`

`/etc/sysconfig/redhat-logviewer` è il file di configurazione per l'applicazione di visualizzazione dei log interattivi **redhat-logviewer**. Questo file può essere modificato tramite il menu a tendina **Edit => Preferences** dell'applicazione **redhat-logviewer** e non deve essere modificato manualmente. Per ulteriori informazioni sull'utilizzo di questa applicazione, consultate il capitolo *Log Files* della *Official Red Hat Linux Customization Guide*.

3.7.1.30. `/etc/sysconfig/samba`

Il file `/etc/sysconfig/samba` serve per inviare argomenti ai demoni `smbd` e `nmdbd` durante l'avvio. Il demone `smbd` fornisce connettività di condivisione file per i client Windows nella rete. Il demone `nmdbd` fornisce NetBIOS tramite servizi di denominazione IP. Per maggiori informazioni su quali parametri potete utilizzare in questo file, digitate `man smbd`. Per default, questo file imposta `smbd` e `nmdbd` in modo che siano eseguiti in modalità `daemon`.

3.7.1.31. `/etc/sysconfig/sendmail`

Il file `/etc/sysconfig/sendmail` consente di inviare messaggi a uno o più destinatari, indirizzando il messaggio sulla rete necessaria. Il file imposta i valori di default per eseguire l'applicazione **Sendmail**. I valori di default eseguono il programma come demone in background e ne controllano la coda di attesa ogni ora nel caso in cui qualche messaggio sia tornato indietro.

È possibile usare i seguenti valori:

- `DAEMON=<valore>`, dove `<valore>` può essere sostituito con uno dei seguenti valori:
 - `yes` — **Sendmail** può essere configurato per controllare l'arrivo di posta alla porta 25. `yes` implica l'uso delle opzioni `-bd` di **Sendmail**.
 - `no` — **Sendmail** può non essere configurato per controllare l'arrivo di posta alla porta 25.
- `QUEUE=1h` viene trasmesso a **Sendmail** come `-q$QUEUE`. L'opzione `-q` non viene trasmessa a **Sendmail** se esiste `/etc/sysconfig/sendmail` e `QUEUE` è vuota o non definita.

3.7.1.32. `/etc/sysconfig/soundcard`

Il file `/etc/sysconfig/soundcard` viene generato da `sndconfig` e non deve essere modificato. Va usato solo per determinare quale scheda inserire nel menu, in modo che all'esecuzione successiva di `sndconfig` venga usata come scheda predefinita. Le informazioni sulla configurazione della scheda sonora si trovano nel file `/etc/modules.conf`.

Potrebbe contenere:

- `CARDTYPE=<valore>`, dove `<valore>` può essere per esempio, `SB16`, nel caso di una scheda audio Soundblaster 16.

3.7.1.33. `/etc/sysconfig/squid`

Il file `/etc/sysconfig/squid` serve per inviare argomenti al demone `squid` durante l'avvio. Il demone è un server proxy per applicazioni Web client. Per maggiori informazioni su come configurare un server proxy `squid`, utilizzate un browser Web per aprire la directory `/usr/share/doc/squid-<versione>/` (dove `<versione>` va sostituito con il numero di versione di `squid` installato sul sistema). Per default, questo file imposta `squid` in modo che si avvii in modalità `daemon` e stabilisce la quantità di tempo che trascorre prima che esso si arresti.

3.7.1.34. `/etc/sysconfig/tux`

`/etc/sysconfig/tux` è il file di configurazione per il server Web Red Hat Accelerator basato sul kernel (in precedenza noto come TUX). Per maggiori informazioni su come configurare il Red Hat Accelerator, utilizzate un browser Web per aprire `/usr/share/doc/tux-<versione>/tux/index.html` (dove `<versione>` va sostituito con

il numero di versione di TUX installato sul sistema). I parametri disponibili per questo file sono elencati in `/usr/share/doc/tux-<versione>/tux/parameters.html`.

3.7.1.35. `/etc/sysconfig/ups`

Il file `/etc/sysconfig/ups` è utilizzato per specificare le informazioni relative a qualsiasi *UPS* (*Uninterruptible Power Supplies*) collegato al vostro sistema. Un UPS può essere molto utile per un sistema Red Hat Linux perché vi dà il tempo di chiudere correttamente il sistema in caso di interruzione della corrente elettrica. Si possono usare i seguenti valori:

- `SERVER=<valore>`, dove `<valore>` può essere sostituito con uno dei seguenti valori:
 - `yes` — è collegato un dispositivo UPS al sistema.
 - `no` — non è collegato alcun dispositivo UPS al sistema.
- `MODEL=<valore>`, dove `<valore>` deve essere impostato su `NONE` se nessun dispositivo UPS è collegato al sistema oppure deve essere:
 - `apcsmart` — per un APC SmartUPS™ oppure un dispositivo simile.
 - `fentonups` — per un UPS™ Fenton.
 - `optiups` — per un dispositivo UPS™ OPTI.
 - `bestups` — per un UPS Best Power™.
 - `genericups` — per un dispositivo UPS generico.
 - `ups-trust425+625` — per un UPS Trust™.
- `DEVICE=<valore>`, dove `<valore>` specifica il punto in cui è collegato il dispositivo UPS, per esempio: `/dev/ttyS0`.
- `OPTIONS=<valore>`, dove `<valore>` è un comando speciale da trasmettere al dispositivo UPS.

3.7.1.36. `/etc/sysconfig/vncservers`

Il file `/etc/sysconfig/vncservers` configura l'avvio del server VNC (*Virtual Network Computing*). VNC è un sistema di visualizzazione remoto che consente di mostrare un ambiente desktop non solo sull'elaboratore dove è in esecuzione ma anche su reti diverse (da una LAN a Internet), utilizzando una vasta gamma di architetture.

Può contenere:

- `VNCSERVERS=<valore>`, dove `<valore>` è impostato come `"1:fred"` per indicare che un server VNC deve essere avviato dall'utente fred per il display :1. L'utente fred deve avere configurato una password VNC utilizzando il comando `vncpasswd` prima di connettersi al server VNC remoto.

Quando utilizzate un server VNC, le comunicazioni non sono criptate e perciò il server VNC non deve essere utilizzato su una rete non sicura. Per istruzioni particolari riguardanti l'uso di SSH per rendere le comunicazioni VNC sicure, consultate le informazioni all'indirizzo <http://www.uk.research.att.com/vnc/sshvnc.html>. Per maggiori informazioni su SSH consultate il Capitolo 9 o la *Official Red Hat Linux Customization Guide*.

3.7.1.37. /etc/sysconfig/xinetd

Il file `/etc/sysconfig/xinetd` serve per inviare argomenti al demone `xinetd` durante l'avvio. Il demone avvia programmi che forniscono servizi Internet quando viene ricevuta una richiesta sulla porta per quel servizio. Per maggiori informazioni su quali parametri potete utilizzare in questo file digitate `man xinetd`. Per informazioni sul servizio `xinetd`, consultate la Sezione 8.3.

3.7.2. Sottodirectory della directory /etc/sysconfig/

Le seguenti directory si trovano normalmente in `/etc/sysconfig/`, insieme a una breve descrizione di ciò che contengono:

- `apm-scripts` — contiene lo script di sospensione/ripristino Red Hat APM. Non modificate questo file manualmente. Se avete esigenza di personalizzarlo, create semplicemente un file denominato `/etc/sysconfig/apm-scripts/apmcontinue` e verrà richiamato in fondo allo script. Potete anche controllare lo script modificando `/etc/sysconfig/apmd`.
- `cbq` — questa directory contiene i file di configurazione necessari per effettuare il Class Based Queuing per la gestione della larghezza di banda sulle interfacce di rete.
- `networking` — è utilizzata dal tool **Red Hat Network Administration Tool** e non è consigliabile modificarla manualmente. Per sapere come configurare le interfacce tramite **Red Hat Network Administration Tool**, consultate il capitolo intitolato *Configurazione della rete* nella *Official Red Hat Linux Customization Guide*.
- `network-scripts` — contiene i seguenti file di configurazione relativi alla rete:
 - File di configurazione della rete per ogni interfaccia di rete configurata (per esempio il file `ifcfg-eth0` per l'interfaccia Ethernet `eth0`).
 - Script utilizzati per attivare e disattivare interfacce di rete, come `ifup` e `ifdown`.
 - Script utilizzati per attivare e disattivare interfacce ISDN, come `ifup-isdn` e `ifdown-isdn`.
 - Svareti script di funzione di rete che non devono essere modificati manualmente.

Per maggiori informazioni sulla directory `network-scripts`, consultate il Capitolo 12.

- `rhn` — contiene i file di configurazione per **Red Hat Network Registration Client**, **Red Hat Update Agent Configuration Tool**, **Red Hat Update Agent** e l'applet del pannello **Red Hat Network Alert Notification Tool** nonché le chiavi `systemid` e GPG. Nessuno dei file contenuti in questa directory deve essere modificato manualmente. Per maggiori informazioni su Red Hat Network, consultate il relativo sito Web all'indirizzo <https://rhn.redhat.com/>.

3.8. Chiusura del sistema

Per chiudere correttamente Red Hat Linux, usate il comando `shutdown`. Potete leggere la pagina `man shutdown` per maggiori dettagli, ma i due utilizzi più comuni sono:

```
/sbin/shutdown -h now
/sbin/shutdown -r now
```

È necessario eseguire il comando `shutdown` come utente `root`. Dopo aver chiuso tutto, l'opzione `-h` spegne l'elaboratore e l'opzione `-r` lo riavvia.

Nessun utente non `root` può utilizzare i comandi `reboot` e `halt` per arrestare il sistema in un `runlevel` compreso tra 1 e 5. Non tutti i sistemi operativi Linux supportano questa caratteristica.

Se il computer non si spegne da solo, aspettate che a video compaia un messaggio che vi comunica l'avvenuto arresto del sistema prima di premere il pulsante di spegnimento.

Se spegnete l'elaboratore prima di questo messaggio, potreste impedire che le partizioni del disco fisso vengano smontate. Questo può causare un danneggiamento dei filesystem, al punto che il sistema potrebbe non riuscire a effettuare l'avvio.

Prima di poter eseguire Red Hat Linux sul vostro sistema, è necessario che venga avviato da un programma speciale denominato *boot loader*, ovvero un codice presente nel disco rigido primario o su un altro dispositivo di supporto, responsabile del caricamento in memoria dei file necessari per il kernel di Linux e, in alcuni casi, di altri sistemi operativi.

4.1. Boot loader e architettura di sistema

Ogni architettura di sistema che può eseguire Red Hat Linux utilizza un boot loader diverso. Per esempio, l'architettura Alpha utilizza il boot loader `aboot`, mentre l'architettura Itanium impiega il boot loader `ELILO`.

Questo capitolo esamina comandi e opzioni di configurazione per i due boot loader disponibili con Red Hat Linux 8.0 per l'architettura x86, cioè GRUB e LILO.

4.2. GRUB

GNU GRUB (GRand Unified Bootloader) è un programma che consente di selezionare quale sistema operativo o kernel caricare al momento dell'avvio del sistema. Consente inoltre di passare argomenti al kernel.

4.2.1. Il processo di avvio di GRUB e x86

Questa sezione esaminerà in maggiore dettaglio il ruolo specifico di GRUB all'avvio di un sistema x86. Per informazioni dettagliate sul processo globale di avvio, consultate la Sezione 3.2.

Il processo di caricamento di GRUB avviene in diverse fasi:

1. *Caricamento del boot loader primario dal BIOS nell'MBR - Fase 1¹*. Il boot loader primario è posizionato nel piccolissimo spazio assegnato all'MBR, inferiore a 512 byte. Per questo motivo, l'unica operazione effettuata dal boot loader primario è il caricamento del boot loader della fase intermedia o della fase secondaria. Ciò è dovuto al fatto che lo spazio nell'MBR è insufficiente per qualsiasi altra operazione.
2. *Il boot loader della fase intermedia è caricato in memoria da quello della fase 1 solo se necessario*. Alcuni elementi hardware richiedono una fase intermedia per giungere al boot loader secondario. Questo avviene quando la partizione `/boot` è superiore ai 1024 cilindri della testina del disco fisso o quando si utilizza la modalità LBA. Il boot loader della fase intermedia è disponibile nella partizione `/boot` o in una piccola parte dell'MBR e della partizione `/boot`.
3. *Caricamento del boot loader secondario - Fase 2*. Il boot loader secondario visualizza il menu e l'ambiente dei comandi di GRUB. Questa interfaccia vi consente di selezionare il sistema operativo o il kernel da avviare, di passare gli argomenti al kernel o di osservare i parametri di sistema, come la RAM disponibile.
4. *Caricamento del sistema operativo o del kernel e di `initrd`*. Dopo che GRUB ha determinato il sistema operativo da avviare, lo carica in memoria e cede il controllo del calcolatore al sistema operativo.

1. Per ulteriori informazioni sul BIOS di sistema e sull'MBR, consultate la Sezione 3.2.1.

Il metodo utilizzato per avviare Red Hat Linux è denominato *caricamento diretto* perché il boot loader carica direttamente il sistema operativo. Non esiste alcun intermediario tra il boot loader e il kernel.

Il processo di avvio utilizzato da altri sistemi operativi può variare leggermente. I sistemi operativi DOS e Windows di Microsoft, per esempio, oltre a numerosi altri sistemi operativi proprietari, vengono caricati mediante il metodo di *caricamento a catena*. Con questo metodo l'MBR fa semplicemente riferimento al primo settore della partizione contenente il sistema operativo, dove trova i file necessari per avviare il sistema operativo.

GRUB supporta entrambi i metodi di avvio di caricamento a catena e diretto, consentendovi di utilizzarli con qualsiasi sistema operativo.



Avvertenza

Durante l'installazione, il programma di installazione di DOS e Windows di Microsoft sovrascrive completamente l'MBR, eliminando qualsiasi boot loader esistente. Se create un sistema a doppio avvio, è preferibile installare prima il sistema operativo Microsoft. Per istruzioni su come effettuare questa operazione, consultate l'appendice *Installing Red Hat Linux in a Dual-Boot Environment* nella *Official Red Hat Linux Installation Guide*.

4.2.2. Caratteristiche di GRUB

GRUB vanta una serie di caratteristiche che lo rendono preferibile ad altri boot loader disponibili. Ecco alcune delle peculiarità più importanti:

- *GRUB fornisce ai calcolatori x86 un ambiente pre-OS basato su comandi.* In questo modo l'utente dispone della massima flessibilità nel caricamento dei sistemi operativi con determinate opzioni o nella raccolta di informazioni sul sistema. Molte architetture, diverse da x86, hanno utilizzato per anni ambienti pre-OS che consentono di controllare il modo in cui il sistema si avvia da una linea di comando. Mentre alcune caratteristiche di comando sono disponibili anche in LILO e altri boot loader x86, GRUB contiene un maggior numero di caratteristiche.
- *GRUB supporta la modalità LBA Logical Block Addressing.* La modalità LBA posiziona nella firmware dell'unità la conversione di indirizzamento utilizzata per trovare i file del disco fisso. Prima dell'introduzione di questa modalità, i boot loader potevano essere limitati dal cilindro 1024 e il BIOS non riusciva a individuare dei file dopo quel punto. Il supporto LBA consente a GRUB di avviare i sistemi operativi dalle partizioni oltre il limite del cilindro 1024, se il BIOS supporta tale modalità (la maggior parte dei BIOS la supporta).
- *GRUB può leggere le partizioni ext2.* GRUB può accedere al relativo file di configurazione `/boot/grub/grub.conf` a ogni avvio del sistema, evitandovi dunque di dover scrivere una nuova versione del boot loader primario nell'MBR tutte le volte che modificate le opzioni. L'unico caso in cui potrebbe essere necessario reinstallare GRUB nell'MBR si verifica se il percorso fisico della partizione `/boot` viene spostato sul disco. Per ulteriori informazioni sull'installazione di GRUB nell'MBR, consultate la Sezione 4.3.

4.3. Installazione di GRUB

Se non avete installato GRUB durante il processo di installazione di Red Hat Linux, ecco il modo di farlo in seguito e di renderlo il vostro boot loader predefinito.

Prima di installare GRUB, dovete accertarvi di disporre dell'ultima versione disponibile di GRUB oppure potete utilizzare il pacchetto GRUB contenuto nel CD di installazione di Red Hat Linux. Per

istruzioni sull'installazione delle diverse versioni, consultate il capitolo *Package Management with RPM* nella *Official Red Hat Linux Customization Guide*.

Dopo avere installato il pacchetto GRUB, aprire un prompt della shell root ed eseguite il comando `/sbin/grub-install <percorso>`, dove `<percorso>` è, la posizione in cui deve essere installato il boot loader GRUB primario.

Il comando riportato di seguito installa GRUB nell'MBR del dispositivo IDE master del bus IDE primario, anche noto come unità C:

```
/sbin/grub-install /dev/hda
```

Al successivo avvio del sistema dovrete visualizzare il menu del boot loader grafico GRUB prima del caricamento del kernel.

4.4. Terminologia

Una delle cose più importanti da capire prima di usare GRUB è il modo in cui il programma fa riferimento a dispositivi come i dischi fissi o le partizioni. Questa informazione è fondamentale se desiderate configurare GRUB per l'avvio di più sistemi operativi.

4.4.1. Nomi dei dispositivi

Il primo disco fisso di un sistema è definito da GRUB (`hd0`). La prima partizione in questa unità disco si chiama (`hd0,0`) e la quinta partizione sul secondo disco fisso viene definita (`hd1,4`). In generale, la convenzione di denominazione utilizzata da GRUB per i filesystem è la seguente:

```
(<tipo-di-dispositivo><bios-numero-dispositivo>,<numero-partizione>)
```

Le parentesi e la virgola sono molto importanti nel nome. Il `<tipo-di-dispositivo>` indica l'unità specificata, disco fisso (`hd`) o disco floppy (`fd`).

Il `<bios-numero-dispositivo>` è il numero del dispositivo in base al BIOS del vostro sistema che inizia con 0. All'unità disco primaria IDE viene assegnato il numero 0, mentre all'unità disco secondaria IDE viene assegnato il numero 1. L'assegnazione sequenziale dei numeri equivale all'incirca al modo in cui il kernel Linux ordina i dispositivi per lettera, quindi la lettera a in `hda` equivale al numero 0, la lettera b in `hdb` al numero 1 e così via.



Note Bene

Il sistema di numerazione di GRUB parte da 0 e non da 1. Questo è uno degli errori più comuni commessi dai nuovi utenti di GRUB.

Il `<numero-partizione>` indica il numero di una partizione specifica del dispositivo. Come per il `<bios-numero-dispositivo>`, la numerazione delle partizioni inizia da 0. Nonostante la maggior parte delle partizioni sia contraddistinta da numeri, le partizioni BSD sono invece rappresentate da lettere come a o c.

Per assegnare un nome a dispositivi e partizioni, GRUB utilizza le regole seguenti:

- Tutti i dischi fissi, non importa se sono IDE o SCSI, iniziano con `hd`. Le unità floppy iniziano con `fd`.
- Per specificare un dispositivo intero, senza tener conto delle sue partizioni, occorre semplicemente tralasciare la virgola e il numero di partizione. Questo aspetto è importante quando GRUB deve

configurare l'MBR per un'unità particolare. Per esempio (hd0) indica il primo dispositivo mentre (hd3) indica il quarto.

- Se possedete diversi dischi fissi, è importante conoscere il loro ordine in base al BIOS. Non è difficile scoprirlo se i dischi sono o solo IDE o solo SCSI, se invece sono misti, diventa tutto più complesso.

4.4.2. Nomi dei file

Quando digitate comandi che comprendono un file, come un elenco di menu da utilizzare per permettere l'avvio di più sistemi operativi, è necessario includere il file subito dopo aver specificato il dispositivo e la partizione. Ecco un esempio di come viene indicato il file in un percorso assoluto:

```
(<tipo-di-dispositivo><bios-numero-dispositivo>,<numero-partizione>)/percorso/per/file
```

La maggior parte delle volte i file vengono specificati mediante il percorso nella partizione e il nome del file.

Potete inoltre indicare a GRUB dei file che non compaiono nel filesystem, come per esempio un loader a catena (chainloader) contenuto nei primissimi blocchi di una partizione. Per specificare questi file, occorre fornire un *elenco dei blocchi*, che indichi a GRUB, blocco per blocco, la posizione del file nella partizione. Poiché un file può essere composto da molti insiemi di blocchi differenti, esiste un modo specifico di scrivere gli elenchi dei blocchi. Tutte le posizioni dei file nelle sezioni vengono descritte dal blocco iniziale a cui si aggiunge il numero di blocchi utilizzati, infine le sezioni vengono raggruppate in ordine e separate da virgole.

Considerate il seguente elenco di blocchi:

```
0+50,100+25,200+1
```

L'elenco sopra descritto indica a GRUB di utilizzare un file che inizia nel primo blocco della partizione e utilizza dal blocco 0 al blocco 49, dal 99 al 124 e il blocco 199.

Sapere come scrivere gli elenchi dei blocchi è utile quando si usa GRUB per caricare dei sistemi operativi che utilizzano il caricamento a catena, come Microsoft Windows. Potete omettere il numero di blocchi se iniziate dal blocco 0. Per esempio, il file di caricamento a catena nella prima partizione del primo disco fisso avrà il seguente nome:

```
(hd0,0)+1
```

Potete inoltre utilizzare il comando `chainloader` con la stessa indicazione nella linea di comando di GRUB dopo aver impostato come root il dispositivo e la partizione corretti:

```
chainloader +1
```

4.4.3. Il filesystem root di GRUB

Alcuni utenti si confondono nell'utilizzare il termine "filesystem root" con GRUB. È importante ricordare che il filesystem root di GRUB non ha nulla a che fare con il filesystem root di Linux.

Il filesystem root di GRUB è la partizione di root per un dispositivo particolare. GRUB utilizza queste informazioni anche per montare il dispositivo e caricarne i file.

Con Red Hat Linux, una volta che GRUB ha caricato la partizione di root contenente il kernel Linux ed è uguale alla partizione `/boot`, potete eseguire il comando `kernel` indicando come opzione la posizione del file del kernel. Quando il kernel Linux si avvia, installa i propri filesystem root. Il

filesystem root originale di GRUB e quanto con esso correlato vengono rimossi. Il loro unico scopo è quello di caricare il file del kernel.

Per maggiori informazioni, consultate le note relative ai comandi `root` e `kernel` nel la Sezione 4.6.

4.5. Interfacce di GRUB

GRUB dispone di tre potenti interfacce che forniscono diversi livelli di funzionalità. Ognuna di queste interfacce vi permette di avviare dei sistemi operativi e persino di spostarvi da un'interfaccia all'altra all'interno dell'ambiente GRUB.

4.5.1. Interfaccia a menu

Se GRUB è stato configurato in modo automatico dal programma d'installazione di Red Hat Linux, questa è l'interfaccia che già conoscete. Essa presenta un menu dei sistemi operativi o dei kernel preconfigurati con i relativi comandi di avvio, a seconda del nome. Potete utilizzare i tasti freccia per selezionare un'opzione diversa da quella predefinita e premere [Invio] per avviarla. In alternativa è possibile impostare un periodo di timeout, dopo il quale GRUB inizia a caricare l'opzione predefinita.

Dal menu a interfaccia potete inoltre digitare [e] per modificare i comandi della voce di menu evidenziata oppure [c] per visualizzare un'interfaccia a linea di comando.

Per ulteriori informazioni sulla configurazione di questa interfaccia, consultate la Sezione 4.7.

4.5.2. Interfaccia Editor per voci di menu

Per passare a questa interfaccia è necessario premere il tasto [e] dal menu del boot loader. I comandi di GRUB per questa voce vengono visualizzati in questo ambito ed è possibile modificare queste linee di comando, prima di avviare il sistema operativo, aggiungendole ([o] inserisce la nuova linea dopo la riga corrente e [O] prima della riga corrente), modificandole ([e]) o cancellandole ([d]).

Dopo aver effettuato le modifiche, potete digitare [b] per applicare tali modifiche e avviare il sistema operativo. Il tasto [Esc] vi riporta all'interfaccia a menu standard senza applicare le modifiche. Invece digitando [c] potete visualizzare l'interfaccia a linea di comando.

4.5.3. Interfaccia a linea di comando

Si tratta dell'interfaccia di base di GRUB, ma è anche quella che vi fornisce il maggior controllo. Infatti qui è possibile digitare tutti i comandi di GRUB e premere semplicemente [Invio] per eseguirli. Questa interfaccia dispone di caratteristiche avanzate simili a quelle della shell, tra cui la funzione di completamento automatico dei comandi, basata sul contesto, con il tasto [Tab] e le combinazioni di tasti [Ctrl] quando si digitano comandi come [Ctrl]-[a] per spostarsi all'inizio di una riga e [Ctrl]-[e] per spostarsi alla fine. Inoltre i tasti freccia, [Home], [Fine] e [Canc] funzionano come nella shell `bash`.

Per un elenco di comandi comuni, consultate la Sezione 4.6.

4.5.4. Sequenza di utilizzo delle interfacce

Quando l'ambiente GRUB inizia a caricare il boot loader secondario, cerca il suo file di configurazione. Quando lo ha individuato lo utilizza per creare l'elenco di menu e visualizza l'interfaccia a menu di avvio.

Se non è possibile individuare il file di configurazione oppure se questo non è leggibile, GRUB visualizza l'interfaccia a linea di comando per permettervi di digitare manualmente i comandi necessari all'avvio di un sistema operativo.

Se il file di configurazione non è valido, GRUB visualizza l'errore e richiede un input. Ciò può essere molto utile perché vi consente di vedere esattamente dove si è verificato il problema e di risolverlo nel file. Premendo un qualsiasi tasto tornerete al menu a interfaccia, dove potrete modificare l'opzione di menu e correggere il problema in base all'errore segnalato da GRUB. Se la correzione non ha buon esito, l'errore viene segnalato e potete ricominciare da capo.

4.6. Comandi

GRUB dispone di numerosi comandi nell'interfaccia a linea di comando. Per alcuni di questi comandi è possibile digitare delle opzioni, dopo il nome, che vanno separate tramite spazi dal comando e da altre opzioni.

Qui di seguito è riportato un elenco dei comandi più utili:

- `boot` — consente di avviare il sistema operativo o il loader a catena (chainloader) specificato e caricato in precedenza.
- `chainloader <nome-file>` — permette di caricare il file specificato come un loader a catena. Per prelevare il file nel primo settore della partizione indicata, potete utilizzare come nome del file `+1`.
- `displaymem` — serve a visualizzare lo spazio di memoria disponibile, in base alle informazioni fornite dal BIOS. Questo comando è utile per determinare la quantità di RAM di cui dispone il sistema prima di avviarlo.
- `initrd <nome-file>` — consente di indicare un RAM disk iniziale da utilizzare all'avvio, necessario quando il kernel richiede alcuni moduli per effettuare un avvio corretto, come nel caso in cui la partizione root viene formattata con il filesystem ext3.
- `install <fase-1> <installa-disco> <fase-2> p <file-config>` — serve a installare GRUB nel vostro MBR.



Avvertenza

Il comando `install` sovrascrive qualsiasi informazione presente nell'MBR. Se lo eseguite, tutti i dati (non relativi a GRUB) necessari per avviare altri sistemi operativi verranno cancellati.

Prestate dunque molta attenzione nell'eseguirlo.

Questo comando può essere utilizzato in molti modi diversi. È comunque necessario specificare una `<fase-1>`, che caratterizza il dispositivo, la partizione o il file contenenti l'immagine del boot loader primario, come `(hd0,0)/grub/stage1`. Inoltre dovete indicare il disco in cui installare il boot loader primario, come `(hd0)`.

La sezione `<fase-2>` indica al boot loader primario la posizione del boot loader secondario, come `(hd0,0)/grub/stage2`. L'opzione `p` indica al comando `install` che è stato specificato un file di configurazione del menu nella sezione `<file-config>`, come `(hd0,0)/grub/grub.conf`.

- `kernel <nome-file-kernel> <opzione-1> <opzione-N>` — specifica il file kernel da caricare dal filesystem root di GRUB nel caso sia utilizzato il caricamento diretto per avviare il sistema operativo. Le opzioni possono seguire il comando `kernel` trasmesso al kernel una volta caricato.

Per Red Hat Linux è probabile che venga visualizzata una riga simile alla seguente:

```
kernel /vmlinuz root=/dev/hda5
```

Questa linea indica che il file `vmlinuz` viene caricato dal filesystem root di GRUB, come `(hd0,0)`. Al kernel viene inoltre trasmessa un'opzione indicante che il filesystem root per il kernel di Linux dovrà trovarsi in `hda5`, ovvero la quinta partizione sul primo disco rigido IDE. È possibile specificare altre opzioni dopo questa, se necessario.

- `root <dispositivo-e-partizione>` — serve a configurare la partizione di root di GRUB come dispositivo e partizione particolari, come `(hd0,0)`, e monta la partizione in modo tale che i file possano essere letti.
- `rootnoverify <dispositivo-e-partizione>` — ha la stessa funzione del comando `root` ma non monta la partizione.

Sono disponibili altri comandi oltre a questi. Per visualizzarne un elenco completo, digitate `info grub`.

4.7. File di configurazione del menu GRUB

Il file di configurazione usato per creare l'elenco dei sistemi operativi da avviare nell'interfaccia a menu, consente all'utente di selezionare un gruppo di comandi preimpostati da eseguire. Possono essere usati i comandi indicati nella Sezione 4.6 nonché alcuni comandi speciali utilizzabili solo nel file di configurazione.

4.7.1. Comandi speciali del file di configurazione

I comandi che seguono sono utilizzabili solo nel file di configurazione di GRUB:

- `color <colore-normale> <colore-selezionato>` — consente di impostare determinati colori da utilizzare nel menu, uno per il primo piano e l'altro per lo sfondo. È sufficiente indicare i nomi dei colori, come `red/black`. Ecco un esempio di linea:
`color red/black green/blue`
- `default <nome-title>` — si tratta del nome della voce 'title' predefinita da caricare in caso di timeout dell'interfaccia a menu.
- `fallback <nome-title>` — indica il nome della voce 'title' da provare se il primo tentativo fallisce.
- `hiddenmenu` — impedisce la visualizzazione del menu a interfaccia di GRUB caricando la voce `default` al termine del periodo di `timeout`. L'utente può vedere il menu standard di GRUB premendo [Esc].
- `password <password>` — impedisce agli utenti che non conoscono la password di modificare le voci di questa opzione del menu.

Se lo desiderate, potete indicare un file di configurazione del menu alternativo dopo il comando `<password>`. In tal modo, se viene indicata la password, GRUB si riavvia dal boot loader secondario. Se questo file alternativo non viene indicato nel comando, un utente che conosce la password può modificare il file di configurazione in uso al momento.

- `timeout` — questo comando imposta il tempo, in secondi, prima che GRUB avvii la voce indicata dal comando `default`.
- `splashimage` — specifica la posizione dell'immagine splash screen da utilizzare all'avvio di GRUB.
- `title` — imposta un nome da utilizzare con un particolare gruppo di comandi utilizzati per caricare un sistema operativo.

Il carattere `#` può essere utilizzato per inserire dei commenti nel file di configurazione del menu.

4.7.2. Struttura del file di configurazione

Il file di configurazione dell'interfaccia a menu di GRUB è `/boot/grub/grub.conf`. I comandi per le impostazioni generali dell'interfaccia a menu si trovano all'inizio del file e sono seguiti da diverse voci per ognuno dei sistemi operativi o kernel elencati nei menu.

Ecco un esempio di file di configurazione di base del menu di GRUB per l'avvio di Red Hat Linux o di Microsoft Windows 2000:

```
default=0
timeout=10
splashimage=(hd0,0)/grub/splash.xpm.gz

# section to load linux
title Red Hat Linux (2.4.18-5.47)
    root (hd0,0)
    kernel /vmlinuz-2.4.18-5.47 ro root=/dev/sda2
    initrd /initrd-2.4.18-5.47.img

# section to load Windows 2000
title windows
    rootnoverify (hd0,0)
    chainloader +1
```

Questo file indica a GRUB di creare un menu con Red Hat Linux come sistema operativo predefinito che si avvia dopo 10 secondi. Vengono fornite due sezioni, una per ogni sistema operativo indicato, con comandi specifici per la tabella delle partizioni di questo sistema.



Nota Bene

L'elemento predefinito è specificato come numero che si riferisce alla prima linea `title` che GRUB rileva. Se desiderate che `windows` sia il valore predefinito, modificate `default=` in 1.

Tuttavia impostare il file di configurazione del menu di GRUB per l'avvio di più sistemi operativi esula dallo scopo di questo capitolo. Per informazioni più dettagliate relative all'avvio di vari sistemi operativi con GRUB, consultate la Sezione 4.11.

4.8. LILO

LILO è l'acronimo di *Linux LOader* ed è stato utilizzato per avviare Linux nei sistemi x86 per molti anni. Anche se GRUB è ora il boot loader predefinito, alcuni preferiscono LILO perché è più familiare, mentre altri lo utilizzano per necessità, dato che GRUB può creare problemi di avvio relativi all'hardware.

4.8.1. Processo di avvio di LILO e x86

Questa sezione esaminerà in dettaglio il ruolo specifico di LILO all'avvio di un sistema x86. Per un'analisi dettagliata del processo globale di avvio, consultate la Sezione 3.2.

LILO viene caricato in memoria in modo quasi identico a GRUB, a eccezione del fatto che si tratta di un'operazione di sole due fasi.

1. *Il boot loader primario viene caricato dal BIOS dell'MBR - Fase 1².* Il boot loader primario è posizionato nel piccolissimo spazio assegnato all'MBR, inferiore a 512 byte. L'unica operazione effettuata dal boot loader è il caricamento del boot loader secondario e il passaggio a quest'ultimo delle informazioni sulla geometria del disco.
2. *Caricamento del boot loader secondario - Fase 2.* Il boot loader secondario consente di visualizzare la schermata iniziale di Red Hat Linux che permette di selezionare il sistema operativo o il kernel di Linux da avviare.
3. *Caricamento del sistema operativo o kernel e di `initrd` da parte del boot loader secondario.* Dopo che LILO ha determinato quale sistema operativo avviare, lo carica in memoria e passa il controllo del calcolatore al sistema operativo.

Al termine del caricamento del boot loader secondario, LILO visualizza la schermata iniziale di Red Hat Linux con i vari sistemi operativi o kernel che sono stati configurati per l'avvio. Se è stato installato solo Red Hat Linux e non è stata effettuata alcuna modifica al file di configurazione di LILO, solo **linux** verrà visualizzato come opzione. Se installate il supporto per il kernel SMP, disporrete dell'opzione **linux-up**. Se avete configurato LILO per l'avvio di altri sistemi operativi, questa schermata vi consentirà di selezionare il sistema operativo da avviare. Utilizzate i tasti direzionali per evidenziare il sistema operativo e premete [Invio].

Se desiderate un prompt dei comandi per digitare un comando in LILO, premete [Ctrl]-[X]. LILO visualizzerà un prompt `LILO:` e attenderà l'input da parte dell'utente.

4.8.2. LILO e GRUB a confronto

In generale LILO funziona in modo simile a GRUB con l'eccezione di tre differenze principali:

- Non dispone di alcuna interfaccia di comando interattiva.
- Memorizza le informazioni sul percorso del kernel o di un altro sistema operativo da caricare nell'MBR.
- Non è in grado di leggere le partizioni ext2.

Il primo punto indica che il prompt dei comandi per LILO non è interattivo e consente un solo comando con argomenti.

Gli ultimi due punti indicano che se modificate il file di configurazione di LILO o installate un nuovo kernel, dovete riscrivere il boot loader primario di LILO nell'MBR eseguendo il comando `/sbin/lilo -v -v`. Questa operazione è più rischiosa del metodo di GRUB, perché un MBR configurato in modo scorretto non consente l'avvio del sistema. In GRUB se il file di configurazione viene configurato in modo non appropriato, utilizzerà semplicemente l'interfaccia a linea di comando predefinita.



Suggerimento

Se aggiornate il kernel mediante **Red Hat Update Agent**, l'MBR verrà aggiornato automaticamente. Per ulteriori informazioni su RHN, fate riferimento al seguente URL, <https://rhn.redhat.com>

2. Per ulteriori informazioni sul BIOS di sistema e sull'MBR, consultate la Sezione 3.2.1.

4.9. Opzioni di `/etc/lilo.conf`

Il comando `/sbin/lilo -v -v` accede al file di configurazione di LILO `/etc/lilo.conf` per determinare cosa scrivere nell'MBR. Se pensate di utilizzare il comando `lilo`, è necessario che sappiate come modificare questo file.



Avvertenza

Se pensate di modificare `/etc/lilo.conf`, assicuratevi di eseguire una copia di backup del file prima di effettuare qualsiasi modifica. Assicuratevi inoltre di disporre di un disco floppy di avvio funzionante per poter avviare il sistema ed effettuare le modifiche all'MBR se si verificassero problemi. Per ulteriori informazioni sulla creazione di un disco di avvio, consultate le pagine man per `mkbootdisk`.

Il file `/etc/lilo.conf` viene utilizzato da `lilo` per determinare quale sistema operativo o kernel avviare, oltre a sapere dove installarsi, per esempio `/dev/hda` per il primo MBR del primo disco fisso IDE. Un file `/etc/lilo.conf` di esempio è simile a questo, mentre il vostro file `/etc/lilo.conf` può essere in parte diverso:

```
boot=/dev/hda
map=/boot/map
install=/boot/boot.b
prompt
timeout=50
message=/boot/message
lba32
default=linux

image=/boot/vmlinuz-2.4.0-0.43.6
label=linux
initrd=/boot/initrd-2.4.0-0.43.6.img
read-only
root=/dev/hda5

other=/dev/hda1
label=dos
```

Questo esempio mostra un sistema configurato per avviare due sistemi operativi, Red Hat Linux e DOS. Di seguito viene riportata un'analisi più approfondita di alcune linee di questo file:

- `boot=/dev/hda` indica a LILO di installarsi nel primo disco fisso del primo controller IDE.
- `map=/boot/map` localizza il file della mappa. In genere non deve essere modificato.
- `install=/boot/boot.b` indica a LILO di installare il file specificato come nuovo settore di avvio. In genere non deve essere modificato. Se manca la linea `install`, LILO presupporrà un valore predefinito di `/boot/boot.b` come file da utilizzare.
- L'esistenza di `prompt` indica a LILO di mostrare qualsiasi elemento sia referenziato nella linea `message`. Mentre non è consigliabile rimuovere la linea `prompt`. Se la rimuovete, potete comunque ottenere un prompt tenendo premuto il tasto [Maiusc] mentre il computer inizia l'avvio.
- `timeout=50` imposta la quantità di tempo di attesa di LILO per l'input dell'utente prima di procedere con l'avvio della voce della linea `default`. Questo valore è misurato in decine di secondi, con 50 come valore predefinito.
- `message=/boot/message` fa riferimento alla schermata che LILO visualizza per consentirvi di selezionare il sistema operativo o il kernel da avviare.

- `lba32` descrive la geometria del disco fisso a LILO. Un'altra voce comune in questo caso è `linear`. Non dovete modificare questa linea se non siete certi di cosa state facendo. In caso contrario, potrebbe essere impossibile avviare il sistema.
- `default=linux` fa riferimento al sistema operativo predefinito per l'avvio da parte di LILO dalle opzioni elencate sotto questa linea. Il nome `linux` fa riferimento alla linea `label` di ciascuna opzione di avvio.
- `image=/boot/vmlinuz-2.4.0-0.43.6` specifica l'avvio del kernel di Linux con questa particolare opzione.
- `label=linux` nomina l'opzione del sistema operativo nella schermata di LILO. In questo caso si tratta anche del nome indicato dalla linea `default`.
- `initrd=/boot/initrd-2.4.0-0.43.6.img` fa riferimento all'immagine del *ram disk iniziale* che viene utilizzata all'avvio per inizializzare e avviare realmente i dispositivi che rendono possibile l'avvio del kernel. Il *ram disk iniziale* è una serie di driver specifica dei computer necessaria per far funzionare una scheda SCSI, un disco fisso o qualsiasi altro dispositivo necessario per caricare il kernel. Non dovete mai tentare di condividere i *ram disk* iniziali tra computer diversi.
- `read-only` specifica che la partizione `root` (fate riferimento alla linea `root` sottostante) è di sola lettura e non può essere modificata durante il processo di avvio.
- `root=/dev/hda5` indica a LILO quale partizione del disco utilizzare come partizione `root`.

4.10. Modifica dei runlevel all'avvio

In Red Hat Linux è possibile modificare il runlevel predefinito all'avvio.

Se utilizzate LILO come boot loader, effettuate l'accesso al prompt `boot`: digitando [Ctrl]-[X]. Digitate quindi:

```
linux numero
```

In questo comando sostituite *numero* con il numero del runlevel che desiderate avviare, da 1 a 5, o il termine **single**.

Se utilizzate GRUB come boot loader, attenetevi alla seguente procedura:

- Nella schermata grafica del boot loader GRUB selezionate l'etichetta di avvio **Red Hat Linux** e premete [e] per modificarla.
- Utilizzate la freccia giù per passare alla linea del kernel e premete [e] per modificarla.
- Al prompt digitate il numero del runlevel che desiderate avviare, da 1 a 5, o il termine **single** e premete [Invio].
- Tornerete alla schermata di GRUB con le informazioni del kernel. Premete il tasto [b] per avviare il sistema.

Per ulteriori informazioni sui runlevel, consultate la Sezione 3.6.

4.11. Risorse aggiuntive

Questo capitolo è stato concepito solo come un'introduzione a GRUB e LILO. Per saperne di più sul funzionamento di GRUB e LILO, consultate le risorse descritte qui di seguito.

4.11.1. Documentazione installata

- `/usr/share/doc/grub-0.90` — questa directory contiene informazioni utili relative all'utilizzo e alla configurazione di GRUB.
- La pagina info di GRUB, accessibile dopo aver digitato il comando `info grub` al prompt della shell, contiene un'esercitazione, un manuale di riferimento per l'utente, un manuale di riferimento per il programmatore e un documento con le domande e le risposte più frequenti.
- `/usr/share/doc/lilo-21.4.4` — questa directory contiene numerose informazioni sull'utilizzo e la configurazione di LILO. In particolare la sottodirectory `doc` contiene un file postscript denominato `User_Guide.ps` molto utile.

4.11.2. Siti Web utili

- <http://www.gnu.org/software/grub> — è la home page del progetto GNU GRUB. Questo sito contiene le informazioni sull'andamento dello sviluppo di GNU e una serie di FAQ.
- <http://www.uruk.org/orig-grub> — in questo sito è contenuta tutta la documentazione originale di GRUB prima che il progetto venisse ceduto alla Free Software Foundation per ulteriori sviluppi.
- <http://www.redhat.com/mirrors/LDP/HOWTO/mini/Multiboot-with-GRUB.html> — queste pagine Web indagano sui diversi usi di GRUB, tra cui la possibilità di avviare sistemi operativi diversi da Linux.
- <http://www.linuxgazette.com/issue64/kohli.html> — questa pagina Web contiene un articolo introduttivo alla configurazione di GRUB sul vostro sistema e comprende inoltre una panoramica delle opzioni della linea di comando.
- <http://www.tldp.org/HOWTO/mini/LILO.html> — questo mini-HOWTO esamina i vari impieghi di LILO, incluso l'avvio di sistemi operativi diversi da Linux.

Utenti e gruppi

Il controllo degli *utenti* e dei *gruppi* è un elemento principale dell'amministrazione di sistema di Red Hat Linux.

Gli *utenti* possono essere persone, cioè account riuniti in utenti fisici, o utenti logici, vale a dire account che esistono per applicazioni specifiche. Entrambi i tipi di utenti dispongono di un *ID utente (UID)* e di un *ID di gruppo (GID)* univoci.

I *gruppi* sono espressioni logiche dell'organizzazione e uniscono gli utenti fornendo loro i permessi per leggere, scrivere o eseguire i file.

A un file creato viene assegnato un utente e un gruppo, oltre a permessi separati di lettura, scrittura ed esecuzione per il proprietario del file, il gruppo e chiunque altro. L'utente e il gruppo di un file particolare, oltre ai permessi di accesso al file, possono essere modificati da utenti root o, nella maggior parte dei casi, dall'autore del file.

La gestione appropriata degli utenti e dei gruppi e la gestione efficace dei permessi per i file sono tra le attività più importanti che un amministratore di sistema deve svolgere.

5.1. Strumenti per la creazione di utenti e gruppi

La gestione degli utenti e dei gruppi può rivelarsi un compito noioso, ma Red Hat Linux fornisce alcuni strumenti e convenzioni per semplificare la gestione da parte degli amministratori.

Potete utilizzare strumenti come `useradd` o `groupadd` per creare nuovi utenti e gruppi dal prompt della shell, ma un modo più semplice di gestire utenti e gruppi è quello di utilizzare l'applicazione grafica **redhat-config-users**. Per ulteriori informazioni su **redhat-config-users**, consultate la *Official Red Hat Linux Customization Guide*.

In la Sezione 5.4 è riportata un'ulteriore analisi dei concetti relativi ai permessi per i file e delle istruzioni della linea di comando per la gestione degli utenti.

5.2. Utenti standard

Nella Tabella 5-1 troverete gli utenti standard configurati dal processo di installazione nel file `/etc/passwd`. L'ID di gruppo (GID) contenuto in questa tabella rappresenta il *gruppo primario* dell'utente. Per un elenco di gruppi standard, consultate la Sezione 5.3.

Utente	UID	GID	Directory home	Shell
root	0	0	/root	/bin/bash
bin	1	1	/bin	/sbin/nologin
daemon	2	2	/sbin	/sbin/nologin
adm	3	4	/var/adm	/sbin/nologin
lp	4	7	/var/spool/lpd	/sbin/nologin
sync	5	0	/sbin	/bin/sync
shutdown	6	0	/sbin	/sbin/shutdown

Utente	UID	GID	Directory home	Shell
halt	7	0	/sbin	/sbin/halt
mail	8	12	/var/spool/mail	/sbin/nologin
news	9	13	/var/spool/news	
uucp	10	14	/var/spool/uucp	/sbin/nologin
operator	11	0	/root	/sbin/nologin
games	12	100	/usr/games	/sbin/nologin
gopher	13	30	/usr/lib/gopher-data	/sbin/nologin
ftp	14	50	/var/ftp	/sbin/nologin
nobody	99	99	/	/sbin/nologin
vcsa	69	69	/dev	/sbin/nologin
rpm	37	37	/var/lib/rpm	/bin/bash
wnn	49	49	/var/lib/wnn	/bin/bash
ntp	38	38	/etc/ntp	/sbin/nologin
ntp	38	38	/etc/ntp	/sbin/nologin
nscd	28	28	/	/bin/false
apache	48	48	/var/www	/bin/false
mysql	27	27	/var/lib/mysql	/bin/bash
mailnull	47	47	/var/spool/mqueue	/sbin/nologin
smmsp	51	51	/var/spool/mqueue	/sbin/nologin
rpc	32	32	/	/sbin/nologin
xf	43	43	/etc/X11/fs	/sbin/nologin
gdm	42	42	/var/gdm	/sbin/nologin
rpcuser	29	29	/var/lib/nfs	/sbin/nologin
nsfnobody	65534	65534	/var/lib/nfs	/sbin/nologin
ident	98	98	/	/sbin/nologin
radvd	75	75	/	/bin/false
sshd	74	74	/var/empty/sshd	/sbin/nologin
postgres	26	26	/var/lib/pgsql	/bin/bash
squid	23	23	/var/spool/squid	/dev/null
named	25	25	/var/named	/bin/false
pcap	77	77	/var/arpwatch	/sbin/nologin
amanda	33	6	var/lib/amanda/	/bin/bash
mailman	41	41	/var/mailman	/bin/false
netdump	34	34	/var/crash	/bin/bash

Utente	UID	GID	Directory home	Shell
ldap	55	55	/var/lib/ldap	/bin/false
postfix	89	89	/var/spool/postfix	/bin/true
privoxy	100	101	/etc/privoxy	
pvm	24	24	/usr/share/pvm3	/bin/bash

Tabella 5-1. Utenti standard

5.3. Gruppi standard

Nella Tabella 5-2 troverete i gruppi standard configurati dal programma di installazione. I gruppi sono archiviati in Red Hat Linux nel file /etc/group.

Gruppo	GID	Componenti
root	0	root
bin	1	root, bin, daemon
daemon	2	root, bin, daemon
sys	3	root, bin, adm
adm	4	root, adm, daemon
tty	5	
disk	6	root
lp	7	daemon, lp
mem	8	
kmem	9	
wheel	10	root
mail	12	mail
news	13	news
uucp	14	uucp
man	15	
games	20	
gopher	30	
dip	40	
ftp	50	
lock	54	
nobody	99	
users	100	
slocate	21	

Gruppo	GID	Componenti
floppy	19	
vcsa	69	
rpm	37	rpm
utmp	22	
wnn	49	
ntp	38	
nscd	28	
apache	48	
mysql	27	
mailnull	47	
smmsp	51	
rpc	32	
xf	43	
gdm	42	
rpcuser	29	
nfsnobody	65534	
ident	98	
radvd	75	
sshd	74	
postgres	26	
squid	23	
named	25	
pcap	77	
wine	66	
mailman	41	
netdump	34	
ldap	55	
postdrop	90	
postfix	89	
privoxy	101	
pvm	24	

Tabella 5-2. Gruppi standard

5.4. Gruppi utente privati

Red Hat Linux utilizza uno schema di *gruppo utente privato* (UPG, User Private Group), che semplifica l'utilizzo dei gruppi UNIX. Lo schema UPG non aggiunge o modifica nulla nella gestione standard dei gruppi da parte di UNIX, ma offre semplicemente una nuova convenzione. Ogni nuovo utente dispone, per default, di un gruppo univoco. Lo schema funziona come indicato di seguito:

Gruppo utente privato

Ogni utente dispone di un gruppo principale. L'utente è l'unico componente di tale gruppo.

`umask = 002`

In genere nei sistemi UNIX `umask` è 022 e impedisce ad altri utenti e ad altri componenti di un gruppo primario di modificare i file dell'utente. Dato che ciascun utente dispone del proprio gruppo privato nello schema UPG, questa "protezione dei gruppi" non è necessaria. Una `umask` pari a 002 impedirà agli utenti di modificare i file privati di altri utenti. La `umask` è impostata nel file `/etc/profile`.

setgid bit nelle directory

Se impostate `setgid` bit in una directory (mediante il comando `chmod g+s directory`), il gruppo dei file creati in tale directory sarà impostato come gruppo della directory.

Molte organizzazioni IT creano un gruppo per ciascun progetto principale e vi assegnano persone se è necessario che accedano ai file del gruppo. Con questo schema tradizionale, la gestione dei file si è rivelata complessa, perché un file creato viene associato al gruppo primario a cui appartiene l'utente. Quando un singolo utente lavora su più progetti, è difficile associare i file corretti con il gruppo appropriato. Con lo schema UPG, invece, i gruppi vengono assegnati automaticamente ai file creati all'interno di una directory con il `setgid` bit impostato, operazione che semplifica notevolmente la gestione dei progetti di gruppo che condividono una directory comune.

Supponete, per esempio, di disporre di un progetto importante denominato `devel`, a cui partecipano molti utenti che modificano i file `devel` in una directory `devel`. Create un gruppo denominato `devel`, mediante `chgrp` aggiungete la directory `devel` e tutti gli utenti `devel` al gruppo `devel`.

Potete aggiungere un utente a un gruppo mediante **redhat-config-users** (consultate la *Official Red Hat Linux Customization Guide*) oppure, se preferite utilizzare la linea di comando, digitate il comando `/usr/sbin/groupadd nomegruppo` per creare un gruppo. Il comando `/usr/bin/gpasswd -a nomeidlogin nomegruppo` consentirà di aggiungere il `nomeidlogin` dell'utente a un gruppo. (Per ulteriori informazioni sulle relative opzioni, consultate le pagine `man` di `groupadd` e `gpasswd`.) Il file `/etc/group` contiene le informazioni sui gruppi per il vostro sistema.

Se avete creato il gruppo `devel`, aggiunto gli utenti al gruppo `devel`, modificato il gruppo per la directory `devel` del gruppo `devel` e impostato il `setgid` bit per la directory `devel`, tutti gli utenti `devel` saranno in grado di modificare i file `devel` e di creare nuovi file nella directory `devel`. I file creati manterranno sempre il proprio stato di gruppo `devel`, perché altri utenti `devel` possano sempre essere in grado di modificarli.

Se sono in corso numerosi progetti come `devel` a cui partecipano molti utenti, tali utenti non dovranno mai modificare la `umask` o il gruppo quando passano da un progetto all'altro. Se impostato correttamente, il `setgid` bit della directory principale di ciascun progetto "seleziona" il gruppo appropriato per tutti i file creati in tale directory.

Dato che la directory `home` di ogni utente appartiene all'utente e al proprio gruppo privato, è più sicuro impostare il `setgid` bit nella `home` directory `home`. Tuttavia, per default, i file vengono creati con il gruppo primario dell'utente, pertanto il `setgid` risulterebbe ridondante.

5.4.1. Logica dei gruppi utente privati

Anche se l'UPG (User Private Group) è già in uso in Red Hat Linux da tempo, molte persone hanno ancora dei dubbi, per esempio sulla motivazione per cui è necessario questo schema. Per spiegarne l'utilizzo, considerate lo scenario riportato di seguito.

Stabilite di fare in modo che un gruppo di persone lavori a una serie di file nella directory `/usr/lib/emacs/site-lisp/`. Per la modifica della directory, avete fiducia solo in alcune persone, ma certamente non in tutte. Quindi create innanzi tutto un gruppo `emacs`:

```
/usr/sbin/groupadd emacs
```

Per associare il contenuto della directory al gruppo `emacs`, digitate:

```
chown -R root.emacs /usr/lib/emacs/site-lisp
```

È ora possibile aggiungere gli utenti appropriati al gruppo con il comando `gpasswd`:

```
/usr/bin/gpasswd -a <nomeutente> emacs
```

Consentite agli utenti di creare realmente i file nella directory con il comando riportato di seguito:

```
chmod 775 /usr/lib/emacs/site-lisp
```

Un nuovo file creato viene assegnato al gruppo privato di default dell'utente. Per evitare che questo accada, digitate il comando riportato di seguito che fa sì che tutto nella directory venga creato nel gruppo `emacs`:

```
chmod 2775 /usr/lib/emacs/site-lisp
```

Se il nuovo file deve essere in modalità 664 perché un altro utente possa modificarlo all'interno del gruppo `emacs`, utilizzate `umask 002` come valore predefinito.

A questo punto, l'impostazione di `umask` a 002 come valore predefinito consente di impostare facilmente i gruppi da cui gli utenti possano trarre vantaggio senza ulteriori sforzi ogni volta che vengono scritti i file nella directory comune del gruppo. Quindi create semplicemente il gruppo, aggiungetevi degli utenti ed eseguite i comandi `chown` e `chmod` riportati in precedenza in relazione alle directory del gruppo.

5.5. Utility shadow

Se siete in un ambiente multiutente e non utilizzate uno schema di autenticazione di rete come Kerberos, dovrete considerare l'utilizzo di utility shadow, anche note come *password shadow*, per la protezione avanzata offerta ai file di autenticazione del vostro sistema. Durante l'installazione di Red Hat Linux la protezione tramite password shadow per il vostro sistema è abilitata di default, così come le *password MD5*, un metodo alternativo e discutibilmente più sicuro di cifratura per l'archiviazione delle password nel sistema.

Le password shadow offrono alcuni vantaggi evidenti rispetto allo standard precedente di archiviazione delle password nei sistemi UNIX e Linux, tra i quali:

- Sicurezza del sistema migliorata spostando le password cifrate, in genere contenute in `/etc/passwd`, in `/etc/shadow`, un file leggibile solo da utenti `root`.
- Informazioni relative alla datazione delle password, vale a dire quanto tempo è trascorso dall'ultima modifica della password.

- Controllo sulla durata della password, vale a dire quanto tempo può rimanere invariata prima che venga richiesto all'utente di cambiarla.
- Capacità di utilizzare il file `/etc/login.defs` per rafforzare un criterio di sicurezza, relativo in particolare alla datazione della password.

Il pacchetto `shadow-utils` contiene numerose utility che supportano:

- La conversione da password normali a shadow e viceversa (`pwconv`, `pwunconv`).
- La verifica delle password, del gruppo e dei file shadow associati (`pwck`, `grpck`).
- I metodi standard per l'aggiunta, l'eliminazione e la modifica degli account utente (`useradd`, `usermod` e `userdel`).
- I metodi standard per l'aggiunta, l'eliminazione e la modifica dei gruppi di utenti (`groupadd`, `groupmod` e `groupdel`).
- Il metodo standard per l'amministrazione del file `/etc/group` mediante il comando `gpasswd`.

Queste utility dispongono di altri punti di interesse, quali:

- Le utility funzionano in modo corretto indipendentemente dallo stato di attivazione/disattivazione dello shadowing.
- Le utility sono state leggermente modificate per supportare lo schema di gruppi utente privati di Red Hat Linux. Per una descrizione delle modifiche, consultate la pagina `man useradd`. Per ulteriori informazioni sui gruppi utente privati, consultate la Sezione 5.4.
- Lo script `adduser` è stato sostituito con un link simbolico a `/usr/sbin/useradd`.
- Gli strumenti contenuti nel pacchetto `shadow-utils` non sono compatibili con Kerberos, NIS, `hesiod` o LDAP. I nuovi utenti saranno solo locali. Per ulteriori informazioni su Kerberos e LDAP, consultate il Capitolo 10 e il Capitolo 18.

Il sistema X Window

Sebbene il kernel sia il fulcro di Red Hat Linux, per molti utenti, l'interlocutore del sistema operativo è l'ambiente grafico fornito dal *Sistema X Windows*, più semplicemente noto come *X*.

Questo capitolo è un'introduzione per quanto concerne il dietro le quinte del mondo *XFree86*, l'implementazione Open Source di *X* fornita con Red Hat Linux.

6.1. La potenza di X

Linux ha cominciato come un potente sistema operativo basato su server, particolarmente efficiente nell'elaborazione di programmi complicati che necessitavano di un ampio utilizzo della CPU e la gestione di centinaia, se non migliaia, di client durante la connessione di rete. Successivamente, grazie alla sua natura Open Source e alla sua stabilità, Linux si è rapidamente trasformato in un sistema operativo per workstation, basato su GUI, molto diffuso sia in ambito domestico che lavorativo.

UNIX™, che può vantare la presenza degli ambienti a finestra da decenni, ha anticipato molti dei principali sistemi operativi esistenti. Il sistema X Windows è ormai l'interfaccia utente grafica predominante per sistemi operativi UNIX e simili.

Per creare quest'interfaccia grafica utente, *X* utilizza un'architettura client-server. L'avvio di un processo *X server* consente ai processi *X client* di collegarsi tramite interfaccia loopback di rete o locale. Il server gestisce la comunicazione con l'hardware, in termini di scheda video, monitor, tastiera e mouse. Il client *X* agisce nello spazio utente inviando richieste al server *X*.

Sui sistemi Red Hat Linux il server *XFree86* svolge il medesimo ruolo del server *X*. Poiché si tratta di un progetto di software open source con un vasto campo d'azione e che coinvolge centinaia di sviluppatori in tutto il mondo, *XFree86* consente un rapido sviluppo e garantisce un ampio ventaglio di opzioni per l'assistenza relativa a diversi dispositivi e architetture hardware, oltre alla possibilità di esecuzione su differenti piattaforme e sistemi operativi.

La maggior parte degli utenti Red Hat Linux è solitamente più interessata al particolare ambiente desktop in cui si trova a lavorare e non è consapevole del fatto che il proprio sistema utilizza il server *XFree86*. Il programma d'installazione di Red Hat Linux è perfetto per la configurazione del server *XFree86* durante il processo d'installazione poiché assicura il funzionamento ottimale di *X* fin dalla prima volta in cui lo lancerete.

Il server *X* è in grado di eseguire molti compiti difficili utilizzando un'ampia gamma di hardware che richiedono una configurazione dettagliata. Se vengono modificati alcuni aspetti del sistema, come ad esempio la scheda video o del monitor, *XFree86* dovrà essere nuovamente configurato. Inoltre, in caso avete un problema con *XFree86* che non è risolvibile usando una utility di configurazione, quale **X Configuration Tool** (`redhat-config-xfree86`), potrebbe essere necessario accedere direttamente ai suoi file di configurazione.



Suggerimento

X Configuration Tool è in grado di configurare `xconfigurator` mentre il server *X* è attivo. Per attivare il programma di configurazione dalla linea di comando, digitate `redhat-config-xfree86`. Per avviare **X Configuration Tool** mentre *X* non è attivo, fate clic sul pulsante **Menu principale** => **Sistema** => **Display**. Dopo aver utilizzato **X Configuration Tool** durante una sessione di *X*, per rendere effettive le modifiche dovete scollegarvi dalla sessione attuale e ricollegarvi.

6.2. XFree86

Red Hat Linux utilizza XFree86 4 come base del sistema X Window, il quale include le diverse librerie, i font, le utility, la documentazione e i tool di sviluppo X necessari.



Nota Bene

Red Hat non fornisce più i pacchetti per la versione 3 del server XFree86. Prima di eseguire l'aggiornamento per passare all'ultima versione di Red Hat Linux, assicuratevi che la vostra scheda video sia compatibile con la versione 4 di XFree86 controllando l'elenco di compatibilità hardware di Red Hat Linux all'indirizzo <http://hardware.redhat.com/hcl/>.

Il server X include numerose ottimizzazioni tecnologiche XFree86 all'avanguardia, fra cui il supporto per l'acceleramento 3D dell'hardware, l'estensione XRender per i font anti-alias, un design basato su driver modulare, il supporto per dispositivi video hardware e di input moderni e numerose altre caratteristiche.

Il programma di installazione di Red Hat Linux installerà i componenti base di XFree86 ed eventuali pacchetti XFree86 aggiuntivi di vostra scelta.

Il sistema X Windows si trova prevalentemente in due diverse posizioni nel filesystem:

directory `/usr/X11R6`

Una directory contenente binari client X (la directory `bin`), file header assortiti (la directory `include`), librerie (la directory `lib`), pagine man (la directory `man`) e altro materiale di documentazione X (la directory `/usr/X11R6/lib/X11/doc/`).

directory `/etc/X11`

La gerarchia della directory `/etc/X11` contiene tutti i file di configurazione per i vari componenti che costituiscono il sistema X Window, inclusi i file di configurazione per lo stesso server X, per il server font (`xf86`), il Display Manager X (`xdm`) e molti altri componenti di base. Display manager quali `gdm` e `kdm`, come altri Window Manager e tool X, archiviano la propria configurazione in questa gerarchia.

La versione 4 del server XFree86 è un binario singolo eseguibile (`/usr/X11R6/bin/XFree86`). Al runtime è in grado di caricare in modo dinamico dinamicamente vari moduli del server X dalla directory `/usr/X11R6/lib/modules`, inclusi driver video, driver motore di font e altri moduli speciali eventualmente necessari. Alcuni di questi moduli sono caricati automaticamente dal server, altri sono invece caratteristiche opzionali che dovrete specificare nel file `/etc/X11/XF86Config-4`, il file di configurazione del server XFree86 4, affinché questi possano essere utilizzati. I driver video per XFree86 4 sono posizionati nella directory `/etc/X11R6/lib/modules/drivers`. I driver 3D accelerati dell'hardware DRI si trovano in `/etc/X11R6/lib/modules/dri`.

6.2.1. File di configurazione del server XFree86

I file di configurazione del server XFree86 sono archiviati nella directory `/etc/X11`. Il server XFree86 vers. 4 utilizza `/etc/X11/XF86Config`. Una volta installato Red Hat Linux, i file di configurazione per XFree86 vengono creati mediante informazioni raccolte durante il processo di installazione.

Raramente vi capiterà di dover modificare manualmente questi file, è tuttavia utile conoscere le varie sezioni e i parametri opzionali in esse presenti.

Ogni sezione inizia con una riga `Section` "`<nome-sezione>`" e termina con la riga `EndSection`. All'interno di ciascuna sezione troverete diverse righe contenenti un nome opzione e almeno un valore

d'opzione, a volte fra virgolette. Di seguito è riportato un elenco delle sezioni più utili di un file della versione XFree86 4 e le funzioni di varie impostazioni comuni.

Device

Specifica le informazioni sulla scheda video utilizzata dal sistema. Il file di configurazione deve avere almeno una sezione `Device`. In caso di schede video multiple o impostazioni multiple che eseguono una singola scheda, dovete avere sezioni `Device` multiple. Le seguenti opzioni sono richieste o ampiamente utilizzate:

- `BusID` — Specifica la posizione di bus della scheda video. Quest'opzione è necessaria solo per i sistemi con schede multiple e deve essere impostata in modo tale che la sezione `Device` utilizzi le impostazioni appropriate per la scheda corretta.
- `Driver` — Dice a XFree86 quale driver caricare per utilizzare il dispositivo relativo alla scheda video.
- `Identifier` — Fornisce un nome unico per la scheda video in questione. In genere, questo nome è impostato sul nome esatto della scheda video utilizzata in questa sezione `Device`.
- `Screen` — Impostazione opzionale utilizzata quando la scheda video ha più di una *testa*, oppure un connettore in uscita per un monitor separato. Se disponete di più monitor collegati a una scheda video, esisteranno sezioni `Device` separate per ciascuno di questi, cui sarà attribuito un valore `Screen` specifico per ogni sezione `Device`. Quest'opzione utilizza valori numerici: il minimo è 0 e può essere incrementato di uno per ogni testa della scheda video.
- `VideoRam` — La capacità della RAM disponibile sulla scheda video, in kilobyte. In genere, quest'impostazione non è richiesta, poiché il server XFree86 è in grado di richiedere alla scheda video di determinare automaticamente la quantità di RAM video. Tuttavia, vi sono alcuni hardware che non consentono a XFree86 di eseguire quest'operazione in modo corretto. Questa opzione vi permette, quindi, di specificare manualmente l'esatto ammontare della RAM video.

DRI

La *Direct Rendering Infrastructure (DRI)* è un'interfaccia che consente alle applicazioni software 3D di trarre il massimo vantaggio dalle capacità di accelerazione 3D dell'hardware sui moderni hardware video supportati. Inoltre, la DRI può migliorare le prestazioni di accelerazione hardware 2D quando si utilizzano driver ottimizzati per l'uso della DRI durante operazioni 2D. Questa sezione viene ignorata, a meno che la DRI viene attivata nella sezione `Module`.

Il modo di utilizzo della DRI varia a seconda del tipo di scheda video. Prima di modificare i valori della DRI, leggete il file `/usr/X11R6/lib/X11/doc/README.DRI` per informazioni specifiche sulle singole schede video.

Files

Questa sezione imposta i percorsi per servizi particolari necessari al server, come il percorso per i font. Le opzioni più comuni includono:

- `FontPath` — Definisce le posizioni in cui il server XFree86 può trovare i font. Qui possono essere posizionati i file font, separati da virgole. Di default, Red Hat Linux utilizza `xf86` come server font e orienta `FontPath` verso `unix/:7100`. Istruisce il server XFree86 per ottenere informazioni sui font usando le socket del dominio UNIX per l'IPC (Inter Process Communication).

Consultate la Sezione 6.5 per maggiori informazioni su XFree86 e i font.

- `ModulePath` — Vi consente di impostare directory multiple per l'archiviazione dei moduli caricati dal server XFree86.

- `RgbPath` — Informa il server XFree86 sulla posizione del database di colori RGB. Questo file database definisce i nomi validi dei colori in XFree86 e li collega a specifici valori RGB.

InputDevice

Configura i dispositivi in ingresso, fra cui mouse e tastiera, usati per introdurre informazioni nel sistema mediante il server XFree. La maggior parte dei sistemi ha almeno due sezioni `InputDevice`, per tastiera e mouse. Ogni sezione include le due linee seguenti:

- `Driver` — Comunica a XFree86 il nome del driver da caricare per usare il dispositivo.
- `Identifier` — Imposta il nome del dispositivo, in genere il nome dello stesso dispositivo seguito da un numero. Lo 0 corrisponde al primo dispositivo. Per esempio, la prima tastiera `InputDevice` avrà un `Identifier` di questo tipo: "Keyboard0".

La maggior parte delle sezioni `InputDevice` contiene delle righe che assegnano opzioni specifiche del dispositivo in questione. Ciascuna di queste inizia con `Option` e contiene il nome dell'opzione fra virgolette, seguita dal valore da assegnarle. I mouse presentano, in genere, opzioni come `Protocol`, `PS/2` e `Device`. Quest'ultima definisce il mouse da utilizzare per la sezione corrente. La sezione `InputDevice` è ben commentata e consente di configurare opzioni aggiuntive per dispositivi particolari semplicemente disabilitando alcune linee.

Module

Dice al server XFree86 quali moduli della directory `/etc/X11R6/lib/modules` devono essere caricati. I moduli forniscono funzionalità aggiuntive al server XFree86.



Attenzione

Si consiglia di non apportare variazioni a questi valori.

Monitor

Relativo al tipo di monitor utilizzato dal sistema. Potrebbero esservi diverse sezioni `Monitor`, una per ogni monitor in uso con l'elaboratore. Deve esserci almeno una sezione `Monitor`.



Avvertenza

Prestate particolare attenzione quando modificate manualmente i valori relativi alle opzioni della sezione `Monitor`. L'impostazione inappropriata dei valori di questa sezione potrebbe danneggiare, o rendere inutilizzabile, il monitor. Consultate la documentazione fornita unitamente all'apparecchio per la disponibilità dei parametri operativi sicuri.

Abitualmente si ha la configurazione delle seguenti opzioni durante il processo di installazione o quando si utilizza **X Configuration Tool**:

- `HorizSync` — Informa XFree86 sul range di frequenze di sincronizzazione orizzontale compatibili con il monitor in kHz. Tali valori sono utilizzati dal server come guida, in modo tale che XFree86 sappia se usare valori d'inserimento di una `Modeline` specifica per il monitor in questione.
- `Identifier` — L'identificatore dispone di un nome unico per il monitor considerato, in genere un valore numerico a partire da 0. Il primo monitor sarà dunque denominato `Monitor0`, il secondo `Monitor1`, e così via.
- `Modeline` — Viene utilizzato per specificare le modalità video usate dal monitor a particolari risoluzioni, con specifica sincronizzazione orizzontale e risoluzioni refresh verticali. Le voci di `Modeline` sono in genere precedute da un commento che spiega quanto specificato dalla `mode line`.

Se il vostro file di configurazione non include i commenti per le varie mode line, potete ricercare i valori (o le *descrizioni mode*) per scoprire l'operazione che la mode line sta cercando di eseguire. Consultate la pagina `man XF86Config` per spiegazioni esaurienti su ciascuna sezione di descrizione mode.

- `ModelName` — Parametro opzionale che visualizza il modello del monitor.
- `VendorName` — Parametro opzionale che visualizza il venditore che si è occupato della produzione del monitor.
- `VertRefresh` — Elenca, in kHz, le frequenze di range relative al refresh verticale supportate dal monitor. Tali valori sono utilizzati dal server come guida, in modo tale che XFree86 sappia se usare valori d'inserimento di una `Modeline` specifica per il monitor in questione.

Screen

Abbina un particolare `Device` e `Monitor` utilizzabili come copia e contenenti specifiche impostazioni. È necessario avere almeno una sezione `Screen` nel file di configurazione. Le seguenti sono opzioni comuni:

- `DefaultDepth` — Dice alla sezione `Screen` quale intensità di default del colore da provare, in bit. 8 è il valore di default, 16 mette a disposizione migliaia di colori mentre 32 è in grado di visualizzarne milioni.
- `Device` — Indica il nome della sezione `Device` da utilizzare con la sezione `Screen` considerata.
- `Identifier` — Identifica la sezione `Screen`, in modo che si possa fare riferimento ad essa mediante una sezione `ServerLayout` e usarla.
- `Monitor` — Comunica il nome della sezione `Monitor` da utilizzare con la sezione `Screen` considerata.

È anche possibile disporre di una sotto-sezione `Display` che istruisce il server XFree86 sull'intensità di colore (`Depth`) e la risoluzione (`Mode`) da provare per prima quando si usano un monitor e una scheda video particolari.

ServerFlags

Contiene varie impostazioni globali del server XFree86. Tali impostazioni possono essere escluse mediante opzioni posizionate nella sezione `ServerLayout`. Fra le impostazioni più utili:

- `DontZap` — Impedisce l'uso della combinazione dei tasti `[Ctrl]-[Alt]-[Backspace]` per chiudere in modo immediato il server XFree86.
- `DontZoom` — Impedisce l'uso di `[Ctrl]-[Alt]-[Keypad-Plus]` e `[Ctrl]-[Alt]-[Keypad-Minus]` con le risoluzioni video configurate.

ServerLayout

Combina una sezione `Screen` con le sezioni `InputDevice` necessarie e varie opzioni per creare un insieme unificato di preferenze usate dal server XFree96 all'avvio. Se disponete di più sezioni `ServerLayout` e quella da utilizzare non è specificata sulla linea di comando quando si richiama il server XFree86, viene di norma la sezione `ServerLayout` che compare per prima nel file di configurazione.

In una sezione `ServerLayout` vengono usate le seguenti opzioni:

- `Identifier` — Il nome univoco usato per descrivere quella sezione `ServerLayout`.
- `InputDevice` — I nomi di ogni sezione `InputDevice` da usare con il server XFree86. In quest'opzione, la maggior parte degli utenti avrà solo due linee, `Keyboard0` e `Mouse0`, che

indicano la prima tastiera e il primo mouse configurati per il sistema. Le opzioni `CoreKeyboard` e `CorePointer` si riferiscono al fatto che questi sono la tastiera e il mouse di default per l'uso con il server `XFree86`.

- `Screen` — Il nome della sezione `Screen` da utilizzare. Il numero alla sinistra del nome della sezione `Screen` fa riferimento al numero di schermo specifico da usare in una configurazione multi-head. Per schede video con un'unica uscita, questo valore è 0. I numeri a destra forniscono le coordinate assolute X e Y per l'angolo superiore sinistro dello schermo, di default 0 0.

Ecco un esempio di schermata di `Screen`:

```
Screen      0  "Screen0"  0  0
```

Per ulteriori informazioni, fate riferimento alla pagina `man XF86Config`.

Per rivedere la configurazione attuale del vostro server `XFree86`, digitate il comando `xset -q`. Ciò vi fornirà informazioni sulla tastiera, il mouse, lo screen saver e i percorsi font.

6.3. Ambienti desktop e Window Manager

La configurazione di un server `XFree86` risulta inutile fino all'accesso per opera di un client X. Questo utilizzerà il server per visualizzare un programma con l'hardware controllato dal server X. I client X sono programmi ideati per trarre vantaggio dall'uso dell'hardware del server X, solitamente per garantire l'interattività con l'utente.

Per utilizzare le applicazioni client, non è necessario lanciare un Window Manager complesso in congiunzione con un particolare ambiente desktop. Supponendo che non siate già in un ambiente X e che la vostra home directory non contenga un file `.xinitrc`, digitate il comando `xinit` per avviare X con una finestra terminale di base (l'applicazione predefinita `xterm`). Vedrete che questo ambiente di base usa tastiera, mouse, scheda video e monitor con il server `XFree86` mediante le preferenze hardware del server. Digitate `exit` al prompt `xterm` per lasciare l'ambiente X di base.

Naturalmente, la maggior parte degli utenti richiede più caratteristiche e utility dal proprio GUI. Gli sviluppatori hanno aggiunto layer di funzioni per creare ambienti interattivi con un alto livello di sviluppo e che sfruttano appieno la potenza del server `XFree86`. Questi layer sono divisi in due gruppi fondamentali creati per questo scopo specifico.

6.3.1. Window Manager

I *Window Manager* sono programmi di client X che controllano il modo in cui vengono posizionati, ridimensionati o spostati gli altri client X. I Window Manager possono disporre anche di barre dei titoli per finestre, anteprima tastiera mediante tastiera o mouse, corrispondenze tasti e pulsanti mouse specificate dall'utente. I Window Manager operano con un insieme di client X differenti, proteggono il programma e gli conferiscono un aspetto particolare e una posizione sullo schermo.

Red Hat Linux 8.0 comprende quattro manager:

- `twm` — Il *Tab Window Manager* minimalista, che fornisce il set di tool base per i Window Manager.
- `mwm` — il window manager di default per l'ambiente desktop GNOME: `mwm` sta per Metacity Window Manager. Si tratta di un window manager efficiente che supporta elementi personalizzati.
- `sawfish` — questo window manager completo era quello predefinito fino alla versione 8.0 di Red Hat Linux. Può essere utilizzato con o senza l'ambiente desktop GNOME.
- `wmaker` — *WindowMaker* è un window manager completo di GNU progettato per emulare l'aspetto dell'ambiente NEXTSTEP.

Questi Window Manager possono essere lanciati come client X individuali per ottenere una migliore visione delle differenze presenti. Digitare il comando `xinit <percorso-verso-il-Window-Manager>`, dove `<percorso-verso-il-Window-Manager>` è la posizione del file binario del Window Manager. Questo file può essere individuato digitando `which <nome-Window-Manager>` oppure cercando il nome del Window Manager in una directory `bin`.

6.3.2. Ambienti desktop

Un *ambiente desktop* unisce diversi client X che possono essere lanciati insieme usando metodi simili, utilizzando un ambiente di sviluppo comune.

Gli ambienti desktop differiscono dai Window Manager, che controllano unicamente l'aspetto e la posizione delle finestre dei client X. Gli ambienti desktop contengono funzioni avanzate che consentono ai client X e ad altri processi correnti di comunicare fra loro. In questo modo, tutte le applicazioni scritte per lavorare in quell'ambiente possono integrarsi e avere nuovi usi, fra cui la possibilità di utilizzare la tecnica di trascinamento e rilascio (`drag-and-drop`) del testo.

GNOME è l'ambiente desktop di default di Red Hat Linux che utilizza il toolkit widget di base GTK+ e altri widget misti che estendono la funzionalità di base. KDE, un altro ambiente desktop, usa un diverso toolkit chiamato Qt. GNOME e KDE comprendono entrambi applicazioni di produttività avanzate, quali word processor, spreadsheet e dispositivi del pannello di controllo per avere il controllo completo sull'esperienza come utenti. Le applicazioni dei client X standard possono essere lanciate da entrambi gli ambienti e, se le librerie Qt sono installate, alcune applicazioni KDE sono eseguibili in GNOME.

Quando avviate X con il comando `startx`, viene utilizzato l'ambiente desktop specificato in precedenza. Per modificare l'ambiente di default usato all'avvio di X, aprite un terminale e digitate il comando `switchdesk`. Questo comando richiama una utility grafica che permette di selezionare l'ambiente desktop o il Window Manager da usare la prossima volta che viene avviato X.

Gli ambienti desktop usano i Window Manager per fornire un aspetto consistente tra applicazioni differenti. KDE contiene il proprio Window Manager, `kwm`, specificatamente per quella funzionalità.

Per informazioni sulla personalizzazione degli ambienti desktop GNOME e KDE, consultate la *Official Red Hat Linux Getting Started Guide*.

6.4. Runlevel

La maggior parte degli utenti avvia X dal runlevel 3 o 5. Il runlevel 3 imposta il sistema in modalità multi-utente con piena capacità di networking. L'elaboratore si avvierà con un prompt di login basato su testo con tutti i servizi preconfigurati già avviati. Quasi tutti i server vengono avviati nel runlevel 3 poiché, per fornire i servizi usati dalla maggior parte degli utenti, non viene richiesto X. Il runlevel 5 è simile al 3, fatta eccezione per il fatto che avvia automaticamente X e fornisce uno schermo di login grafico. Molti utenti di workstation preferiscono questo metodo perché non li costringe a visualizzare il prompt di comando.

Il runlevel di default usato all'avvio del sistema può essere trovato nel file `/etc/inittab` file. Se il file presenta una riga simile a `id:3:initdefault:`, il vostro sistema si apre al runlevel 3. Se invece la riga è `id:5:initdefault:`, il sistema è impostato per avviarsi al runlevel 5. Come root, modificate il numero di runlevel in questo file per impostare un runlevel di default differente. Per ulteriori informazioni sui runlevel, consultate la Sezione 3.6.

6.4.1. Runlevel 3: `startx`

Se vi trovate nel runlevel 3, è preferibile avviare una sessione X digitando il comando `startx.startx` è un front-end del programma `xinit`, che lancia il server XFree86 e connette a esso i client X.

Poiché dovete già aver eseguito il login nel sistema al runlevel 3 per poter digitare i comandi, `startx` è ideato unicamente per richiamare certi client X, come un ambiente desktop, in un modo particolare. Non dispone dell'autenticazione utente.

Quando il comando `startx` entra in esecuzione, cerca nella home directory un file `.xinitrc` definito dall'utente per individuare i client X da avviare. Se non trova il file specificato, avvia al suo posto `/etc/X11/xinit/xinitrc`, lo script di default del sistema. Lo script `startx` fa la stessa cosa con il file `.xserverrc`: lo cerca nella home directory dell'utente e, se non lo trova, avvia lo script di default `/etc/X11/xinit/xserverrc`. Data l'esistenza di molteplici client X, i file `xinitrc` sono molto importanti. Lo script `xserverrc` è invece meno rilevante, poiché imposta soltanto il server X per la connessione con i client X. Visto che il server X di default è già configurato con il link `/etc/X11/X`, Red Hat Linux non installa il `xserverrc` di default.

Lo script di default `xinitrc` cerca quindi i file definiti dall'utente e i file di sistema predefiniti, inclusi `.Xresources`, `.Xmodmap` e `.Xkbmap` nella home directory dell'utente, e i file `Xresources`, `Xmodmap` e `Xkbmap` nella directory `/etc/X11`. I file `Xmodmap` e `Xkbmap`, se presenti, sono usati dall'utility `xmodmap` per configurare la tastiera. Vengono letti i file `Xresources` per assegnare valori di default specifici ad applicazioni particolari.

Una volta impostate queste opzioni, lo script `xinitrc` esegue gli script presenti in `/etc/X11/xinit/xinitrc.d`. Fra gli script più importanti contenuti in questa directory vi è `xinput`, che configura le impostazioni, per esempio la lingua di default da usare e l'ambiente desktop da avviare (da `/etc/sysconfig/desktop`).

Successivamente, lo script `xinitrc` tenta di eseguire `.Xclients` presente nella home directory dell'utente, e si rivolge a `/etc/X11/init/Xclients` nel caso in cui il primo non fosse disponibile. Lo scopo del file `Xclients` è di avviare l'ambiente desktop o, se possibile, anche solo un Window Manager di base. Lo script `.Xclients` nella home directory dell'utente avvia l'ambiente desktop o il Window Manager specificato dall'utente nel file `.Xclients-default`. Se `.Xclients` non è disponibile nella home directory dell'utente, lo script standard `/etc/X11/init/Xclients` cerca di avviare un altro ambiente desktop, prima con GNOME e successivamente con KDE. Se a questo punto non è possibile trovare un ambiente desktop, `Xclients` passa in rassegna una lista di Window Manager per trovarne uno che sia possibile avviare, dopo aver provato con il Window Manager di default indicato nel file `.wm_style` contenuto nella home directory dell'utente. Se non riesce a portare a termine questa operazione, esegue la ricerca in un elenco predefinito di gestori di finestre.

A questo punto, le applicazioni client X preferite dovrebbero essere avviate, e con esse anche il server XFree86. Se necessitate di maggiori dettagli sull'avvio di X nel runlevel 3, fate riferimento alle pagine `man startx` e `xinit` e leggete attentamente gli script citati in precedenza.

6.4.2. Runlevel 5: `prefdm`

Il metodo d'avvio di X con il runlevel 5 è leggermente differente. Quando il sistema si avvia, di default nessuno è registrato nel sistema. Per iniziare una sessione, l'utente vi si deve registrare. Gli utenti che eseguono l'autenticazione presso la console nel runlevel 5, utilizzano un `display manager`, uno speciale client X che consente di sottoporre il proprio nome di login e la password.

A seconda degli ambienti desktop installati sul vostro sistema Red Hat Linux, sono disponibili tre diversi display manager per la gestione dell'autenticazione utente. Il display manager `xdm` è il tool d'autenticazione X originale. `xdm` vi consente unicamente di registrarvi e iniziare una sessione X. Il display manager `gdm`, ideato per operare con ambienti desktop GNOME, e `kdm`, in uso con l'ambiente desktop KDE, vi consentono di impostare l'ambiente desktop o la *sessione* che desiderate utilizzare dopo l'autenticazione. Inoltre, potete riavviare o arrestare il sistema dalla schermata di login. Il display manager `gdm` vi permette anche di configurare la lingua che desiderate usare.

Quando il sistema entra nel runlevel 5, una riga nel file `/etc/inittab` specifica che viene eseguito lo script `prefdm` al fine di determinare il display manager preferito per richiamare l'autenticazione utente. Lo script `prefdm` utilizza le preferenze indicate nel file `/etc/sysconfig/desktop` per trovare il display manager appropriato. Se non viene specificato alcun ambiente desktop, `prefdm` passa

in rassegna i display manager `gdm`, `kdm`, e `xdm` per trovarne uno da usare. Una volta fatto questo, `prefdm` lo lancia per la gestione del login utente.

I display manager fanno riferimento al file `/etc/X11/xdm/Xsetup_0` per impostare la schermata di registrazione. Effettuata la registrazione nel sistema, lo script `/etc/X11/xdm/GiveConsole` assegna all'utente la proprietà della console. Successivamente, lo script `/etc/X11/xdm/Xsession` entra in esecuzione per compiere molti dei compiti di cui si occupa in genere lo script `xinitrc` all'avvio di X nel runlevel 3, incluse le impostazioni del sistema e delle risorse utente, oltre all'esecuzione degli script nella directory `/etc/X11/xinit/xinitrc.d`.

I display manager `gdm` e `kdm` consentono agli utenti di specificare quale ambiente desktop utilizzare durante l'autenticazione selezionandolo dal menu **Session**. Se il display manager non specifica l'ambiente desktop, lo script `/etc/X11/xdm/Xsession` controllerà i file `.xsession` e `.Xclients` nella home directory utente per decidere quale ambiente desktop dovrà essere caricato. Come ultima risorsa viene usato il file `/etc/X11/xinit/Xclients` per selezionare un ambiente desktop oppure un Window Manager da usare allo stesso modo del runlevel 3.

Quando l'utente termina una sessione X sul display predefinito (`:`) ed effettua l'uscita, lo script `/etc/X11/xdm/TakeConsole` si avvia e assegna nuovamente la proprietà della console all'utente di root. Il display manager originale, che è rimasto operativo dopo la registrazione dell'utente, prende il controllo eseguendo lo spawn di un nuovo display manager. Questo riavvia il server XFree86, visualizza una nuova finestra di login e avvia ancora l'intero processo.

Per ulteriori informazioni relative al controllo del display manager sull'autenticazione utente, leggete la pagina `man xdm`.

6.5. Font

Red Hat Linux utilizza `xfs` (X Font Server) per fornire i font al server XFree86 e alle applicazioni client X che si connettono a esso. Sebbene sia possibile non ricorrere a `xfs` e posizionare i percorsi delle directory dei font nei file di configurazione `XFree86Config` e `XFree86Config-4`, `xfs` ha alcuni vantaggi:

- *Facilita l'aggiunta o la rimozione di font, incluso il percorso font per l'editing.* Il percorso font consiste in un insieme di percorsi nel filesystem in cui sono archiviati i file font. La funzione `xfs` mantiene il percorso font fuori dei file di configurazione di XFree86, semplificando notevolmente l'editing.
- *I font possono essere archiviati in una macchina che agisce come server font di rete, e pertanto condivisi tra i server X multipli della rete.* È possibile mantenere in un determinato luogo un set di font comuni e dividerlo con tutti gli utenti senza operazioni complicate.
- *Sono supportati più tipi di font.* `xfs` è in grado di gestire `TrueType`, `Type1` e i font bitmap.

I file di configurazione di XFree86 riconoscono se usare `xfs` o percorsi font codificati interni grazie all'impostazione `FontPath` presente nelle sezioni `Files`. Di default, `FontPath` è impostato su `unix/:7100`. In questo modo, il server XFree86 esegue la connessione alla porta 7100 usando un collegamento per la comunicazione interno. Il server `xfs`, in ascolto su quella porta, risponderà con informazioni relative ai font durante la query di XFree86.

La funzione `xfs` deve essere in esecuzione all'avvio di X. In caso contrario, si ritorna a un prompt di comando con un errore simile a `failed to set default font path 'unix/:7100'`. Controllate che `xfs` sia in esecuzione usando il comando `ps aux | grep xfs`. Di default, `xfs` è impostato per l'avvio nei runlevel 2, 3, 4 e 5; copre pertanto tutti i runlevel utilizzabili per l'avvio di X. Se `xfs` non è in esecuzione sul vostro sistema, avviatelo come root usando il comando `/sbin/service xfs start`. Usate le utility `/usr/sbin/ntsysv`, `serviceconf 0` / `/sbin/chkconfig` per garantirvi che si avvii ai runlevel corretti. Per maggiori informazioni su come configurare determinati servizi per un

particolare runlevel, consultate il capitolo chiamato *Controllare l'accesso ai servizi* nella *Official Red Hat Linux Customization Guide*

6.5.1. Configurazione di `xfs`

Lo script `/etc/rc.d/init.d/xfs` avvia il server `xfs`. Nel file `/etc/X11/fs/config` è possibile configurare diverse opzioni:

- `alternate-servers` — Imposta un elenco di server font alternati da usare se il server font corrente non è disponibile. Ogni server font della lista deve essere separato da una virgola.
- `catalogue` — Un elenco ordinato di percorso dei font, i quali contengono file font. Per avviare un nuovo font dell'elenco, ogni percorso deve essere seguito da una virgola.
È possibile usare la stringa `unscaled` subito dopo il percorso dei font per caricare per primi i font non scalabili di quel percorso. Potete quindi specificare nuovamente l'intero percorso in modo tale che siano caricati anche i font scalabili.
- `client-limit` — Stabilisce il numero di client serviti dal server font corrente prima di rifiutarne altri. Il valore di default è 10.
- `clone-self` — Decide se il server font esegue il clone di una nuova versione di se stesso quando si raggiunge `client-limit`. Di default, quest'opzione è impostata su `on`. Impostarla su `off` per disattivare questa caratteristica.
- `default-point-size` — Imposta la dimensione predefinita in punti per i font che non specificano tale valore. Il valore è impostato in punti decimali e 120, predefinito, corrisponde a 12 punti.
- `default-resolutions` — Specifica un elenco di risoluzioni supportate dal server XFree86. Ogni risoluzione dell'elenco deve essere separata da una virgola.
- `deferglyphs` — Dice a `xfs` se rimandare il caricamento di *glyphs*, un'immagine usata per rappresentare graficamente il font. È possibile disabilitare questa caratteristica (`none`), abilitarla per tutti i font (`all`) oppure impostarla solo per i font a 16 bit (16), ampiamente usati con le lingue asiatiche.
- `error-file` — Vi consente di specificare il path e il nome del file quando è possibile registrare errori `xfs`.
- `no-listen` — Dice a `xfs` di non ascoltare mediante un protocollo particolare. Di default, quest'opzione è impostata su `tcp` per impedire che `xfs` si metta in ascolto sulle porte TCP, principalmente per ragioni di sicurezza. Se state progettando di usare `xfs` per fornire i font a delle workstation collegate in una LAN, rimuovete `tcp` da questa linea.
- `port` — Specifica la porta TCP su cui è in ascolto `xfs` se `no-listen` non è presente o sarà disabilitata.
- `use-syslog` — Dice a `xfs` di usare il log di errore di sistema, se impostato su `on`.

6.5.2. Aggiungere font

Quando si usa `xfs`, aggiungere i font al sistema è piuttosto semplice. Usate il comando `chkfontpath --list` per visualizzare i percorsi dei font correntemente configurati sul sistema. Per aggiungere nuovi font in una nuova directory, seguite le istruzioni come utente root:

1. Create una directory dei font, come per esempio `/usr/share/fonts`, e posizionate i font al suo interno. Assicuratevi di impostare i permessi in modo corretto; è sufficiente che i file possano essere letti, non sono richiesti altri permessi.

2. Digitate il comando `chkfontpath --add <percorso-directory-font>`, dove `<percorso-directory-font>` è il percorso completo per la directory dei font. Questo aggiunge il font path considerato al file di configurazione `xfs`.



Nota Bene

La vostra directory dei font deve comprendere il file `fonts.dir`, affinché il comando `chkfontpath` operi correttamente. La creazione del file `fonts.dir`, come del resto di qualsiasi altro file usato da `xfs` con questi font, non viene trattata da questo documento.

Molti gruppi di font disponibili per Linux includono già questi file, pertanto potreste non avere la necessità di crearli manualmente.

3. Riavviate `xfs` usando il comando `service xfs restart`. Si renderà necessario riavviare anche la sessione X.
4. Digitando il comando `chkfontpath --list` viene visualizzato il nuovo percorso font. I font aggiunti sono ora disponibili per l'uso.

Il sito Web di Red Hat fornisce maggiori informazioni a riguardo. Andate all'indirizzo:

<http://www.redhat.com/support> per consultare ulteriore documentazione di supporto.

6.6. Risorse aggiuntive

Vi è ancora molto da aggiungere a proposito del server XFree86, dei client che si connettono a esso e dei vari Window Manager e ambienti desktop. Gli utenti più esperti interessati a personalizzare maggiormente la configurazione di XFree86, troveranno queste fonti d'informazione aggiuntive molto utili.

6.6.1. Documentazione installata

- `/usr/X11R6/lib/X11/doc/README` — descrive brevemente l'architettura XFree86 e fornisce ai nuovi utenti consigli su come reperire maggiori informazioni sul progetto XFree86.
- `/usr/X11R6/lib/X11/doc/README.Config` — Spiega le opzioni avanzate di configurazione per gli utenti della versione 3 di XFree86.
- `/usr/X11R6/lib/X11/doc/RELNOTES` — Per gli utenti più esperti che desiderano informarsi sulle nuove caratteristiche disponibili in XFree86.
- `man XF86Config` — Contiene informazioni sul file di configurazione di XFree86, inclusi i significati e le sintassi per le diverse sezioni all'interno dei file.
- `man XFree86` — La pagina `man` principale per tutte le informazioni su XFree86. Illustra in modo dettagliato le differenze tra connessioni locali e di rete del server X, esplora variabili d'ambiente comuni, elenca le opzioni delle righe di comando e fornisce utili combinazioni per l'uso dei tasti.
- `man Xserver` — Descrive il server X.

6.6.2. Siti Web utili

- <http://www.xfree86.org> — Home page del progetto XFree86 e che realizza la versione XFree86 Open Source del Sistema X Window. XFree86 è venduto insieme a Red Hat Linux per controllare l'hardware necessario e fornire un ambiente GUI.

- <http://dri.sourceforge.net> — Home page del progetto DRI (Direct Rendering Infrastructure). DRI è il componente di accelerazione 3D hardware di XFree86.
- <http://www.redhat.com/mirrors/LDP/HOWTO/XFree86-HOWTO> — Un documento HOWTO che spiega in modo dettagliato l'installazione manuale e la configurazione personalizzata di XFree86.
- <http://www.gnome.org> — Home page del progetto GNOME.
- <http://www.kde.org> — La home page dell'ambiente desktop KDE.

6.6.3. Libri correlati

- *The Concise Guide to XFree86 for Linux* di Aron Hsiao; Que — Fornisce la visione dell'operatività di XFree86 su sistemi Linux da un punto di vista professionale.
- *The New XFree86* di Bill Ball; Prima Publishing — Fornisce una buona panoramica generale di XFree86 e della sua interazione con i più diffusi ambienti desktop, quali GNOME e KDE.
- *Beginning GTK+ and GNOME* di Peter Wright; Wrox Press, Inc. — Introduce ai programmatori l'architettura di GNOME, mostrando loro come acquistare familiarità con GTK+.
- *GTK+/GNOME Application Development* di Havoc Pennington; New Riders Publishing — Uno sguardo approfondito nel cuore della programmazione GTK+, che dedica particolare attenzione ai codici campione e alle API disponibili.
- *KDE 2.0 Development* di David Sweet e Matthias Ettrich; Sams Publishing — Istruisce gli sviluppatori principianti e più esperti su come trarre vantaggio dalle istruzioni fondamentali per l'ambiente richieste per creare applicazioni QT per KDE.

Sicurezza

Moduli di autenticazione PAM

I programmi che forniscono privilegi devono essere in grado di autenticare gli utenti. Per esempio, dopo avere inserito il vostro nome utente e la vostra password per collegarvi al sistema, il processo di login utilizza questi dati per verificare la vostra identità.

I moduli di autenticazione PAM (*Pluggable Authentication Modules*) permettono all'amministratore di sistema di creare un procedimento di autenticazione per applicazioni che supportano tali moduli, senza dover ricompilare i programmi di autenticazione. Questo è possibile grazie a un'architettura modulare. I moduli che una particolare applicazione chiama sono determinati dal file di configurazione PAM relativo all'applicazione che si trova nella directory `/etc/pam.d/`.

Nella maggior parte dei casi non dovete modificare i file di configurazione PAM per le applicazioni che supportano tali moduli. Se usate l'RPM per installare programmi che richiedono un'autenticazione, vengono effettuate automaticamente le modifiche necessarie per l'autenticazione delle password. Tuttavia, se desiderate personalizzare la vostra configurazione, dovete conoscere la struttura del file (vedere la Sezione 7.2 per maggiori informazioni).

7.1. I vantaggi dei PAM

Se usati correttamente, i PAM sono di grande aiuto all'amministratore di sistema poiché:

- Forniscono uno schema di autenticazione comune che può essere usato con molte applicazioni diverse.
- Sono molto flessibili e consentono un elevato controllo di autenticazione per l'amministratore e lo sviluppatore di applicazioni.
- Permettono agli sviluppatori di produrre applicazioni senza implementare uno schema di autenticazione particolare. In tal modo, possono concentrarsi completamente sullo sviluppo dei loro programmi.

7.2. File di configurazione PAM

La directory `/etc/pam.d/` contiene i file di configurazione PAM delle applicazioni che supportano questi moduli. Nelle versioni precedenti di PAM veniva usato il file `/etc/pam.conf`, ma ora il suo utilizzo è sconsigliato. Il file `pam.conf` viene letto solo se la directory `/etc/pam.d/` non è presente sul sistema.

Ogni applicazione che supporta i PAM, o *servizio*, nome solitamente attribuito alle applicazioni utilizzate da molti utenti, ha un suo file nella directory `/etc/pam.d/`.

Questi file hanno un layout specifico che contiene delle chiamate ai moduli generalmente posizionati nella directory `/lib/security/`. Inoltre all'interno del file di configurazione PAM devono essere specificati: il tipo di modulo, l'opzione di controllo, il percorso del modulo e, talvolta, degli argomenti.

7.2.1. Nomi di servizio PAM

Il nome di ogni file di configurazione PAM contenuto nella directory `/etc/pam.d/` prende spunto dal servizio per il quale controlla l'accesso. Spetta al programma PAM definire il nome del servizio e installare il relativo file di configurazione PAM nella directory `pam.d`. Per esempio, il programma `login` definisce il nome del servizio `/etc/pam.d/login`.

In generale, il nome di servizio corrisponde al nome del programma usato per *accedere* al servizio, non al programma usato per *fornire* il servizio. Ecco perché il servizio `wu-ftpd` definisce il nome del suo servizio `/etc/pam.d/ftp`.

Le seguenti quattro sezioni descrivono il formato di base dei file di configurazione PAM e come sono utilizzati i moduli PAM per autenticare le applicazioni che supportano PAM.

7.3. Moduli PAM

Esistono quattro tipi diversi di moduli PAM che permettono di controllare l'accesso a determinati servizi e si correlano ad aspetti diversi del processo di autorizzazione:

- Il modulo `auth` fornisce l'effettiva autenticazione, per esempio, richiedendo e controllando una password e fornisce "credenziali", quali l'appartenenza al gruppo o i "ticket" di Kerberos.
- Il modulo `account` esegue un controllo per assicurarsi che l'accesso sia possibile. Per esempio, controlla se un account è scaduto o se l'utente ha il permesso di collegarsi a un determinato orario.
- Il modulo `password` viene usato per impostare le password.
- Il modulo `session` viene chiamato in causa una volta che l'autenticazione di un utente è stata eseguita e svolge ulteriori compiti richiesti per autorizzare l'accesso, per esempio montando la home directory dell'utente o rendendo disponibile la sua mailbox.



Nota Bene

Un singolo modulo può rivolgersi a uno o più moduli specificati sopra. Per esempio `pam_unix.so` si rivolge a tutti e quattro i tipi di moduli.

In un file di configurazione PAM il tipo di modulo è il primo aspetto definito. Per esempio una linea tipica di configurazione è:

```
auth      required /lib/security/pam_unix.so
```

Al PAM viene specificato di occuparsi del componente `auth` del modulo `pam_unix.so`.

7.3.1. Impilare i moduli

I moduli possono essere messi l'uno sull'altro in modo da poter essere utilizzati contemporaneamente. L'ordine con cui vengono elencati i moduli è molto importante nel processo di autenticazione.

I moduli inseriti a stack rendono più facile per l'amministratore richiedere la verifica di determinate condizioni prima di autenticare l'utente. Per esempio, `rlogin` utilizza normalmente almeno cinque moduli `auth`, come lo dimostra il suo file di configurazione PAM:

```
auth      required /lib/security/pam_nologin.so
auth      required /lib/security/pam_securetty.so
auth      required /lib/security/pam_env.so
auth      sufficient /lib/security/pam_rhosts_auth.so
auth      required /lib/security/pam_stack.so service=system-auth
```

Prima che qualcuno possa usare il comando `rlogin`, PAM controlla che non esista alcun file `/etc/nologin`, verifica se si sta provando a effettuare un collegamento remoto come `root` su una connessione di rete non cifrata e se possono essere caricate tutte le variabili d'ambiente. Viene quindi

eseguita un'autenticazione `rhosts` prima della connessione. Se l'autenticazione `rhosts` non va a buon fine, viene effettuata un'autenticazione standard della password.

7.3.2. Creazione di moduli

Nuovi moduli PAM possono essere aggiunti in qualsiasi momento e le applicazioni che supportano PAM li possono usare. Per esempio, se realizzate un nuovo sistema di creazione delle password e scrivete un modulo PAM per supportarlo, i programmi compatibili con PAM possono immediatamente usare il nuovo modulo e la nuova password senza dover essere ricompilati o modificati. Ciò permette di abbinare e verificare molto velocemente i metodi di autenticazione per diversi programmi senza doverli ricompilare.

La documentazione sulla scrittura dei moduli è disponibile nella directory `/usr/share/doc/pam-numero_versioni/`.

7.4. Opzioni di controllo PAM

Quando vengono controllati, tutti i moduli PAM generano un risultato positivo o negativo. Le opzioni indicano a PAM cosa fare con il risultato. Poiché i moduli possono essere ordinati in determinati modi, le opzioni permettono di stabilire l'importanza di un modulo rispetto ai moduli successivi.

Considerate il file di configurazione PAM `rlogin`:

```
auth      required      /lib/security/pam_nologin.so
auth      required      /lib/security/pam_securetty.so
auth      required      /lib/security/pam_env.so
auth      sufficient   /lib/security/pam_rhosts_auth.so
auth      required      /lib/security/pam_stack.so service=system-auth
```



Importante

L'ordine di chiamata dei moduli `required` non è fondamentale. Le opzioni di controllo `sufficient` e `requisite` determinano l'importanza dell'ordine. Per una descrizione dei tipi di opzioni di controllo leggete qui sotto.

Una volta specificato il tipo di modulo, le opzioni di controllo decidono quanta importanza ha quel particolare modulo al fine dell'obiettivo generale che consiste nel fornire l'accesso al servizio.

Lo standard PAM definisce quattro tipi di opzioni di controllo:

- `required` — il modulo deve superare il controllo perché l'autenticazione sia autorizzata. Se il controllo di un modulo `required` fallisce, l'utente non ne viene avvisato finché tutti gli altri moduli dello stesso tipo non sono stati controllati.
- `requisite` — il modulo deve superare la verifica perché l'autenticazione vada a buon fine. Tuttavia, se la verifica di un modulo `requisite` fallisce, l'utente ne viene immediatamente avvisato tramite un messaggio che richiama il primo modulo `required` o `requisite` che non ha superato la verifica.
- `sufficient` — il controllo del modulo viene ignorato se fallisce. Tuttavia, se un modulo con opzione `sufficient` supera la verifica così come tutti i moduli `required` che lo precedono, nessun altro modulo di questo tipo viene controllato e l'utente viene autenticato.

- `optional` — il controllo del modulo viene ignorato se non va a buon fine. Se riesce, non riveste un ruolo cruciale per il superamento o il fallimento dell'autenticazione di questo tipo di modulo. L'unico caso in cui un modulo con opzione `optional` è necessario ai fini dell'autenticazione è quando l'autenticazione di altri moduli dello stesso tipo non ha dato alcun risultato. In tal caso, il modulo `optional` determina l'autenticazione di tutti i moduli dello stesso tipo.

Adesso è disponibile una nuova sintassi di controllo ancora più efficace per PAM. Per maggiori informazioni, consultate la documentazione su PAM contenuta nella directory `/usr/share/doc/pam-numero_versione/`.

7.5. Percorsi dei moduli PAM

I percorsi dei moduli indicano a PAM dove trovare il modulo da usare con il tipo di modulo specificato. Solitamente viene fornito l'intero percorso del modulo, quale `/lib/security/pam_stack.so`. Tuttavia, se non viene indicato tutto il percorso (in altre parole, se il percorso non inizia con `/`), allora si suppone che il modulo indicato si trovi in `/lib/security/` — la directory di default dei moduli PAM.

7.6. Argomenti PAM

PAM utilizza degli argomenti per fornire informazioni a un modulo pluggable durante il processo di autenticazione di un determinato tipo di modulo. Tali argomenti permettono ai file di configurazione PAM di usare un modulo PAM comune, ma in modi differenti per un programma particolare.

Per esempio il modulo `pam_userdb.so` utilizza file nascosti del Berkeley DB per autenticare l'utente. Il Berkeley DB è un database Open Source concepito per essere incorporato in varie applicazioni per tenere traccia delle informazioni. Il modulo prende un argomento `db` specificando il file Berkeley DB da usare, che può variare in funzione del servizio.

Pertanto la linea `pam_userdb.so` di un file di configurazione PAM è:

```
auth      required /lib/security/pam_userdb.so db=percorso/del/file
```

Gli argomenti non validi vengono ignorati e non influenzano il superamento né il fallimento del modulo PAM. Quando viene passato un argomento non valido, viene solitamente inviato un errore a `/var/log/messages`. Tuttavia, poiché il metodo di report è controllato dal modulo PAM, è compito del modulo rilevare l'errore.

7.7. Esempi di file di configurazione PAM

Di seguito è riportato un file di configurazione PAM di esempio:

```
##PAM-1.0
auth      required /lib/security/pam_securetty.so
auth      required /lib/security/pam_unix.so shadow nullok
auth      required /lib/security/pam_nologin.so
account   required /lib/security/pam_unix.so
password  required /lib/security/pam_cracklib.so retry=3
password  required /lib/security/pam_unix.so shadow nullok use_authok
session   required /lib/security/pam_unix.so
```

La prima riga è un commento, come indicato dal carattere `#`, cioè il simbolo dei commenti per i file di configurazione PAM. Le righe da due a quattro contengono tre moduli da usare per l'autenticazione del login.

```
auth    required /lib/security/pam_securetty.so
```

Questa riga si assicura che *se* l'utente sta provando a collegarsi come root, la tty che sta utilizzando sia elencata nel file `/etc/securetty`, *se* tale file esiste.

```
auth    required /lib/security/pam_unix.so nullok
```

Questa riga chiede all'utente una password, che poi verifica usando le informazioni archiviate in `/etc/passwd` e, se questo file esiste, `/etc/shadow`. Il modulo `pam_unix.so` rileva e utilizza automaticamente le password shadow archiviate in `/etc/shadow` per autenticare gli utenti. Consultate la Sezione 5.5 per maggiori informazioni sulle password shadow.

L'argomento `nullok` specifica al modulo `pam_unix.so` di accettare una password vuota.

```
auth    required /lib/security/pam_nologin.so
```

Questa è la fase finale di autenticazione. Controlla se il file `/etc/nologin` esiste. Se `nologin` esiste e l'utente non è root, l'autenticazione non va a buon fine.



Nota Bene

In questo esempio, tutti e tre i moduli `auth` vengono controllati, anche se il primo modulo `auth` non supera la verifica. Questa strategia impedisce all'utente di sapere perché l'autenticazione non è permessa. Se conoscesse il motivo, l'utente riuscirebbe a capire come irrompere nel sistema.

```
account required /lib/security/pam_unix.so
```

Questa riga effettua, se necessario, una verifica dell'`account`. Per esempio se le password shadow sono state attivate, il modulo `pam_unix.so` verifica se l'`account` è scaduto o se l'utente non ha modificato la password nel periodo stabilito.

```
password required /lib/security/pam_cracklib.so retry=3
```

Se la password è scaduta, il componente del modulo `pam_cracklib.so` ne richiede una nuova. Quindi verifica la nuova password per vedere se può essere facilmente indovinata da un programma che ricostruisce le password. Se la verifica non va a buon fine, offre all'utente altre due possibilità per creare una password meno facile, in base all'argomento `retry=3`.

```
password required /lib/security/pam_unix.so shadow nullok use_authtok
```

Questa riga specifica che se il programma modifica la password dell'utente, dovrebbe utilizzare il componente `password` del modulo `pam_unix.so` per farlo. Succede solo se la porzione `auth` del modulo `pam_unix.so` ha stabilito che la password deve essere cambiata — per esempio se una password shadow è scaduta.

L'argomento `shadow` specifica al modulo di creare password shadow durante l'aggiornamento della password dell'utente.

L'argomento `nullok` specifica al modulo di consentire all'utente di modificare la password *da* una password vuota, altrimenti questa viene considerata come un blocco dell'`account`.

L'argomento finale di questa riga, `use_authtok`, fornisce un buon esempio di come utilizzare una struttura a stack per i moduli PAM. Questo argomento specifica al modulo di non richiedere all'utente una nuova password. Deve invece accettare qualsiasi password passata dal modulo precedente. In questo modo tutte le nuove password devono passare il controllo `pam_cracklib.so` per verificare la sicurezza delle password prima di accettarle.

```
session required /lib/security/pam_unix.so
```

La riga finale specifica che il componente del modulo di sessione `pam_unix.so` viene usato per gestire la sessione. Questo modulo registra il nome utente e il tipo di servizio nel file `/var/log/messages` all'inizio e alla fine di ogni sessione. Può essere supportato da altri moduli di sessione per ottenere una migliore funzionalità.

Nel file di configurazione di esempio successivo viene illustrato l'uso del modulo `auth` per il programma `rlogin`, che consente agli utenti di effettuare una connessione remota.

```
##PAM-1.0
auth      required    /lib/security/pam_nologin.so
auth      required    /lib/security/pam_securetty.so
auth      required    /lib/security/pam_env.so
auth      sufficient  /lib/security/pam_rhosts_auth.so
auth      required    /lib/security/pam_stack.so service=system-auth
```

Per prima cosa, `pam_nologin.so` verifica se `/etc/nologin` esiste. In caso positivo, può collegarsi solo l'utente `root`.

```
auth      required    /lib/security/pam_securetty.so
```

Il modulo `pam_securetty.so` impedisce quindi i login di `root` su terminali non sicuri. In questo modo tutti i tentativi `rlogin` di `root` sono disabilitati per ragioni di sicurezza.



Suggerimento

Se dovete collegarvi come `root`, usate `OpenSSH`. Per maggiori informazioni sul protocollo `SSH`, consultate il Capitolo 9.

```
auth      required    /lib/security/pam_env.so
```

Questa riga carica il modulo `pam_env.so`, che imposta le variabili d'ambiente specificate in `/etc/security/pam_env.conf`.

```
auth      sufficient  /lib/security/pam_rhosts_auth.so
```

Il modulo `pam_rhosts_auth.so` quindi autentica l'utente per l'uso di `.rhosts` nella sua home directory. Se va a buon fine, PAM autentica immediatamente la sessione `rlogin`. Se `pam_rhosts_auth.so` non riesce ad autenticare l'utente, il tentativo di autenticazione non riuscito viene ignorato.

```
auth      required    /lib/security/pam_stack.so service=system-auth
```

Se il modulo `pam_rhosts_auth.so` non riesce ad autenticare l'utente, il modulo `pam_stack.so` esegue una regolare autenticazione della password.

L'argomento `service=system-auth` indica che l'utente deve ora sottoporsi all'autenticazione PAM per l'autorizzazione del sistema in `/etc/pam.d/system-auth`.



Nota Bene

Se non volete che venga visualizzato il prompt per inserire la password quando `securetty` fallisce e determina che l'utente sta provando a collegarsi come `root` in modo remoto, potete cambiare il modulo `pam_securetty.so` da `required` a `requisite`.

7.8. PAM e proprietà dei dispositivi

Red Hat Linux consente al primo utente che si collega con i privilegi corretti alla console fisica della macchina di gestire i dispositivi ed effettuare operazioni generalmente riservate all'utente root. Questo avviene mediante il modulo PAM `pam_console.so`.

7.8.1. Proprietà dei dispositivi

Quando un utente si collega a una macchina in Red Hat Linux, il modulo `pam_console.so` viene chiamato da `login` o dai programmi di login grafici, **gdm** e **kdm**. Se questo utente è il primo a collegarsi alla console fisica — chiamata *utente console* — il modulo gli assegna la proprietà di vari dispositivi generalmente appartenenti all'utente root. L'utente console ha la proprietà di questi dispositivi fino al termine della sessione locale. Quando l'utente non è più collegato, la proprietà dei dispositivi gli viene tolta.

I dispositivi interessati includono, ma non sono limitati a, schede audio, unità floppy e CD-ROM.

Questo consente all'utente locale di gestire tali dispositivi senza essere root, semplificando i task più comuni dell'utente console.

Potete modificare l'elenco dei dispositivi controllati da `pam_console.so` nel file `/etc/security/console.perms`.

7.8.2. Accesso alle applicazioni

L'utente console può accedere a qualsiasi programma che contiene il nome del comando nella directory `/etc/security/console.apps/`. Questi file non devono contenere dati, ma devono avere lo stesso nome del comando a cui corrispondono.

Uno dei gruppi di applicazioni a cui l'utente console può accedere è composto da tre programmi per spegnere o riavviare il sistema. Sono:

- `/sbin/halt`
- `/sbin/reboot`
- `/sbin/poweroff`

Si tratta di applicazioni che supportano i moduli PAM, quindi chiamano `pam_console.so` per poter funzionare.

Per maggiori informazioni consultate le pagine man per `pam_console`, `console.perms` e `console.apps`.

7.9. Risorse aggiuntive

Di seguito è riportato un elenco di fonti di informazioni per l'utilizzo e la configurazione di PAM sul vostro sistema. Oltre a queste fonti, dovrete consultare i file di configurazione PAM del vostro sistema per una migliore comprensione di come sono strutturati.

7.9.1. Documentazione installata

- Pagina man di `pam` — Buone informazioni introduttive su PAM, tra cui la struttura e lo scopo dei file di configurazione PAM.
- `/usr/share/doc/pam-numero_versione` — Contiene una *System Administrators' Guide*, un *Module Writers' Manual* e un *Application Developers' Manual*. Contiene inoltre una copia dello standard PAM, DCE-RFC 86.0.

7.9.2. Siti Web utili

- <http://www.kernel.org/pub/linux/libs/pam> — Il primo sito Web di distribuzione del progetto Linux-PAM, contenente informazioni su vari moduli e applicazioni PAM, le relative FAQ e una documentazione aggiuntiva su PAM.

Wrapper TCP e xinetd

Il controllo dell'accesso ai servizi di rete può rappresentare una sfida. I firewall permettono di controllare l'accesso in ingresso e in uscita da una rete, ma possono essere difficili da configurare. I wrapper TCP e `xinetd` controllano l'accesso ai servizi tramite il nome di host e gli indirizzi IP. Inoltre, questi tool includono delle capacità di gestione del login e dell'utilizzo facili da configurare.

8.1. Cosa sono i wrapper TCP?

I wrapper TCP sono installati per default con l'installazione di classe server per Red Hat Linux 8.0 e forniscono il controllo dell'accesso a numerosi servizi. Molti dei servizi di rete moderni, quali SSH, Telnet e FTP, utilizzano i *wrapper TCP*, un programma studiato per posizionarsi tra una richiesta in ingresso e il servizio richiesto.

L'idea alla base dei wrapper TCP è che le richieste del client alle applicazioni del server sono "wrapped" (inglobate) da un servizio di autenticazione. L'operazione permette di esercitare un maggior controllo su chi cerca di utilizzare il servizio rispetto al metodo tradizionale che prevede la connessione diretta del client a un dato servizio.

La funzionalità dei wrapper TCP è fornita da `libwrap.a`, una libreria con la quale i servizi di rete, quali `xinetd`, `sshd` e `portmap`, sono compilati. Altri servizi di rete, anche i programmi di networking che potreste scrivere, possono essere compilati con `libwrap.a`. Red Hat Linux raggruppa la libreria e i programmi di wrapper TCP necessari nel file RPM `tcp_wrappers-<versione>`.

8.1.1. Vantaggi dei wrapper TCP

Quando qualcuno prova ad accedere a un servizio di rete usando i wrapper TCP, un piccolo programma wrapper riporta il nome del servizio richiesto e le informazioni host del client. Il programma wrapper non invia informazioni direttamente al client e, una volta che le direttive di controllo dell'accesso sono applicate, il programma wrapper scompare per alleggerire la comunicazione tra il client e il server. Il client e il server possono riprendere le attività senza l'ulteriore intervento del wrapper.

I wrapper TCP forniscono due vantaggi in più rispetto alle altre tecniche di controllo dei servizi di rete:

- *Il client che si collega non sa che vengono usati i wrapper TCP.* Gli utenti abilitati non si accorgono di niente, mentre gli hacker non sapranno mai perché il collegamento non è andato a buon fine.
- *I wrapper TCP funzionano indipendentemente dalle applicazioni protette dal programma wrapper.* In questo modo, molte applicazioni possono condividere un insieme comune di file di configurazione semplificandone così la gestione.

8.2. Elenchi di controllo dell'accesso basati sugli host

L'accesso basato sugli host per i servizi che utilizzano i wrapper TCP è controllato da due file: `hosts.allow` e `hosts.deny`. Questi file usano un formato semplice per controllare gli accessi ad alcuni servizi sul server.

La regola generale è quella di permettere a chiunque di accedere ai servizi se non è specificata alcuna regola in `hosts.allow` o `hosts.deny`.

L'ordine ha molta importanza, poiché le regole contenute in `hosts.allow` hanno la precedenza su quelle specificate in `hosts.deny`. Anche se in `hosts.deny` è contenuta una regola che vieta l'accesso a un servizio, gli host a cui è stata data l'autorizzazione in `hosts.allow` possono comunque accedere al servizio. Le regole specificate in ogni file hanno effetto dal basso verso l'alto.

Le modifiche apportate ai file hanno effetto immediato sui servizi interessati; non è necessario riavviare i servizi.

8.2.1. Regole di formattazione

Tutte le regole di controllo dell'accesso sono contenute in `hosts.allow` e `hosts.deny`, e tutte le linee vuote e quelle che iniziano con `#` vengono ignorate. Ogni regola deve essere su una linea separata.

Le regole devono essere formattate nel modo seguente:

```
<elenco_demoni>:
<elenco_client>[: spawn
<comando_shell> ]
```

Ognuna di queste opzioni si riferisce a una parte della regola:

- `daemon_list` — elenco di uno o più nomi di processi o caratteri jolly, separati da spazi.
- `client_list` — uno o più nomi e indirizzi di host, modelli o caratteri jolly, separati da spazi e da usare quando un nome di processo corrisponde a un servizio richiesto.
- `shell_command` — elemento opzionale che specifica qualcosa che deve essere fatto nel caso venga utilizzata una regola.

I modelli sono particolarmente utili quando devono essere specificati gruppi di client che possono o meno avere accesso a servizi. Inserendo il carattere "." all'inizio di una stringa, tutti gli host che condividono la fine della stringa obbediscono alla regola. Perciò, `.domain.com` vale sia per `system1.domain.com` sia per `system2.domain.com`. Il carattere "." posizionato alla fine di una stringa ha lo stesso effetto, tranne per il fatto che va in un'altra direzione, e viene principalmente usato negli indirizzi IP. Se inserito in `192.168.0.` viene applicato a tutta la classe C degli indirizzi IP. Anche le espressioni della maschera di rete possono servire da modello per controllare l'accesso a un gruppo di indirizzi IP. Gli asterischi (*) e i punti di domanda (?) permettono di selezionare interi gruppi di nomi o di indirizzi IP, purché non vengano utilizzati nella stessa stringa dove sono presenti altri tipi di modelli.

Se l'elenco dei nomi autorizzati ad accedere a un servizio è troppo lungo o difficile da controllare all'interno di `hosts.allow` o `hosts.deny`, potete anche specificare l'intero percorso di un file (per esempio `/etc/telnet.hosts.deny`). Il file deve contenere i nomi, gli indirizzi o i modelli, separati da spazi, ai quali si vuole autorizzare o rifiutare l'accesso al servizio. Il metodo è consigliabile anche per condividere elenchi di controllo dell'accesso fra diversi servizi, poiché basta inserire le modifiche in un file perché siano attive per ogni servizio.

Le opzioni di seguito riportate possono essere utilizzate in sostituzione a specifici host o gruppi di host:

- `ALL` — si applica a tutti i client con quel particolare servizio oppure a tutti i servizi che usano il controllo dell'accesso. L'opzione `ALL` può essere usata anche per i demoni.
- `LOCAL` — si applica a tutti gli host che non contengono il carattere ".".
- `KNOWN` — si applica a ogni host il cui nome o indirizzo viene riconosciuto o quando l'utente viene riconosciuto.

- UNKNOWN — si applica a ogni host il cui nome o indirizzo di host non viene riconosciuto o quando l'utente non è riconosciuto.
- PARANOID — si applica quando il nome di host non corrisponde all'indirizzo di host.



Attenzione

Usate le opzioni `KNOWN`, `UNKNOWN` e `PARANOID` con molta cautela poiché rischiate di impedire a utenti di accedere a servizi di rete ai quali invece dovrebbero poter accedere.

Il linguaggio di controllo dell'accesso contiene anche un potente operatore, `EXCEPT`, che permette di associare elenchi separati alla stessa regola. Quando `EXCEPT` viene usato fra due elenchi, ha effetto il primo elenco a meno che una voce del secondo elenco corrisponda a un'entità del primo elenco. `EXCEPT` può essere usato con demoni e client. Di seguito è riportato un esempio di `hosts.allow`:

```
# all domain.com hosts are allowed to connect
# to all services except cracker.domain.com
ALL: .domain.com EXCEPT cracker.domain.com

# 123.123.123.* addresses can use all services except FTP
ALL EXCEPT in.ftpd: 123.123.123.
```



Nota Bene

Anziché usare `EXCEPT`, è più logico, da un punto di vista organizzativo, inserire le eccezioni alla regola nell'altro file di controllo dell'accesso. In questo modo gli amministratori possono esaminare rapidamente i file adeguati per vedere quali host hanno o non hanno accesso ai servizi, senza dover impiegare vari operatori `EXCEPT` ed elaborare una soluzione.

Per gestire al meglio il controllo dell'accesso con `hosts.allow` e `hosts.deny`, usate questi due file contemporaneamente.

Gli utenti che desiderano autorizzare solo ad alcuni host di accedere ai servizi, inseriscono `ALL: ALL` in `hosts.deny`. Successivamente inseriscono delle linee in `hosts.allow`, quali:

```
in.telnetd: 10.0.1.24
in.ftpd: 10.0.1. EXCEPT 10.0.1.1
```

In alternativa, se desiderate concedere l'utilizzo dei servizi di rete a tutti tranne che determinati host, lasciate vuoto `hosts.allow` e aggiungete a `hosts.deny` le necessarie restrizioni, quali:

```
in.fingerd: 192.168.0.2
```



Avvertenza

Siate prudenti quando usate nomi di host e di dominio con i file di controllo, soprattutto con `hosts.deny`. Le persone che non hanno accesso ai servizi potrebbero usare dei sotterfugi per annullare l'effetto delle regole e accedere ai servizi. Inoltre, se il sistema permette l'accesso basandosi sui nomi di host e di dominio, qualsiasi interruzione nel servizio DNS impedirebbe anche agli utenti autorizzati di usare i servizi di rete.

L'utilizzo degli indirizzi IP, dove possibile, può prevenire molti problemi durante la creazione delle regole di controllo dell'accesso, soprattutto quelle che vietano l'accesso.

Oltre ad autorizzare o a negare ad alcuni host l'accesso ai servizi, il linguaggio di controllo dell'accesso supporta anche i comandi della shell quando tale regola viene utilizzata. I comandi della shell vengono usati soprattutto con le regole che negano l'accesso e permettono di impostare delle trappole che solitamente registrano in un file informazioni sui tentativi di accesso o inviano queste informazioni all'amministratore. Di seguito è riportato un esempio di trappola nel file `hosts.deny` che scrive una linea di log contenente la data e le informazioni sul client ogni volta che un host compreso tra 10.0.1.0 e 10.0.1.255 prova a collegarsi via Telnet:

```
in.telnetd: 10.0.1.: spawn (/bin/echo `date` %c >> /var/log/telnet.log) &
```

Utilizzando i comandi della shell, avete anche a disposizione diverse *espansioni* contenenti informazioni sul client, il server e il processo utilizzato. Qui di seguito è riportato un elenco delle espansioni supportate:

- `%a` — l'indirizzo IP del client.
- `%A` — l'indirizzo IP del server.
- `%c` — vari tipi di informazioni sul nome utente e host, oppure il nome utente e l'indirizzo IP.
- `%d` — il nome del processo demone.
- `%h` — il nome host del client (o l'indirizzo IP se il nome host non è disponibile).
- `%H` — il nome host del server (o l'indirizzo IP se il nome host non è disponibile).
- `%n` — il nome host del client. Se non è disponibile, viene visualizzato `unknown`. Se il nome e l'indirizzo host del client non corrispondono, viene visualizzato `paranoid`.
- `%N` — il nome host del server. Se non è disponibile, viene visualizzato `unknown`. Se il nome e l'indirizzo host del server non corrispondono, compare a video `paranoid`.
- `%p` — l'ID del processo demone.
- `%s` — vari tipi di informazioni del server, quali il processo demone e l'indirizzo host o IP del server.
- `%u` — il nome utente del client. Se non è disponibile, compare a video `unknown`.

Per una spiegazione esauriente dei comandi della shell, nonché ulteriori esempi del controllo di accesso, consultate la pagina man di `hosts access`.



Nota Bene

Un'attenzione particolare deve essere data al comando `portmap` quando viene usato con gli elenchi di controllo dell'accesso host. Poiché i nomi host non sono supportati, solo gli indirizzi IP o l'opzione `ALL` possono essere usati quando vengono specificati gli host che hanno o non hanno accesso ai servizi. Inoltre, le modifiche apportate agli elenchi di controllo dell'accesso host relative a `portmap` potrebbero non essere subito attive.

Stante il funzionamento dei servizi più utilizzati, come NIS e NFS, dipende da `portmap`, tenete a mente questo limite prima che facciate dipendere il controllo dell'accesso di alcuni host da `hosts.allow` e `hosts.deny`.

8.3. Controllo dell'accesso tramite xinetd

I vantaggi offerti dai wrapper TCP aumentano notevolmente se la libreria `libwrap.a` viene usata con `xinetd`, un *super demone* che permette un maggiore controllo dell'accesso, del login, del binding, del reindirizzamento e dell'utilizzo delle risorse.

Red Hat Linux configura una varietà di servizi di rete famosi utilizzabili con `xinetd`, inclusi FTP, IMAP, POP e telnet. Quando si accede a questi servizi tramite i loro numeri di porta in `/etc/services`, il demone `xinetd` gestisce la richiesta. Prima di attivare il servizio di rete richiesto, `xinetd` controlla che le informazioni host del client corrispondano alle regole di controllo dell'accesso, il numero di prove per questi servizi è limitato e tutte le altre regole specificate per quel servizio o tutti i servizi `xinetd` vengono applicati. Una volta che il servizio richiesto viene attivato per il client in connessione, `xinetd` torna a dormire e aspetta le altre richieste per i servizi che gestisce.

8.3.1. File di configurazione xinetd

Il servizio `xinetd` è controllato dal file `/etc/xinetd.conf` e dai vari file specifici per i servizi contenuti nella directory `/etc/xinetd.d`.

8.3.1.1. /etc/xinetd.conf

Il file `xinetd.conf` appartiene alla famiglia di tutti gli altri file di configurazione dei servizi controllati da `xinetd` e, ogni volta che `xinetd` si avvia, vengono analizzati anche i file specifici al servizio. Per default, `xinetd.conf` contiene alcuni parametri basici di configurazione che si applicano a tutti i servizi. Ecco un esempio di un tipico file `xinetd.conf`:

```
defaults
{
  instances                = 60
    log_type                = SYSLOG authpriv
    log_on_success         = HOST PID
    log_on_failure         = HOST
  cps                      =25 30
}

includedir /etc/xinetd.d
```

Queste linee controllano vari aspetti del funzionamento di `xinetd`:

- `instances` — imposta il numero massimo di richieste che un servizio può gestire alla volta.
- `log_type` — indica a `xinetd` di usare il registro `authpriv`, specificato in `/etc/syslog.conf` e impostato su `/var/log/secure` per default, anziché usare un altro file. Se venisse invece usata l'opzione `FILE /var/log/xinetdlog`, il login di `xinetd` verrebbe spostato in un altro file `/var/log/xinetdlog`.
- `log_on_success` — dice a `xinetd` cosa registrare se la connessione avviene correttamente. Per default, vengono registrati l'indirizzo IP dell'host remoto e l'ID del server che elabora la richiesta.
- `log_on_failure` — indica a `xinetd` cosa registrare se la connessione fallisce o non è autorizzata.
- `cps` — indica a `xinetd` di consentire più di 25 connessioni al secondo verso un dato servizio. Quando tale limite viene raggiunto, il servizio entra in pausa per 30 secondo.



Nota Bene

I parametri `log_on_success` e `log_on_failure` in `/etc/xinetd.conf` vengono spesso modificati da ogni servizio, il che significa che le connessioni corrette o fallite registrano più di quanto sia indicato qui.

Diverse opzioni di registrazione sono utilizzabili in `/etc/xinetd.conf` e nei file di configurazione `xinetd` specifici per i servizi:

- `ATTEMPT` — registra i tentativi falliti di accesso (`log_on_failure`).
- `DURATION` — registra il tempo di utilizzo di un servizio da parte di un sistema remoto (`log_on_success`).
- `EXIT` — registra il segnale di chiusura del servizio (`log_on_success`).
- `HOST` — registra l'indirizzo IP dell'host remoto (`log_on_failure` e `log_on_success`).
- `PID` — registra l'ID del server che riceve la richiesta (`log_on_success`).
- `RECORD` — registra informazioni relative al sistema remoto nel caso in cui il servizio non si possa avviare. Solo alcuni servizi, quali `login` e `finger`, possono usare quest'opzione (`log_on_failure`).
- `USERID` — registra l'utente remoto usando il metodo definito in RFC 1413 per tutti i servizi stream multi-thread (`log_on_failure` e `log_on_success`).

Ulteriori opzioni sono disponibili per `/etc/xinetd.conf`, per esempio `per_source`, che limita il numero massimo di connessioni da un indirizzo IP a un servizio. Per maggiori informazioni, consultate la pagina `man` di `xinetd`.

8.3.1.2. File nella directory `/etc/xinetd.d`

I vari file contenuti nella directory `/etc/xinetd.d` vengono letti ogni volta che si avvia `xinetd` e ciò è dovuto al fatto che `includedir /etc/xinetd.d` è specificato in fondo al file `/etc/xinetd.conf`. Questi file, chiamati `finger`, `ipop3` e `rlogin`, sono collegati ai vari servizi controllati da `xinetd`.

I file in `/etc/xinetd.d` usano le stesse convenzioni di `/etc/xinetd.conf`. Il motivo principale per cui essi sono immagazzinati in file di configurazione separati è per agevolare l'aggiunta o la rimozione di servizi da `xinetd` senza coinvolgere altri servizi.

Per avere un'idea della struttura di questi file, esaminate il file `wu-ftp`:

```

        service ftp
{
    socket_type          = stream
    wait                = no
    user                = root
    server               = /usr/sbin/in.ftpd
    server_args         = -l -a
    log_on_success      += DURATION USERID
    log_on_failure      += USERID
    nice                = 10
    disable              = yes
}

```

La prima linea definisce il nome del servizio in via di configurazione. Le linee fra parentesi contengono vari parametri che definiscono il modo in cui il servizio viene avviato e utilizzato. Il file

`wu-ftp` indica che il servizio FTP utilizza un tipo di socket `stream` (anziché `dgram`), il file binario eseguibile da usare, gli argomenti da passare al binario, le informazioni da registrare oltre ai parametri `/etc/xinetd.conf`, le priorità con cui avviare il servizio e altro ancora.

L'utilizzo di `xinetd` con un determinato servizio può servire anche come livello base di protezione da un attacco di rifiuto del servizio (Dos, Denial of Service). L'opzione `max_load` usa un valore floating point per impostare una soglia di utilizzo CPU quando nessun'altra connessione a un servizio viene accettata, per evitare che alcuni servizi sovraccarichino il sistema. L'opzione `cps` accetta un valore intero per impostare un limite nel numero di connessioni disponibili al secondo. Impostando un valore basso, per esempio 3, evitate che gli hacker effettuino troppe richieste simultanee per un servizio.

8.3.1.3. Controllo dell'accesso tramite `xinetd`

Gli utenti dei servizi `xinetd` possono scegliere di usare i file di controllo dell'accesso host TCP wrapper (`hosts.allow` e `hosts.deny`), fornire il controllo dell'accesso tramite i file di configurazione `xinetd`, o un insieme dei due. Informazioni relative all'uso dei file di controllo dell'accesso host wrapper TCP sono riportate alla Sezione 8.2. La sezione spiega come usare `xinetd` per regolare l'accesso ai servizi che controlla.



Nota Bene

A differenza dei file di configurazione del controllo dell'accesso wrapper TCP, le modifiche apportate ai file di configurazione `xinetd` richiedono che vengano riavviati il servizio `xinetd` e i servizi interessati.

Il controllo dell'accesso fornito da `xinetd` è diverso dal metodo usato dai wrapper TCP. Mentre i wrapper TCP raggruppano tutta la configurazione dell'accesso in due file `/etc/hosts.allow` e `/etc/hosts.deny`, ogni file di servizio in `/etc/xinetd.d` può contenere regole di controllo dell'accesso basate sugli host che hanno il permesso di usare il servizio.

Le opzioni seguenti sono supportate nei file `xinetd` per controllare l'accesso di host:

- `only_from` — permette agli host specificati di usare il servizio.
- `no_access` — impedisce a questi utenti di usare il servizio.
- `access_times` — specifica la fascia oraria in cui un particolare servizio può essere utilizzato. La fascia oraria deve essere specificata nel formato `HH:MM-HH:MM` e utilizzare le 24 ore della giornata.

Le opzioni `only_from` e `no_access` possono usare un elenco di indirizzi IP o nomi host, oppure potete specificare un'intera rete. Come per i wrapper TCP, se associate il controllo di `xinetd` con la configurazione di registrazione adeguata, potete non solo bloccare la richiesta di accesso, ma registrare anche ogni tentativo effettuato.

Per esempio, il seguente file `/etc/xinetd.d/telnet` può bloccare l'accesso `telnet` a un sistema da parte di uno specifico gruppo di rete e restringere la fascia oraria durante la quale anche gli utenti autorizzati possono collegarsi:

```
service telnet
{
    disable           = no
    flags             = REUSE
    socket_type       = stream
    wait              = no
    user              = root
    server            = /usr/sbin/in.telnetd
    log_on_failure    += USERID
}
```

```

no_access      = 10.0.1.0/24
log_on_success += PID HOST EXIT
access_times   = 09:45-16:15
}

```

In questo esempio, quando un sistema della sotto rete 10.0.1.0/24, per esempio 10.0.1.2, prova a collegarsi tramite telnet all'host boo, riceve il messaggio `Connection closed by foreign host.` Inoltre, il tentativo di login viene registrato in `/var/log/secure`:

```

May 15 17:35:47 boo xinetd[16188]: START: telnet pid=16191 from=10.0.1.2
May 15 17:38:49 boo xinetd[16252]: START: telnet pid=16256 from=10.0.1.2
May 15 17:38:49 boo xinetd[16256]: FAIL: telnet address from=10.0.1.2
May 15 17:38:49 booxinetd[16252]: EXIT: telnet status=0 pid=16256

```

8.3.1.4. Collegamento e ridirezionamento delle porte

I file di configurazione dei servizi di xinetd supportano anche il collegamento dei servizi a particolari indirizzi IP e il ridirezionamento delle richieste in entrata ad altri indirizzi IP, nomi host o porte.

Il collegamento, controllato tramite l'opzione `bind` nei file di configurazione dei servizi, collega un servizio a un particolare indirizzo IP usato con il sistema, permettendo così di accettare unicamente le richieste che utilizzano tale indirizzo IP per accedere al servizio. Questo è particolarmente utile con più adattatori di rete e indirizzi IP, come per esempio nelle macchine usate come firewall, con un adattatore di rete rivolto verso Internet e l'altro collegato a una rete interna. Gli hacker che tentano di collegarsi a un servizio, come Telnet o FTP, tramite la connessione a Internet vengono bloccati mentre gli utenti interni possono collegarsi al servizio tramite la NIC collegata alla rete interna.

L'opzione `redirect`, che accetta un indirizzo IP o un nome host seguito da un numero di porta, dice al servizio di ridirezionare tutte le richieste per il servizio verso la posizione specificata. Questa funzione può essere usata per puntare verso un altro numero di porta sullo stesso sistema, ridirezionare la richiesta verso altri indirizzi IP sulla stessa macchina, inviare la richiesta a un numero di porta e sistema completamente diverso oppure a una combinazione di queste opzioni. In questo modo, un utente che si collega a un servizio su un sistema può essere instradato verso un altro sistema senza causare problemi.

Il demone xinetd è in grado di compiere questo reindirizzamento creando un processo che rimane attivo durante la connessione tra la macchina client che svolge la richiesta e l'host che effettivamente fornisce il servizio e che trasferisce i dati fra i due sistemi.

La vera forza delle opzioni `bind` e `redirect` si nota quando queste vengono usate insieme. Collegando un servizio a un indirizzo IP su un sistema e reindirizzando successivamente la richiesta per il servizio a una seconda macchina che può essere vista solo dalla prima macchina, potete usare un sistema interno che fornisce servizi a una rete totalmente diversa. Altrimenti, le due opzioni possono essere usate per limitare l'esposizione di un servizio su una macchina multihome a un indirizzo IP conosciuto, e ridirigere tutte le richieste per il servizio verso un'altra macchina appositamente configurata.

Per esempio, esaminate un sistema usato come firewall i cui parametri per il servizio telnet sono:

```

service telnet
{
    socket_type = stream
    wait       = no
    server     = /usr/sbin/in.telnetd
    log_on_success += DURATION USERID
    log_on_failure += USERID
    bind       = 123.123.123.123
    redirect   = 10.0.1.13 21 23
}

```

```
}
```

Le opzioni `bind` e `redirect` contenute in questo file assicurano che il servizio telnet della macchina sia collegato all'indirizzo IP esterno (123.123.123.123), quello rivolto verso Internet. Inoltre, tutte le richieste per il servizio telnet inviate a 123.123.123.123 vengono ridirezionate tramite un altro adattatore di rete a un indirizzo IP interno (10.0.1.13) accessibile solo dal firewall e dai sistemi interni. Il firewall invia in seguito la comunicazione tra i due sistemi e il sistema in collegamento crede di essere collegato a 123.123.123.123 mentre in realtà è collegato a un'altra macchina.

Questa funzione è particolarmente utile per gli utenti con connessioni a banda larga e un unico indirizzo IP fisso. Quando viene utilizzato il NAT (Network Address Translation), i sistemi dietro alla macchina gateway che utilizzano indirizzi IP unicamente interni non sono disponibili fuori dal sistema gateway. Tuttavia, quando alcuni servizi controllati da `xinetd` sono configurati con le opzioni `bind` e `redirect`, la macchina gateway può agire come un tipo di proxy fra i sistemi esterni e una macchina interna configurata per fornire il servizio. Inoltre, le varie opzioni di controllo dell'accesso e di registrazione di `xinetd` sono disponibili per ulteriori protezioni, come la limitazione del numero di connessioni simultanee per il servizio reindirizzato.

8.4. Risorse aggiuntive

Ulteriori informazioni relative ai wrapper TCP e a `xinetd` sono disponibili sul vostro sistema e sul Web.

8.4.1. Documentazione installata

La documentazione fornita con il sistema costituisce un'ottimo punto di partenza per cercare altre informazioni sui wrapper TCP, su `xinetd` e sulle opzioni per la configurazione del controllo di accesso.

- `/usr/share/doc/tcp_wrappers-<versione>` — contiene un file `README` che spiega il funzionamento dei wrapper TCP e i vari rischi esistenti nell'utilizzo dei nomi e indirizzi host.
- `/usr/share/doc/xinetd-<versione>` — include un file `README` che spiega i vari aspetti del controllo di accesso e un file `sample.conf` contenente varie idee per modificare le configurazioni del servizio `/etc/xinetd.d`.
- Per informazioni dettagliate relative alla creazione delle regole di controllo dell'accesso dei wrapper TCP, consultate le pagine `man hosts_access(5)` e `hosts_options(5)`.
- Le pagine `man xinetd(8)` e `xinetd.conf(5)` contengono ulteriori informazioni per la creazione dei file di configurazione `xinetd` e una descrizione del funzionamento di `xinetd`.

8.4.2. Siti Web utili

- <http://www.xinetd.org> — la pagina principale di `xinetd`, contenente file di configurazione di esempio, un elenco di tutte le caratteristiche e delle FAQ informative.
- <http://www.macsecurity.org/resources/xinetd/tutorial.shtml> — sito che spiega in modo dettagliato i vari modi per modificare i file di configurazione predefiniti di `xinetd` per adattarli alle proprie necessità di sicurezza.

Protocollo SSH

SSH™ consente il collegamento in modalità remota. A differenza di `rlogin` o `telnet`, SSH cripta la sessione di login, impedendo alle persone non autorizzate di raccogliere le password in chiaro.

SSH è progettato per sostituire applicazioni precedenti, meno sicure utilizzate per l'accesso a sistemi remoti come **telnet** o **rsh**. Un programma chiamato `scp` sostituisce i programmi meno recenti per copiare i file tra host, quali **ftp** o **rcp**. Dato che queste applicazioni non cifrano le password tra il client e il server, si consiglia di utilizzarle il meno possibile. Se usate dei metodi sicuri per collegarvi ad altri sistemi in remoto, correte meno rischi per la sicurezza del vostro sistema e del sistema a cui vi collegate.

9.1. Caratteristiche di SSH

SSH (o *Secure SHell*) è un protocollo che vi consente di stabilire connessioni sicure tra due sistemi mediante un'architettura client server. Nel protocollo SSH il computer client avvia tutte le connessioni con il computer server.

Il protocollo SSH fornisce le seguenti misure di protezione:

- Dopo una connessione iniziale, il client verifica il collegamento allo stesso server durante sessioni successive.
- Il client trasmette le proprie informazioni di autenticazione al server, per esempio il nome utente e la password, in forma cifrata.
- Tutti i dati inviati e ricevuti durante la connessione vengono trasferiti utilizzando una cifratura a 128 bit. In questo modo è estremamente complesso decifrare e leggere le trasmissioni decifrate.
- Il client può utilizzare le applicazioni X11¹ avviate dal prompt della shell. Questa tecnica, chiamata *X11 forwarding*, fornisce un'interfaccia grafica e sicura.

Dato che il protocollo SSH cifra tutto ciò che invia e riceve, può essere usato per cifrare dei protocolli che altrimenti non sarebbero sicuri. Se usate il *port forwarding*, potete utilizzare un server SSH per cifrare protocolli non sicuri, come POP, aumentando la sicurezza dei dati e del sistema in generale.

Red Hat Linux 8.0 include i pacchetti OpenSSH generico (`openssh`), server OpenSSH (`openssh-server`) e client (`openssh-clients`). Per istruzioni sull'installazione di OpenSSH, consultate il capitolo *OpenSSH* della *Official Red Hat Linux Customization Guide*. I pacchetti OpenSSH richiedono l'installazione del pacchetto OpenSSL (`openssl`). OpenSSL installa numerose librerie per la cifratura che consentono a OpenSSH di fornire comunicazioni cifrate.

Molti programmi client e server possono utilizzare il protocollo SSH. Esistono varie versioni del client SSH per quasi tutti i maggiori sistemi operativi in uso. Anche se gli utenti che si connettono al vostro sistema non possiedono Red Hat Linux, possono comunque trovare e utilizzare un client SSH nativo per il proprio sistema operativo.

9.1.1. Perché usare SSH?

Utenti con intenzioni malvagie dispongono di un'ampia varietà di strumenti per interrompere, intercettare e reindirizzare il traffico di rete allo scopo di ottenere l'accesso al vostro sistema. In generale queste minacce possono essere raggruppate in due categorie:

1. X11 si riferisce al sistema di visualizzazione a finestre X11R6, generalmente conosciuto come X. Red Hat Linux comprende **XFree86**, un sistema X Window Open Source molto diffuso e basato su X11R6.

- *Intercettazione delle comunicazioni tra due sistemi* — in questo scenario un malintenzionato può trovarsi in qualche punto della rete tra le due entità in comunicazione ed eseguire una copia delle informazioni trasmesse tra i due sistemi, per conservarle o inviarle modificate al destinatario originale.

Questo attacco può essere sventato attraverso l'utilizzo di una comune utility di rete per la rilevazione delle intrusioni.

- *Imitazione di un host particolare* — con questa strategia, il malintenzionato finge di essere il destinatario di un messaggio. Se la strategia funziona, il sistema dell'utente non si accorge dell'inganno.

Questo attacco può essere sventato attraverso tecniche note come DNS poisoning² o IP spoofing³.

Entrambe le tecniche descritte sopra consentono l'intercettazione di informazioni potenzialmente importanti e, se l'intrusione avviene a scopi malvagi, i risultati possono essere disastrosi.

Se SSH è usato per i login con la shell remota e per la copia dei file, le minacce alla sicurezza si riducono notevolmente. Questo perché il server utilizza le firme digitali per verificare l'identità degli utenti. Inoltre, tutte le comunicazioni tra i sistemi client e server sono cifrate. I tentativi di assumere l'identità di uno dei due sistemi comunicanti non funzioneranno, poiché ogni pacchetto è cifrato con un codice conosciuto solo dai sistemi locali e remoti.

9.2. Sequenza degli eventi di una connessione SSH

Una serie di eventi contribuisce a salvaguardare l'integrità di una comunicazione SSH tra due host.

Innanzitutto viene creato un livello di trasporto sicuro, in modo che il client sappia che sta comunicando con il server corretto. Poi viene cifrata la comunicazione tra il client e il server con l'uso di un cifratore simmetrico.

In seguito, il client si autentica in modo sicuro al server senza inviare informazioni in testo semplice.

Infine, quando il client si è autenticato al server tramite la connessione, è possibile utilizzare in modo sicuro i vari servizi usati, per esempio una sessione interattiva della shell, le applicazioni X11 e le porte TCP/IP immerse in un tunnel.

9.3. Livelli di sicurezza SSH

Il protocollo SSH consente a ogni programma client e server creato in base alle specifiche del protocollo di comunicare in modo sicuro e di essere utilizzato in maniera interscambiabile.

Attualmente esistono due diverse varietà di SSH. La versione 1 contiene diversi algoritmi di cifratura brevettati (anche se molti brevetti sono scaduti) e una "falla" nella sicurezza che potenzialmente permette di inserire informazioni nel flusso di dati. La suite OpenSSH in Red Hat Linux 8.0 utilizza la versione 2.0 di SSH di default, sebbene supporti anche la versione 1. Si consiglia di utilizzare, se possibile, server e client compatibili con la versione 2.

Entrambe le versioni 1 e 2 del protocollo SSH aggiungono livelli di sicurezza. Ogni livello offre il proprio tipo di protezione.

2. Il DNS poisoning si verifica quando un intruso ottiene l'accesso a un server DNS server, indirizzando i sistemi client a un host duplicato con intenti dannosi.

3. L'IP spoofing si verifica quando un intruso invia pacchetti di rete che sembrano provenire da un host fidato della rete.

9.3.1. Livello di trasporto

Lo scopo principale del livello di trasporto è quello di facilitare una comunicazione sicura tra due host al momento dell'autenticazione e subito dopo. Il livello di trasporto, che utilizza di solito il protocollo TCP/IP, cerca di raggiungere tale scopo cifrando e decifrando i dati, verificando che il server corrisponda al computer corretto per l'autenticazione e fornendo la protezione integrale di pacchetti di dati in fase di trasmissione e ricezione. Inoltre, il livello di trasporto può fornire la compressione dei dati, aumentando la velocità di trasferimento delle informazioni.

Quando un client SSH contatta un server, avviene uno scambio di informazioni in modo che i due sistemi possano creare correttamente il livello di trasporto. Durante questo scambio ecco cosa succede:

- Scambio delle chiavi
- Algoritmo della chiave pubblica
- Algoritmo della cifratura simmetrica
- Algoritmo per l'autenticazione del messaggio
- Algoritmo hash

Durante lo scambio delle chiavi il server si fa riconoscere dal client tramite una *chiave host*. Se il client non ha mai comunicato con questo particolare server, non è in grado di riconoscere la chiave del server e non verrà effettuata la connessione. OpenSSH aggira questo problema accettando la chiave host del server dopo che l'utente è stato notificato e verifica che venga accettata la nuova chiave dell'host. Nelle connessioni successive, la chiave host del server viene verificata con una versione salvata sul client, in modo che il client sia "sicuro" di comunicare con il server corretto. Se in futuro la chiave host non sarà più corrispondente, l'utente dovrà rimuovere la versione salvata nel client prima di effettuare una nuova connessione.



Attenzione

Un malintenzionato potrebbe "mascherarsi" da utente del server SSH durante il contatto iniziale, poiché il sistema non conosce necessariamente la differenza tra il server corretto e quello "mascherato". Per evitare che questo accada, dovrete verificare l'integrità del nuovo server SSH contattando l'amministratore del server prima di collegarvi per la prima volta o dopo la modifica della chiave host.

SSH è stato ideato per funzionare con quasi ogni tipo di algoritmo per le chiavi pubbliche o formato di codifica. Dopo lo scambio iniziale delle chiavi che genera un valore hash usato per gli scambi e un valore segreto condiviso, i due sistemi iniziano immediatamente a calcolare nuove chiavi e algoritmi per proteggere l'autenticazione e i dati futuri inviati tramite la connessione.

Dopo la trasmissione di una certa quantità di dati con l'utilizzo di una chiave e un algoritmo particolari (la quantità esatta dipende dall'implementazione di SSH), avviene un altro scambio di chiavi, che genera a sua volta un'ulteriore serie di valori hash e un nuovo valore segreto condiviso. In questo modo, anche se un hacker riuscisse a determinare tali valori, dovrebbe rideterminarli dopo ogni scambio delle chiavi.

9.3.2. Autenticazione

Dopo aver costruito un tunnel sicuro per inviare le informazioni da un sistema all'altro, il server indica al client i diversi metodi di autenticazione supportati, come per esempio una firma privata codificata o la digitazione di una password. Il client tenta, poi, di autenticarsi al server tramite uno dei metodi supportati.

Poiché i server possono essere configurati per autorizzare diversi tipi di autenticazione, questo metodo offre un ottimo controllo a entrambe le parti. Il server stabilisce i metodi di cifratura supportati

e il client può scegliere l'ordine dei metodi di autenticazione da utilizzare. Grazie alla sicurezza del livello di trasporto SSH, è possibile utilizzare senza problemi perfino metodi di autenticazione apparentemente non sicuri, come l'autenticazione basata sull'host.

La maggior parte degli utenti che richiede una shell sicura utilizza una password per l'autenticazione. Dato che la password viene cifrata durante lo spostamento sul livello di trasporto, può essere inviata in modo sicuro attraverso qualsiasi rete.

9.3.3. Connessione

Dopo un'autenticazione corretta su un livello di trasporto SSH, vengono aperti dei *canali* mediante il *multiplexing*⁴ della singola connessione tra i due sistemi. Ognuno di questi canali gestisce la comunicazione per una diversa sessione di terminale, le informazioni di X11 inviate o qualsiasi altro servizio che cerca di utilizzare la connessione SSH.

Entrambi i client e i server possono creare un nuovo canale. A ogni canale viene assegnato un numero diverso per ogni estremità della connessione. Quando il client tenta di aprire un nuovo canale, viene inviato, insieme alla richiesta, il suo numero per il canale. Questa informazione viene archiviata sul server e utilizzata per indirizzare una particolare comunicazione di servizio per il canale. In questo modo i diversi tipi di sessione non si disturbano a vicenda e i canali possono essere chiusi senza interrompere la connessione primaria SSH tra i due sistemi.

I canali supportano inoltre il *controllo del flusso*, che consente loro di inviare e ricevere i dati ordinatamente. In questo modo i dati non vengono inviati attraverso il canale finché il client non riceve un messaggio che lo avverte che il canale è in grado di ricevere.

Il client e il server negoziano le caratteristiche di ogni canale automaticamente, a seconda del tipo di servizio che il client richiede e del tipo di connessione di rete che l'utente usa. I diversi tipi di connessioni remote possono essere gestiti in modo flessibile senza dover modificare l'infrastruttura di base del protocollo.

9.4. File di configurazione OpenSSH

OpenSSH possiede due diversi tipi di file di configurazione, uno per i programmi client (`ssh`, `scp` e `sftp`) e l'altro per il demone del server (`sshd`).

Le informazioni di configurazione SSH sono memorizzate nella directory `/etc/ssh/`:

- `moduli` — contiene gruppi Diffie-Hellman usati per lo scambio delle chiavi. In sostanza, questo consente di creare un livello di trasporto sicuro. Quando le chiavi sono scambiate all'inizio di una sessione SSH, viene creato un valore segreto e condiviso che non può essere determinato da una singola parte e viene usato per fornire l'autenticazione dell'host.
- `ssh_config` — il file di configurazione del client SSH. Viene sovrascritto se anche nella home directory dell'utente (`~/.ssh/config`) ne è presente uno.
- `sshd_config` — il file di configurazione per il demone `sshd`.
- `ssh_host_dsa_key` — la chiave privata DSA utilizzata dal demone `sshd`.
- `ssh_host_dsa_key.pub` — la chiave pubblica DSA usata dal demone `sshd`.
- `ssh_host_key` — la chiave privata RSA usata dal demone `sshd` per la versione 1 del protocollo SSH.

4. Una connessione multiplexed è costituita da diversi segnali inviati tramite un supporto condiviso. Con SSH, canali differenti vengono inviati tramite una connessione comune sicura.

- `ssh_host_key.pub` — la chiave pubblica RSA usata dal demone `sshd` per la versione 1 del protocollo SSH.
- `ssh_host_rsa_key` — la chiave privata RSA usata dal demone `sshd` per la versione 2 del protocollo SSH.
- `ssh_host_rsa_key.pub` — la chiave pubblica RSA usata dal demone `sshd` per la versione 2 del protocollo SSH.

Le informazioni di configurazione SSH specifiche per l'utente sono archiviate nella sua directory home all'interno di `~/.ssh/`:

- `authorized_keys` — il file che contiene un elenco delle chiavi pubbliche "autorizzate" per i server. Se un utente che si sta connettendo può provare di conoscere la chiave privata corrispondente a una delle chiavi pubbliche, viene autenticato. Questo, però, è un metodo di autenticazione opzionale.
- `id_dsa` — contiene l'identità di autenticazione DSA dell'utente.
- `id_dsa.pub` — la chiave DSA pubblica dell'utente.
- `id_rsa` — la chiave RSA pubblica usata da `sshd` per la versione 2 del protocollo SSH.
- `identity` — la chiave RSA privata usata da `sshd` per la versione 1 del protocollo SSH.
- `known_hosts` — contiene le chiavi host DSA dei server a cui l'utente si collega tramite SSH. Questo file è importante per garantire che il client SSH si colleghi al server SSH corretto. Se una chiave host viene modificata, e non ne conoscete la ragione, dovrete contattare l'amministratore di sistema del server SSH per assicurarvi che non sia stato compromesso. Se un server possiede delle chiavi host modificate in modo legale mediante la reinstallazione di Red Hat Linux, al successivo collegamento al server l'utente verrà avvertito che la chiave host memorizzata nel file `known_hosts` non corrisponde. Per collegarsi al server, l'utente deve aprire il file `known_hosts` con un editor di testo e cancellare la chiave host. In questo modo il client SSH potrà creare una nuova chiave host.

Per informazioni sulle diverse direttive disponibili nei file di configurazione SSH, consultate le pagine man per `ssh` e `sshd`.

9.5. Shell più che sicura

Un'interfaccia a linea di comando sicura è solo uno dei tanti modi in cui è possibile usare una SSH. Considerata la larghezza di banda, le sessioni X11 possono essere indirizzate tramite un canale SSH. Oppure, utilizzando il TCP/IP forwarding, le connessioni di porta un tempo non sicure tra sistemi diversi possono essere mappate su canali SSH specifici.

9.5.1. X11 forwarding

Aprire una sessione X11 tramite una connessione SSH stabilita è facile quanto avviare un programma X su un computer locale. Quando un programma X viene eseguito dalla shell, il client e il server SSH creano un nuovo canale sicuro e i dati del programma X vengono inviati tramite questo canale al vostro client.

X11 forwarding può essere molto utile. Per esempio, potete usarlo per creare una sessione interattiva e sicura con l'interfaccia grafica utente `up2date` sul server per aggiornare i pacchetti. Per farlo, connettetevi al server utilizzando `ssh` e digitate:

```
up2date &
```

Vi viene richiesto di inserire la password di root per il server. Compare il **Red Hat Update Agent** e potete così aggiornare i vostri pacchetti sul server proprio come se foste seduti davanti al computer.

9.5.2. Port forwarding

Con SSH potete rendere sicuri i protocolli TCP/IP altrimenti insicuri mediante il port forwarding. Con questa tecnica, il server SSH diventa un passaggio cifrato al client SSH.

Il port forwarding consiste nel mappare una porta locale sul client verso una porta remota sul server. SSH vi permette di mappare qualsiasi porta dal server verso qualsiasi porta sul client. I numeri delle porte non devono essere uguali per funzionare.

Per creare un canale TCP/IP di port forwarding che attenda le connessioni sull'host locale, utilizzate il seguente comando:

```
ssh -L porta-locale:nomehost-remoto:porta-remota nomeutente@nomehost
```



Nota Bene

Per impostare il port forwarding e avviare servizi su porte inferiori alla 1024 è necessario l'accesso di root.

Se per esempio desiderate controllare la vostra posta su un server chiamato mail.domain.com utilizzando il protocollo POP tramite una connessione cifrata, potete usare il comando seguente:

```
ssh -L 1100:mail.domain.com:110 mail.domain.com
```

Una volta impostato il canale per il port forwarding tra i due computer, potete indicare al vostro client di posta POP di usare la porta 1100 sull'host locale per controllare l'arrivo di nuova posta. Qualsiasi richiesta inviata alla porta 1100 sul vostro sistema sarà indirizzata in modo sicuro al server mail.domain.com.

Se sul server mail.domain.com non è in esecuzione un demone server SSH, ma potete collegarvi tramite SSH a un computer della stessa rete, potete ancora utilizzare SSH per rendere sicura quella parte della connessione POP. È tuttavia necessario un comando leggermente diverso:

```
ssh -L 1100:mail.domain.com:110 other.domain.com
```

In questo esempio, inviate la vostra richiesta POP dalla porta 1100 sul vostro computer, tramite la connessione SSH sulla porta 22 a other.domain.com. A questo punto, other.domain.com si connette alla porta 1100 su mail.domain.com per permettervi di controllare l'arrivo di nuova posta. Con questa tecnica, solo la connessione tra il vostro sistema e other.domain.com è sicura.

Il port forwarding può risultare particolarmente utile per ricevere informazioni in modo sicuro tramite i firewall di rete. Se il firewall è configurato per consentire il traffico SSH tramite la porta standard (22), ma blocca l'accesso alle altre porte, una connessione tra due host che usano porte bloccate è comunque possibile se si reindirizza la comunicazione tramite una connessione SSH.



Nota Bene

L'utilizzo del port forwarding per inoltrare connessioni in questo modo consente a qualsiasi utente sul sistema client di connettersi al servizio a cui state inviando le connessioni. Se il sistema client viene compromesso, anche le persone non autorizzate potranno accedere ai servizi inoltrati.

Gli amministratori di sistema che si occupano del port forwarding possono disabilitare questa funzione sul server specificando il parametro `No` per la riga `AllowTcpForwarding` nel file `/etc/ssh/sshd_config` e riavviando il servizio `sshd`.

9.6. Richiesta di SSH per le connessioni remote

Per rendere SSH davvero efficace nel proteggere le vostre connessioni di rete, è necessario smettere di utilizzare tutti i protocolli di connessione non sicuri, come `telnet` e `rsh`. Se questo non avviene, una password protetta tramite `ssh` può essere individuata alla prima connessione via `telnet`.

Per disabilitare i metodi di connessione poco sicuri, utilizzate il comando `chkconfig`, il programma basato su `ncurses` `ntsysv` o l'applicazione grafica **Services Configuration Tool**. Per utilizzare questi strumenti dovete collegarvi come `root`.

Alcuni servizi da disabilitare includono:

- `telnet`
- `rsh`
- `ftp`
- `rlogin`
- `wu-ftp`
- `vsftpd`

Per maggiori informazioni sui runlevel e sulla configurazione di servizi con `chkconfig`, `ntsysv` e **Services Configuration Tool**, fate riferimento al capitolo relativo al *controllo dell'accesso ai servizi* nella *Official Red Hat Linux Customization Guide*.

Kerberos

Kerberos è un protocollo di autenticazione dei servizi di rete creato dal MIT che si serve della crittografia a chiave segreta — evitando così la necessità di inviare password attraverso la rete. Autenticare mediante Kerberos impedisce agli utenti non autorizzati di intercettare le password inviate attraverso la rete.

10.1. Vantaggi di Kerberos

La maggior parte dei sistemi di rete usa uno schema di autenticazione basato sulle password. Quando un utente si autentica per accedere a un server di rete deve fornire un user name e una password per tutti i servizi che richiedono l'autenticazione. Tali informazioni vengono inviate via rete e in questo modo il server verifica l'identità dell'utente.

Tuttavia, la trasmissione delle informazioni di autenticazione per molti servizi avviene in chiaro. Qualunque utente che ha accesso alla rete e che può utilizzare un analizzatore di pacchetti di rete (solitamente chiamato packet sniffer) può intercettare le password che attraversano la rete.

Lo scopo principale di Kerberos è quello di eliminare la trasmissione delle informazioni di autenticazione attraverso la rete. Il corretto utilizzo di Kerberos vi permette di ridurre drasticamente la possibilità che dei packet sniffer si intercettino tali informazioni.

10.2. Svantaggi di Kerberos

Tramite Kerberos si riesce a proteggere la rete dagli attacchi più comuni. Tuttavia, potrebbe risultare complesso da implementare, per varie ragioni:

- Non esiste alcun meccanismo automatico che consenta di migrare rapidamente le password dal database delle password di UNIX (per esempio `/etc/passwd` o `/etc/shadow`) a quello di Kerberos. Per maggiori informazioni a riguardo, consultate la domanda numero 2.23 nella sezione FAQ di Kerberos all'indirizzo <http://www.nrl.navy.mil/CCS/people/kenh/kerberos-faq.html>.
- Kerberos è solo in parte compatibile con il sistema PAM (Pluggable Authentication Modules) usato dalla maggior parte dei server Red Hat Linux. Per maggiori informazioni, consultate la Sezione 10.5.
- Affinché una applicazione di rete possa usare Kerberos, è necessario modificare il suo codice sorgente per effettuare le chiamate alle librerie Kerberos. Per altre applicazioni, occorre modificare il protocollo di comunicazione utilizzato tra server e client e ciò comporta molta programmazione. Le applicazioni "a sorgente chiusa" che non supportano Kerberos di default risultano quelle più problematiche.
- Kerberos parte dal presupposto che stiate usando host fidati su una rete non sicura. Il suo obiettivo principale è di impedire che le password in solo testo vengano inviate lungo questa rete. Tuttavia, se qualcuno diverso dall'utente effettivo ha accesso fisico a uno degli host, specialmente quello che emette i ticket usati per l'autenticazione, l'intero sistema di autenticazione è a rischio.
- Kerberos è una soluzione che non prevede vie di mezzo. Se decidete di utilizzare Kerberos sulla rete, dovete ricordarvi tutte le password trasferite a un servizio che non lo supporta, poiché l'autenticazione rischia di essere intercettata dai packet sniffer. Per proteggere la vostra rete con Kerberos dovete "kerberizzare" tutte le applicazioni che inviano password in testo oppure evitate del tutto di utilizzarle.

10.3. Terminologia di Kerberos

Come ogni altro sistema, anche Kerberos ha una propria terminologia per definire i vari aspetti del servizio. Prima di descriverne il funzionamento, vi elenchiamo i termini utilizzati:

ciphertext

Dati cifrati.

in testo

Dati non cifrati, leggibili.

client

Un'entità sulla rete (un utente, un host o un'applicazione) che riceve un ticket da Kerberos.

credential cache o file dei ticket

Un file che contiene le chiavi per la comunicazione cifrata fra un utente e vari servizi di rete. Kerberos 5 fornisce l'architettura per altri tipi di cache (come per esempio la memoria condivisa), ma i file sono supportati meglio.

crypt hash

Un hash a senso unico per l'autenticazione degli utenti. Pur essendo più sicuro del testo, è comunque piuttosto semplice da decifrare per i cracker più esperti.

chiave

Dati usati per cifrare e decifrare le informazioni. Le informazioni cifrate non possono essere decifrate senza la chiave corretta o senza un fiuto infallibile.

Key Distribution Center (KDC)

Un servizio che distribuisce i ticket Kerberos, eseguito di solito sullo stesso host del Ticket Granting Server.

Tabella delle chiavi o keytab

Un file che contiene un elenco non cifrato delle chiavi. I server recuperano le chiavi di cui necessitano dai file keytab invece di utilizzare il comando `kinit`. Il file keytab di default è `/etc/krb5.keytab`. Il comando `/usr/kerberos/sbin/kadmind` è l'unico servizio che usa altri file (normalmente il file `/var/kerberos/krb5kdc/kadm5.keytab`).

principal

Un utente o un servizio che si possono autenticare tramite Kerberos. Il nome di un principal ha la seguente forma: `root[/instance]@REALM`. Per un utente standard, il root è lo stesso dell'ID di login. `instance` è opzionale. Se il principal presenta un'istanza, è separato da root con `/`. La stringa vuota è una istanza valida (che differisce dall'istanza di default `NULLA`), ma il suo utilizzo può portare confusione. Tutti i principal hanno la propria chiave, derivata dalle loro password (per gli utenti) o da un insieme casuale (per i servizi).

realm

Una rete basata su Kerberos, formata da uno o più server (chiamati anche KDC) e da un insieme più o meno grande di client.

servizio

Un programma accessibile via rete.

ticket

Una serie di credenziali elettroniche temporanee che verificano l'identità di un client per un particolare servizio.

Ticket Granting Service (TGS)

I ticket distribuiti dal TGS permettono all'utente di accedere a un particolare servizio. Solitamente il TGS viene eseguito sullo stesso host del KDC.

Ticket Granting Ticket (TGT)

Un ticket speciale che permette al client di ottenere ulteriori ticket senza richiederli al KDC.

10.4. Funzionamento di Kerberos

Conclusa la spiegazione della terminologia di Kerberos, vi presentiamo ora una breve panoramica su come funziona il sistema di autenticazione Kerberos.

Al posto di attuare un'autenticazione tra ciascuna ogni macchina client e ciascun server, Kerberos utilizza la cifratura simmetrica e un sistema fidato di terze parti — noto come Key Distribution Center o KDC — per autenticare gli utenti su una rete e consentire loro di accedere ai servizi desiderati. Avvenuta l'autenticazione, Kerberos immagazzina un ticket specifico per quella sessione sul computer dell'utente e tutti i servizi "kerberizzati", per effettuare l'autenticazione, invece di chiedere la password all'utente cercheranno direttamente quel ticket.

Quando un utente si collega alla propria workstation collegata a una rete Kerberos, il suo principal viene inviato al KDC sotto forma di richiesta TGT. Questa richiesta può essere inviata dal programma di login (in modo trasparente) o dal programma `kinit` una volta che l'utente si è collegato.

Il KDC controlla il principal nel suo database. Se viene trovato, crea un TGT, lo cifra usando la chiave dell'utente e lo invia come risposta all'utente.

Il programma di login sulla macchina client o `kinit` decifra il TGT utilizzando la chiave (che ricava dalla password dell'utente). La data di "scadenza" viene impostata in modo che un TGT compromesso possa essere utilizzato solo per un periodo limitato (che generalmente è di otto ore); questo è un metodo più sicuro perché la password non viene mai inviata attraverso la rete. In seguito, non occorre che l'utente reinserta la password se non quando scade il TGT o quando si ricollega.

Quando un utente deve accedere a un servizio di rete, il client utilizza il TGT per richiedere un ticket per il servizio al Ticket Granting Service (TGS), in esecuzione sul KDC. Il TGS rilascia un ticket che viene usato per autenticare l'utente.



Attenzione

Il sistema Kerberos può risultare compromesso ogni volta che l'autenticazione di un utente per un servizio che non utilizza Kerberos avviene per mezzo dell'invio di una password di testo. Pertanto, non è consigliato utilizzare servizi che non prevedono Kerberos, come per esempio **telnet**. È invece accettabile l'utilizzo di protocolli sicuri, come OpenSSH o SSL.

Chiaramente, la spiegazione sopra riportata non è che una panoramica sul funzionamento del sistema Kerberos di autenticazione dei servizi di rete. Se desiderate approfondire l'argomento, consultate la Sezione 10.8.



Nota Bene

Kerberos dipende da alcuni servizi di rete per poter funzionare correttamente. Prima di tutto è necessario che gli orologi dei vari calcolatori siano sincronizzati. Si consiglia, pertanto, di impostare un programma di sincronizzazione per la rete, come per esempio `ntp`.

Inoltre, alcuni aspetti di Kerberos si basano sul servizio DNS (Domain Name Service), perciò accertatevi che il DNS sia configurato in modo corretto. Per maggiori informazioni, potete consultare la Kerberos V5 System presente in formato PostScript e HTML nella directory `/usr/share/doc/krb5-server-numero-versione`, (dove `numero-versione` è la versione installata sul sistema).

10.5. Kerberos e PAM

Attualmente, i servizi "kerberizzati" non utilizzano i moduli di autenticazione pluggable (PAM) — i server "kerberizzati" evitano completamente il sistema PAM. Le applicazioni che usano PAM possono comunque utilizzare il sistema Kerberos per l'autenticazione se il modulo `pam_krb5` (fornito nel pacchetto `pam_krb5`) è installato. Il pacchetto `pam_krb5` contiene dei file d'esempio per la configurazione che consente a servizi quali `login` e `gdm` di autenticare gli utenti e ricavare le credenziali iniziali dalla loro password. Se l'accesso al server di rete è effettuato sempre tramite servizi kerberizzati (o servizi che utilizzano GSS-API, come IMAP), la rete può essere considerata sufficientemente sicura.

Un amministratore di sistema esperto sa che non bisogna aggiungere il controllo delle password per i servizi di rete, poiché la maggior parte dei protocolli usati da questi servizi non cifrano la password prima di inviarla lungo la rete.

La prossima sezione spiega come configurare un server Kerberos di base.

10.6. Configurazione di un server Kerberos 5

Nel configurare Kerberos, configurate prima il server. Se avete necessità di configurare dei server slave, troverete tutti i dettagli su come impostare i rapporti tra server master e server slave nella *Kerberos 5 Installation Guide* (all'interno della directory `/usr/share/doc/krb5-server-<numero-versione>`).

Per configurare un server Kerberos di base, eseguite quanto segue:

1. Assicuratevi che il vostro server abbia a disposizione un sincronizzatore e un DNS prima di configurare Kerberos 5. Prestate particolare attenzione alla sincronizzazione tra il server Kerberos e i vari client. Se gli orologi del server e del client differiscono di oltre cinque minuti (è possibile impostare un intervallo di tempo predefinito), i client Kerberos non potranno essere autenticati. La sincronizzazione degli orologi è indispensabile per impedire a eventuali malintenzionati di utilizzare un vecchio ticket di Kerberos per farsi passare come utente valido.

Si consiglia di impostare una rete client/server compatibile con Network Time Protocol (NTP) per Red Hat Linux, anche se non si sta utilizzando Kerberos. Red Hat Linux 8.0 comprende il pacchetto `ntp`, per un'installazione agevolata. Per maggiori informazioni su NTP, visitate la pagina Web <http://www.eecis.udel.edu/~ntp>.

2. Installate i pacchetti `krb5-libs`, `krb5-server` e `krb5-workstation` sulla macchina su cui verrà eseguito il vostro KDC. Tale macchina deve essere molto protetta — se possibile, non dovrebbe eseguire altri servizi al di fuori di KDC.

Se desiderate utilizzare un'interfaccia grafica per gestire Kerberos, dovete installare anche il pacchetto `gnome-kerberos`, contenente `krb5`, un tool grafico per la gestione di ticket.

3. Modificate i file di configurazione `/etc/krb5.conf` e `/var/kerberos/krb5kdc/kdc.conf` per riflettere il nome del realm e le mappature del dominio al realm. Si può costruire un realm semplice sostituendo istanze di `EXAMPLE.COM` e `example.com` con il vostro nome di dominio — facendo attenzione che le maiuscole e le minuscole siano corrette — cambiando il KDC da `kerberos.example.com` al nome del vostro server Kerberos. Normalmente, tutti i nomi di realm sono scritti in lettere maiuscole mentre tutti gli hostname DNS e i nomi di dominio sono in formato minuscolo. Per maggiori dettagli sul formato di questi file consultate le rispettive pagine man.

4. Create il database tramite l'utility `kdb5_util` al prompt della shell:

```
/usr/kerberos/sbin/kdb5_util create -s
```

Il comando `create` crea il database che sarà utilizzato per immagazzinare le chiavi per i realm Kerberos. `-s` impone la creazione di un file *stash* che conterrà la chiave del server master. Se non esiste alcun file *stash* da cui reperire la chiave, a ogni avvio il server Kerberos (`krb5kdc`) richiederà all'utente la password del server master (utilizzabile per rigenerare la chiave).

5. Modificate il file `/var/kerberos/krb5kdc/kadm5.acl`. È utilizzato da `kadmin` per stabilire quali principal hanno accesso al database di Kerberos nonché il loro livello di accesso. Generalmente, sarà sufficiente una sola linea:

```
*/admin@EXAMPLE.COM *
```

La maggior parte degli utenti sarà rappresentata, nel database, da un solo principal (con un'istanza `NULL`, o una vuota, come per esempio `joe@EXAMPLE.COM`). Con questa configurazione gli utenti che hanno un secondo principal con un'istanza di `admin` (per esempio, `joe/admin@EXAMPLE.COM`) avranno pieni poteri sul database del realm di Kerberos.

Una volta che `kadmin` viene avviato sul server, qualunque utente può accedere ai suoi servizi eseguendo `kadmin` o qualunque client o server nel realm. Tuttavia, solo gli utenti elencati nel file `kadm5.acl` avranno il permesso di modificare a loro piacimento il database (ma non potranno cambiare le proprie password).



Nota Bene

L'utility `kadmin` comunica con il server `kadmin` via rete e utilizza Kerberos per gestire l'autenticazione. Ovviamente, è necessario creare il primo principal prima di potersi connettere al server via rete per amministrarlo. Create il primo principal usando il comando `kadmin.local`, che è concepito proprio per essere utilizzato sullo stesso host di KDC e non si serve di Kerberos per l'autenticazione.

Per creare il primo principal, digitate il seguente comando `kadmin.local` al terminale di KDC:

```
/usr/kerberos/sbin/kadmin.local -q "addprinc nome-utente/admin"
```

6. Avviate Kerberos mediante i comandi seguenti:

```
/sbin/service krb5kdc start
/sbin/service kadmin start
/sbin/service krb524 start
```

7. Aggiungete altri principal per gli utenti aggiungendo il comando `addprinc` con `kadmin.kadmin` e `kadmin.local` sul KDC master sono interfacce a linea di comando per il sistema di amministrazione di Kerberos. Dopo aver lanciato il programma `kadmin` si rendono disponibili molti comandi. Per maggiori informazioni, consultate la pagina man di `kadmin`.

8. Verificate che il vostro server rilasci ticket. Anzitutto, eseguite `kinit` per ottenere un ticket e immagazzinatelo in un credential cache (o file dei ticket). Successivamente utilizzate `klist` per visualizzare l'elenco di credenziali nella vostra cache e usate `kdestroy` per eliminare la cache e le credenziali in essa contenute.



Nota Bene

Per default, `kinit` cerca di effettuare l'autenticazione utilizzando lo username di login dell'account utilizzato per il primo collegamento al sistema (non al server Kerberos). Se lo username di quel sistema non corrisponde al principal contenuto nel database di Kerberos, riceverete un messaggio di errore. In tal caso, assegnate a `kinit` il nome del vostro principal come argomento per la linea di comando (`kinit principal`).

Una volta completate le operazioni sopra descritte, il server Kerberos dovrebbe essere attivo e funzionante. A questo punto, dovete configurare un client.

10.7. Configurazione di un client Kerberos 5

La configurazione di un client Kerberos 5 è meno impegnativa rispetto a quella del server. Dovete, come minimo, installare i pacchetti client e fornire a ciascun client un file di configurazione `krb5.conf` valido. Le versioni di `rsh` e `rlogin` che utilizzano Kerberos necessitano di alcune modifiche di configurazione.

1. Assicuratevi che il client kerberos e il KDC siano sincronizzati. Per maggiori informazioni in merito, consultate la Sezione 10.6. Accertatevi, inoltre, che il DNS funzioni correttamente sul client Kerberos prima di configurare i programmi relativi al client.

2. Installate i pacchetti `krb5-libs` e `krb5-workstation` su tutti i client delle vostre macchine. Dovete anche fornire una versione di `/etc/krb5.conf` per ciascun client; in genere, si può utilizzare lo stesso `krb5.conf` usato dal KDC.

3. Prima che una postazione di lavoro presente nel realm possa consentire agli utenti di connettersi utilizzando versioni Kerberos di `rsh` e `rlogin`, occorre installarvi il pacchetto `xinetd` ed è necessario che il principal del suo host sia incluso nel database di Kerberos. Anche per i programmi `kshd` e `klogin` sarà necessario l'accesso alle chiavi per i principal dei loro servizi.

Utilizzando `kadmin`, aggiungete un principal host alla postazione di lavoro sul KDC. L'istanza, in questo caso, sarà l'hostname della postazione. Potete utilizzare l'opzione `-randkey` per il comando di `kadmin` `addprinc sul KDC` per creare il principal e assegnargli una chiave casuale:

```
addprinc -randkey host/blah.example.com
```

Una volta creato il principal, potete estrarre le chiavi per la postazione di lavoro eseguendo `kadmin` direttamente sulla postazione e utilizzare il comando `ktadd` all'interno di `kadmin`:

```
ktadd -k /etc/krb5.keytab host/blah.example.com
```

Per utilizzare la versione con Kerberos di `rsh` e `rlogin`, dovete abilitare `klogin`, `eklogin` e `kshell`.¹

4. Sarà necessario avviare altri servizi di rete Kerberos. Per utilizzare la versione "kerberizzata" di `telnet`, dovete attivare `krb5-telnet`.

Per l'accesso FTP, create ed estraete una chiave per un principal con una root di ftp, con l'istanza impostata sull'hostname del server FTP. Successivamente, attivate `gssftp`.

Il server IMAP incluso nel pacchetto `imap` utilizza l'autenticazione GSS-API con Kerberos 5 se trova la chiave corretta all'interno di `/etc/krb5.keytab`. La root per il principal dovrebbe essere `imap`. Il server CVS usa un principal con una root di `cv`s ed è identico a `pserver`.

1. Per sapere come abilitare i servizi, consultate il capitolo *Controllare l'accesso ai servizi* nella *Official Red Hat Linux Customization Guide*.

10.8. Risorse aggiuntive

Per maggiori informazioni su Kerberos, consultate le seguenti fonti.

10.8.1. Documentazione installata

- `/usr/share/doc/krb5-server-<version-number>` — la *Kerberos V5 Installation Guide* e la *Kerberos V5 System Administrator's Guide*, disponibili nei formati PostScript e HTML. È necessario installare il pacchetto RPM `krb5-server`.
- `/usr/share/doc/krb5-workstation-<version-number>` — La *Kerberos V5 UNIX User's Guide*, nei formati PostScript e HTML. Vi occorre il pacchetto RPM `krb5-workstation`.

10.8.2. Siti Web utili

- <http://web.mit.edu/kerberos/www> — home page di *Kerberos: The Network Authentication Protocol* sul sito del MIT.
- <http://www.nrl.navy.mil/CCS/people/kenh/kerberos-faq.html> — sezione FAQ (Frequently Asked Questions) di Kerberos.
- <ftp://athena-dist.mit.edu/pub/kerberos/doc/usenix.PS> — link a una versione in Postscript del libro intitolato *Kerberos: An Authentication Service for Open Network Systems* di Jennifer G. Steiner, Clifford Neuman e Jeffrey I. Schiller. Si tratta della prima documentazione prodotta su Kerberos.
- <http://web.mit.edu/kerberos/www/dialogue.html> — *Designing an Authentication System: a Dialogue in Four Scenes* creato da Bill Bryant nel 1988, modificato da Theodore Ts'o nel 1997. Racconta di una conversazione tra due programmatori che stanno progettando di creare un sistema di autenticazione Kerberos. Lo stile informale della discussione agevola coloro che non conoscono assolutamente Kerberos.
- <http://www.ornl.gov/~jar/HowToKerb.html> — *How to Kerberize your Site*.
- <http://www.networkcomputing.com/netdesign/kerb1.html> — *Kerberos Network Design Manual* offre una panoramica approfondita sul sistema Kerberos.

Tripwire

Il software di garanzia dell'integrità dei dati di Tripwire tiene sotto controllo i file e delle directory di sistema più importanti rilevando tutte le modifiche effettuate a tali file. Le opzioni di configurazione di Tripwire comprendono la possibilità di inviare messaggi di avvertimento via e-mail nel caso in cui vengano modificati dei file particolari e di effettuare controlli sull'integrità attraverso un processo cron. L'uso di Tripwire per rilevare le intrusioni e stabilire i danni può essere utile per aiutarvi a individuare le modifiche apportate al sistema. Poiché Tripwire è in grado di individuare i file che sono stati aggiunti, modificati o rimossi, può rendere più veloce il ripristino dopo l'intrusione, riducendo al minimo il numero di file da ripristinare.

Tripwire confronta i file e le directory con un database fondamentale contenente le posizioni dei file, le date modificate e altri dati. Il database contiene linee di base, ovvero delle "fotografie" di file e directory specifici in un determinato momento. Il contenuto del database della linea di base dovrebbe essere creato prima che si presenti il rischio di un'intrusione nel sistema. Dopo aver creato il database, Tripwire paragona il sistema attuale con il database e segnala tutte le modifiche, le aggiunte o le cancellazioni avvenute.



Warning

Benché sia un valido tool per il controllo dello stato di sicurezza di Red Hat Linux, Tripwire non è supportato da Red Hat, Inc. Per maggiori informazioni su Tripwire, consultate il sito del progetto (<http://www.tripwire.org>).

11.1. Come usare Tripwire

Il seguente diagramma di flusso illustra il funzionamento di Tripwire:

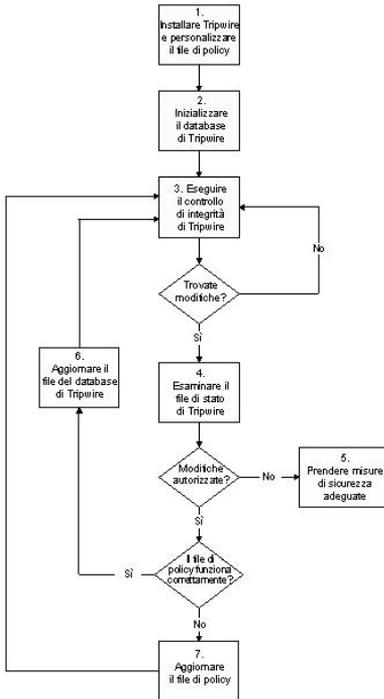


Figura 11-1. Utilizzo di Tripwire

Di seguito trovate una descrizione più dettagliata dei blocchi numerati mostrati nella Figura 11-1:

1. Installazione di Tripwire e personalizzazione del file di policy

Installate l'RPM di tripwire (la Sezione 11.2). In seguito personalizzate i file di configurazione e i file di policy (rispettivamente, `/etc/tripwire/twcfg.txt` e `/etc/tripwire/twpol.txt`) e poi eseguite quindi lo script di configurazione (`/etc/tripwire/twinstall.sh`). Per maggiori informazioni, consultate la Sezione 11.3.

2. Inizializzazione del database di Tripwire

Create un database contenente i file di sistema più importanti e basato sui contenuti del nuovo file di policy di Tripwire (`/etc/tripwire/tw.pol`). Per maggiori informazioni, consultate la Sezione 11.4.

3. Controllo dell'integrità di Tripwire

Confrontate il database appena creato con i file di sistema attuali e cercate i file mancanti o modificati. Per maggiori informazioni, consultate la Sezione 11.5.

4. Controllo del file di report di Tripwire

Visualizzate il file report di Tripwire utilizzando `/usr/sbin/twprint` per evidenziare violazioni dell'integrità. Per maggiori informazioni, consultate la Sezione 11.6.1.

5. In caso di violazioni dell'integrità non autorizzate, adottare misure di sicurezza idonee

Se i file controllati sono stati modificati impropriamente, potete sostituire gli originali con i backup, reinstallare il programma oppure reinstallare l'intero sistema operativo.

6. Se le modifiche al file risultano valide, verificare e aggiornare il file database di Tripwire

Se le modifiche apportate ai file controllati sono state fatte intenzionalmente, modificate il file database in modo che tali modifiche non vengano segnalate come violazioni. Per maggiori informazioni, consultate la Sezione 11.7.

7. Se il file di policy fallisce la verifica, aggiornare file di policy di Tripwire

Per modificare l'elenco dei file da controllare o il modo in cui gestire le violazioni dell'integrità, aggiornate il file di policy (`/etc/tripwire/twpol.txt`), ricreare una copia firmata (`/etc/tripwire/tw.pol`) e aggiornare il database. Per maggiori informazioni, consultate la Sezione 11.8.

Per informazioni più dettagliate, fate riferimento alle relative sezioni contenute in questo capitolo.

11.2. Installazione dell'RPM di Tripwire

Il modo più semplice per installare Tripwire è quello di selezionare l'RPM di `tripwire` durante il processo di installazione di Red Hat Linux 8.0. Tuttavia, se avete già installato Red Hat Linux 8.0, potete usare `rpm` o **Package Management Tool** (`redhat-config-packages`) per installare l'RPM di Tripwire dai CD-ROM di Red Hat Linux 8.0.

Se non siete sicuri di aver installato Tripwire, digitate il comando seguente a un prompt della shell:

```
rpm -q tripwire
```

Se Tripwire è installato, questo comando produrrà il risultato seguente:

```
tripwire-<numero-versione>
```

Le operazioni descritte qui di seguito indicano come trovare e installare Tripwire dal CD-ROM utilizzando l'applicazione a linea di comando di RPM:

1. Inserite il *CD 2* dei CD-ROM di installazione di Red Hat Linux 8.0.
2. Se il CD-ROM non esegue il montaggio automaticamente, digitate il comando seguente:

```
mount /mnt/cdrom
```

3. Verificate che l'RPM di `tripwire` sia sul CD-ROM digitando:

```
ls /mnt/cdrom/RedHat/RPMS/ | grep tripwire
```

Se l'RPM è presente, il comando visualizza il nome del pacchetto.

Se l'RPM *non* è presente, ricompare il prompt della shell. In tal caso, dovrete controllare il *CD 3* e magari anche il *CD 1* dei CD-ROM di installazione di Red Hat Linux 8.0, dapprima smontando il CD-ROM e poi ripetendo le operazioni da 1 a 3.

Per smontare il CD-ROM fate doppio clic sull'icona del CD-ROM e selezionate **Eject** oppure potete digitare il comando seguente al prompt della shell:

```
umount /mnt/cdrom
```

4. Una volta individuato l'RPM di `tripwire`, installatelo collegandovi come root e digitando il comando:

```
rpm -Uvh /mnt/cdrom/RedHat/RPMS/tripwire*.rpm
```

Nella directory `/usr/share/doc/tripwire-<numero-versione>/`, troverete le release notes e i file README relativi a Tripwire. Questi documenti contengono importanti informazioni sul file di policy predefinito e su altri argomenti.

11.3. Personalizzazione di Tripwire

Dopo aver installato l'RPM di Tripwire, dovete eseguire le operazioni qui di seguito descritte per inizializzare il:

11.3.1. Modifica di `/etc/tripwire/twcfg.txt`

Sebbene non vi occorra modificare questo file sample di configurazione di Tripwire, potrebbe comunque rendersi necessario in base alle vostre esigenze. Per esempio, potreste voler modificare i file di posizione di Tripwire, personalizzare le impostazioni di posta elettronica o decidere la quantità di dettagli da inserire nei report.

Qui di seguito è riportato un elenco delle variabili configurabili dall'utente *necessarie*, contenute nel file `/etc/tripwire/twcfg.txt`:

- `POLFILE` — specifica la posizione del file di policy; il parametro di default è `/etc/tripwire/tw.pol`.
- `DBFILE` — specifica la posizione del file database; il parametro predefinito è `/var/lib/tripwire/$(HOSTNAME).twr`
- `REPORTFILE` — specifica la posizione dei file report. Il parametro di default è `/var/lib/tripwire/report/$(HOSTNAME)-$(DATE).twr`.
- `SITEKEYFILE` — specifica la posizione del file con le chiavi del sito; il valore predefinito è `/etc/tripwire/site.key`.
- `LOCALKEYFILE` — specifica la posizione del file con le chiavi locali; il valore predefinito è `/etc/tripwire/$(HOSTNAME)-local.key`.



Importante

Se modificate il file di configurazione e non definite le variabili descritte sopra, il file non viene considerato valido da Tripwire. Questo provoca un errore nell'esecuzione di `tripwire` e siete costretti a uscire dal programma.

Le restatnti variabili di configurazione contenute nel file sample `/etc/tripwire/twcfg.txt` sono facoltative. Eccone un esempio:

- `EDITOR` — specifica l'editor di testo invocato da Tripwire. Il parametro di default è `/bin/vi`.
- `LATEPROMPTING` — Se impostata su `true`, questa variabile configura Tripwire in modo che aspetti il più a lungo possibile prima di richiedere una password all'utente, riducendo così al minimo la quantità di tempo in cui la password rimane in memoria. Il parametro di default è `false`.
- `LOOSEDIRECTORYCHECKING` — Se impostata su `true`, configura Tripwire in modo che invii una segnalazione qualora un file contenuto all'interno di una directory e non segnali invece la modifica alla directory stessa. Serve per limitare la ridondanza dei report di Tripwire. Il parametro predefinito è `false`.

- **SYSLOGREPORTING** — Se impostata su `true`, configura Tripwire in modo che segnali eventuali informazioni al demone `syslog` tramite "utente". Il livello di log è impostato su `notice`. Per maggiori informazioni, consultate la pagina `man` di `syslogd`. Il valore di default è `false`.
- **MAILNOVIOLATIONS** — Se impostata su `true` configura Tripwire in modo che spedisca un report via e-mail a intervalli regolari, indipendentemente dal verificarsi o meno di violazioni. Il valore predefinito è `true`.
- **EMAILREPORTLEVEL** — specifica quanto debbano essere dettagliati i report inviati via e-mail. I valori possibili per questa variabile vanno da 0 a 4. Quello di default è 3.
- **REPORTLEVEL** — specifica quanto debbano essere dettagliati i report generati dal comando `tw-print`. Questo valore può essere sovrascritto sulla linea di comando, ma per default è impostato su 3.
- **MAILMETHOD** — specifica il protocollo di posta che Tripwire deve utilizzare. I valori possibili sono `SMTP` e `SENDMAIL`. Quello di default è `SENDMAIL`.
- **MAILPROGRAM** — specifica quale programma di posta Tripwire deve utilizzare. Il valore di default è `/usr/sbin/sendmail -oi -t`.

Dopo aver modificato il file `sample` di configurazione, dovrete configurare il file `sample` di policy.



Avvertenza

Per questioni di sicurezza, si consiglia di cancellare o di salvare in un luogo sicuro tutte le copie del file di testo `/etc/tripwire/twcfg.txt` dopo aver eseguito lo script di installazione o aver generato un file di configurazione dotato di firma. In alternativa, potete cambiare i permessi in modo che il file non risulti più leggibile da qualsiasi utente esterno.

11.3.2. Modifica di `/etc/tripwire/twpol.txt`

Benché non richiesto, si consiglia di modificare questo file `sample` di policy commentato in modo che tenga conto delle applicazioni specifiche, dei file e delle directory presenti sul vostro sistema. Se lasciate inalterata la configurazione di esempio dall'RPM, il vostro sistema potrebbe non godere di una adeguata protezione.

Modificando il file di policy, inoltre, contribuirete ad aumentare l'utilità dei report di Tripwire riducendo al minimo falsi allarmi relativi ai file e ai programmi che non state utilizzando e aggiungendo ulteriori funzionalità, come per esempio le notifiche via e-mail.



Nota Bene

Le notifiche via e-mail non sono configurate per default. Per sapere come configurare questa funzione, consultate la Sezione 11.8.1.

Se modificate il file `sample` di policy dopo aver eseguito lo script di configurazione, consultate la Sezione 11.8 per ottenere istruzioni su come rigenerare un file di policy dotato di firma.



Avvertenza

Per questioni di sicurezza, si consiglia di cancellare o di salvare in un luogo sicuro tutte le copie del file di testo `/etc/tripwire/twpol.txt` dopo aver eseguito lo script di installazione o aver generato

un file di configurazione dotato di firma. In alternativa, potete cambiare i permessi in modo che il file non risulti più leggibile da qualsiasi utente esterno.

11.3.3. Esecuzione dello script `twinstall.sh`

Come utente di root, digitate `/etc/tripwire/twinstall.sh` al prompt della shell per eseguire lo script di configurazione. Lo script `twinstall.sh` vi richiede la password locale e per il sito, che vengono utilizzate per generare chiavi crittografate per proteggere i file di Tripwire. In seguito, lo script crea questi file e li firma.

Nel selezionare la password per il sito e la password locale, dovrete tener conto delle linee guida seguenti:

- Utilizzate almeno otto caratteri alfanumerici e simbolici, ma per ogni password non superate il parametro 1023.
- Non utilizzate le virgolette.
- Le password di Tripwire devono essere completamente differenti dalla password di root o da qualsiasi altra password impostata per il sistema.
- Utilizzate password diverse e uniche sia per la chiave per il sito sia per quella locale.

La password per la chiave per il sito si occupa di proteggere i file di configurazione e di policy di Tripwire. La password per la chiave locale protegge invece i file database e report.



Avvertenza

Se doveste dimenticare la password, sappiate che non c'è modo di decifrare un file con firma. Se ciò dovesse accadere, i file non sarebbero più utilizzabili e sareste costretti a riconfigurare lo script.

Criptando i file di configurazione, di policy, database e report, Tripwire li protegge da "sguardi indiscreti", impedendo a chiunque non sia in possesso delle password di visualizzarli. Ciò significa che, anche qualora un "intruso" riuscisse ad accedere come root al vostro sistema, non sarebbe poi in grado di alterare i file di Tripwire per nascondere le tracce.

Una volta che criptati e firmati, i file di configurazione e di policy generati mediante lo script `twinstall.sh` non devono mai essere rinominati o spostati.

11.4. Inizializzazione del database di Tripwire

Inizializzate il file database di Tripwire digitando `/usr/sbin/tripwire --init` dalla linea di comando.

Durante l'inizializzazione del database, Tripwire crea una serie di oggetti di filesystem basati sulle regole contenute nel file di policy. Questo database è fondamentale per eseguire i controlli dell'integrità.

Per inizializzare il database di Tripwire, utilizzate il comando seguente:

```
/usr/sbin/tripwire --init
```

L'esecuzione di tale comando potrebbe richiedere qualche minuto.

Una volta portate a termine queste operazioni, Tripwire possiede la "fotografia" del vostro filesystem, necessaria per l'esecuzione di controlli in cerca di eventuali modifiche apportate ai file più importanti. Dopo l'inizializzazione del database di Tripwire, dovrete eseguire un primo controllo dell'integrità. Tale controllo va fatto prima di collegare il computer alla rete e di metterlo in funzione. Per sapere come farlo, consultate la la Sezione 11.5.

Non appena vi riterrete soddisfatte della vostra configurazione di Tripwire, potete tranquillamente mettere in funzione il sistema.

11.5. Esecuzione del controllo dell'integrità

Per default, l'RPM di Tripwire aggiunge alla directory `/etc/cron.daily/` uno script di shella chiamato `tripwire-check` che avvierà quotidianamente un controllo di integrità.

Tuttavia, potete eseguire un controllo di integrità in qualunque momento, digitando il comando seguente:

```
/usr/sbin/tripwire --check
```

Durante il controllo dell'integrità, Tripwire confronta lo stato degli oggetti di filesystem attuali con le proprietà registrate nel suo database. Tutte le violazioni vengono visualizzate sullo schermo e una copia criptata del report viene creata all'interno di `/var/lib/tripwire/report/`, a cui potrete accedere in seguito tramite il comando `twprint`, come descritto nella la Sezione 11.6.1.

Se desiderate ricevere un e-mail che vi informi quando si verificano determinate violazioni, potete configurare questa funzione nel file di policy. Per informazioni su come impostare e fare una verifica di questa funzione, consultate la la Sezione 11.8.1.

11.6. Esaminare i report di Tripwire

Il comando `/usr/sbin/twprint` viene utilizzato per visualizzare i report e i database criptati di Tripwire.

11.6.1. Visualizzazione dei report di Tripwire

Con il comando `twprint -m r` potete visualizzare i contenuti in chiaro di un report. &è comunque necessario specificare a `twprint` il file di report da visualizzare.

Un comando `twprint` per la visualizzazione dei report di Tripwire ha un aspetto simile a questo:

```
/usr/sbin/twprint -m r --twrfile  
/var/lib/tripwire/report/<nome>.twr
```

L'opzione `-m r` indica a `twprint` di decodificare un report di Tripwire. L'opzione `--twrfile` indica a `twprint` di utilizzare un file di report specifico.

Il nome del report che desiderate visualizzare comprende il nome dell'host usato da Tripwire per generare il report, nonché la data e l'ora di creazione. Potete visualizzare i report salvati in passato in qualsiasi momento. Digitate semplicemente `ls /var/lib/tripwire/report` per visualizzare una lista dei report di Tripwire.

I report di Tripwire possono essere piuttosto lunghi, a seconda del numero di violazioni individuate o di errori generati. Ecco un esempio di record:

```
Tripwire(R) 2.3.0 Integrity Check Report
```

```
Report generated by:          root
```

```
Report created on:      Fri Jan 12 04:04:42 2001
Database last updated on:  Tue Jan  9 16:19:34 2001
```

```
=====
Report Summary:
=====
```

```
Host name:             some.host.com
Host IP address:       10.0.0.1
Host ID:               None
Policy file used:      /etc/tripwire/tw.pol
Configuration file used: /etc/tripwire/tw.cfg
Database file used:    /var/lib/tripwire/some.host.com.twd
Command line used:     /usr/sbin/tripwire --check
```

```
=====
Rule Summary:
=====
```

```
-----
Section: Unix File System
-----
```

Rule Name	Severity Level	Added	Removed	Modified
Invariant Directories	69	0	0	0
Temporary directories	33	0	0	0
* Tripwire Data Files	100	1	0	0
Critical devices	100	0	0	0
User binaries	69	0	0	0
Tripwire Binaries	100	0	0	0

11.6.2. Visualizzare database di Tripwire

Potete usare il comando `twprint` anche per visualizzare l'intero database o le informazioni dei file selezionati. Ciò è senz'altro utile per visualizzare la quantità di informazioni sta controllando Tripwire sul vostro sistema.

Per visualizzare il database completo di Tripwire, digitate questo comando:

```
/usr/sbin/twprint -m d --print-dbfile | less
```

Verranno visualizzate moltissime informazioni. Le prime righe saranno simili all'esempio qui fornito:

```
Tripwire(R) 2.3.0 Database
```

```
Database generated by:      root
Database generated on:     Tue Jan  9 13:56:42 2001
Database last updated on:  Tue Jan  9 16:19:34 2001
```

```
=====
Database Summary:
=====
```

```
Host name:             some.host.com
Host IP address:       10.0.0.1
Host ID:               None
Policy file used:      /etc/tripwire/tw.pol
Configuration file used: /etc/tripwire/tw.cfg
Database file used:    /var/lib/tripwire/some.host.com.twd
Command line used:     /usr/sbin/tripwire --init
```

```
=====
Object Summary:
=====
```

```
=====
-----
# Section: Unix File System
-----
      Mode          UID          Size          Modify Time
-----
/
  drwxr-xr-x  root (0)      XXX           XXXXXXXXXXXXXXXXXXXX
/bin
  drwxr-xr-x  root (0)    4096          Mon Jan  8 08:20:45 2001
/bin/arch
  -rwxr-xr-x  root (0)    2844          Tue Dec 12 05:51:35 2000
/bin/ash
  -rwxr-xr-x  root (0)   64860         Thu Dec  7 22:35:05 2000
/bin/ash.static
  -rwxr-xr-x  root (0)  405576        Thu Dec  7 22:35:05 2000
```

Per visualizzare le informazioni relative a un file particolare, per es.: `/etc/hosts`, usate il comando seguente:

```
/usr/sbin/twprint -m d --print-dbfile /etc/hosts
```

Il risultato sarà simile a questo:

```
Object name: /etc/hosts

Property:          Value:
-----
Object Type        Regular File
Device Number      773
Inode Number       216991
Mode               -rw-r--r--
Num Links          1
UID                root (0)
GID                root (0)
```

Per conoscere le altre opzioni consultate la pagina man di `twprint`.

11.7. Aggiornamento del database

Se eseguite un controllo dell'integrità e Tripwire riscontra delle violazioni, determinate innanzitutto se tali violazioni sono davvero delle "falle" nella sicurezza oppure modifiche autorizzate. Se di recente avete installato un'applicazione o modificato dei file di sistema importanti, Tripwire segnalerà in modo corretto tutte le violazioni dell'integrità. In questo caso dovrete aggiornare il vostro database, in modo tale che le modifiche apportate non vengano più segnalate come violazioni. Se tuttavia vengono effettuate modifiche a file di sistema che provocano violazioni dell'integrità, ripristinate i file originali con una copia di backup, reinstallate il programma o, in caso di "falle" molto gravi, reinstallare l'intero sistema operativo.

Per aggiornare il proprio database in modo che accetti violazioni di policy intenzionali, Tripwire crea innanzitutto un file report a riferimenti incrociati per il database e poi integra al suo interno le violazioni "valide" estrapolate dal file report. Nell'aggiornare il database, assicuratevi di usare il report più recente.

Per aggiornare il database, digitate il comando seguente, sostituendo *nome* con il nome del file report più recente:

```
/usr/sbin/tripwire --update --twrfile
```

```
/var/lib/tripwire/report/<nome>.twr
```

Tripwire visualizza il file report utilizzando l'editor di testo predefinito (specificato nel file di configurazione Tripwire sulla linea EDITOR). A questo punto, potete deselezionare i file che non desiderate aggiornare nel database Tripwire.



Importante

È importante modificare solo le violazioni dell'integrità *autorizzate*.

Tutti gli aggiornamenti proposti per il database di Tripwire hanno una [x] che precede il nome del file, simile all'esempio che segue:

```
Added:
[x] "/usr/sbin/longrun"

Modified:
[x] "/usr/sbin"
[x] "/usr/sbin/cpqarrayd"
```

Se volete escludere una violazione valida al database di Tripwire, rimuovete la x. Per accettare tutti i file con x accanto a essi come modifiche.

Per modificare i file nell'editor di testo predefinito (vi), digitate i e premete [Invio] per accedere alla modalità di inserimento e apportare le modifiche necessarie. Quando avrete terminato, premete [Esc], digitate :wq e premete [Invio]

Dopo aver chiuso l'editor, inserite la vostra password locale e il database verrà ricostruito e firmato.

Dopo aver scritto un nuovo database di Tripwire, le violazioni autorizzate non verranno più segnalate al controllo dell'integrità successivo.

11.8. Aggiornamento del file di policy di Tripwire

Se desiderate modificare i file di record nel database di Tripwire, cambiare la configurazione della posta oppure la "gravità" per la verifica delle violazioni, dovete modificare il file di policy Tripwire.

Innanzitutto, effettuate tutte le modifiche necessarie al file di policy d'esempio (/etc/tripwire/twpol.txt). Una modifica che di solito viene apportata è il commento a tutti i file non esistenti sul sistema. Se cancellate questo file (come dovrete fare ogni volta che avete terminato di configurare Tripwire), potete rigenerarlo mediante il comando seguente:

```
twadmin --print-polfile > /etc/tripwire/twpol.txt
```

Una delle modifiche più comuni a questo file di policy è quella di commentare tutti i file esistenti sul sistema in modo che non generino un messaggio d'errore `file not found` nei report di Tripwire. Per esempio, se sul vostro sistema non è presente un file chiamato /etc/smb.conf, specificate a Tripwire di non cercarlo, commentando con il carattere # la riga nel file twpol.txt:

```
# /etc/smb.conf -> $(SEC_CONFIG) ;
```

Egrave; poi necessario indicare a Tripwire di creare un nuovo file /etc/tripwire/tw.pol firmato e di generare un file del database aggiornato in base a queste informazioni di policy. Ponendo il caso che /etc/tripwire/twpol.txt sia il file di policy modificato, usate questo comando:

```
/usr/sbin/twadmin --create-polfile -S site.key /etc/tripwire/twpol.txt
```

Vi verrà richiesta la password per la chiave del sito. A questo punto il file `twpol.txt` viene criptato e firmato.

Dopo aver creato un nuovo file `/etc/tripwire/tw.pol`, è importante aggiornare il database di Tripwire. Il modo più affidabile per farlo è quello di cancellare il database di Tripwire e creare un nuovo database utilizzando un nuovo file di policy.

Se il file del database di Tripwire si chiama `bob.domain.com.twd`, digitate questo comando:

```
rm /var/lib/tripwire/bob.domain.com.twd
```

Digitate poi il comando per creare un nuovo database con il file di policy aggiornato:

```
/usr/sbin/tripwire --init
```

Per assicurarvi che il database sia stato modificato in modo corretto, eseguite manualmente il primo controllo dell'integrità e visualizzate i contenuti del report risultante. Per istruzioni specifiche relative a questi task, consultate la Sezione 11.5 e la Sezione 11.6.1.

11.8.1. Tripwire e la posta elettronica

Potete configurare Tripwire in modo che invii un messaggio e-mail a uno o più account qualora venga violata una specifica regola nel file di policy. Per farlo, occorre prima individuare le regole di policy da controllare e la da contattare se avviene tali regole vengono violate. Per i grandi sistemi con più amministratori, è possibile inviare messaggi di avvertimento a più persone in presenza di un certo tipo di infrazione.

Una volta stabilito a chi inviare il messaggio e che cosa notificare, modificate il file `/etc/tripwire/twpol.txt` aggiungendo una linea **mailto=** alla sezione della direttiva di ogni regola. Per farlo, inserite una virgola alla fine della riga **severity=** e digitate **mailto=** all'inizio della riga seguente, seguito da uno o più indirizzi e-mail. È possibile specificare più indirizzi e-mail, separandoli con un punto e virgola.

Se desiderate che due amministratori di sistema (per esempio, Tullio e Cesare) ricevano una notifica via e-mail nel caso in cui venga modificato un programma di networking, cambiate la direttiva della regola "Networking Programs" nel modo seguente:

```
(
  rulename = "Networking Programs",
  severity = $(SIG_HI),
  mailto = tullio@domain.com;cesare@domain.com
)
```

Dopo aver trasformato il file di policy, seguite le istruzioni fornite nella Sezione 11.8 per generare una copia aggiornata, criptata e firmata del file di policy di Tripwire.

11.8.1.1. Invio di messaggi e-mail di prova

Per verificare che la configurazione delle notifiche e-mail sia corretta, utilizzate il seguente comando:

```
/usr/sbin/tripwire --test --email
vostro@indirizzo.email
```

Il programma `tripwire` provvede immediatamente a inviare una e-mail di notifica all'indirizzo specificato.

11.9. Aggiornamento del file di configurazione di Tripwire

Se volete cambiare il file di configurazione di Tripwire, dovete anzitutto modificare il file di configurazione di esempio `/etc/tripwire/twcfg.txt`. Se cancellate questo file (come dovrete fare ogni volta che terminate di configurare Tripwire), potete rigenerarlo mediante il comando seguente:

```
twadmin --print-cfgfile > /etc/tripwire/twcfg.txt
```

Tripwire non riconoscerà nessuna delle modifiche di configurazione fino a quando il file di testo di configurazione non viene firmato e convertito in `/etc/tripwire/tw.pol` tramite il comando `twadmin`.

Per rigenerare un file di configurazione dal file di testo `/etc/tripwire/twcfg.txt`, usate il comando seguente:

```
/usr/sbin/twadmin --create-cfgfile -S site.key /etc/tripwire/twcfg.txt
```

Poiché il file di configurazione non altera alcuno dei file di policy di Tripwire o dei file individuati dall'applicazione, non è necessario rigenerare il database di Tripwire.

11.10. Riferimenti alla posizione del file di Tripwire

Prima di lavorare con Tripwire, dovrete conoscere la posizione dei file più importanti necessari all'applicazione. Tripwire salva i propri file in numerosi posti diversi, a seconda del loro ruolo.

- All'interno della directory `/usr/sbin/` troverete i programmi seguenti:
 - `tripwire`
 - `twadmin`
 - `twprint`
- All'interno della directory `/etc/tripwire/` troverete i file seguenti:
 - `twinstall.sh` — lo script di inizializzazione per Tripwire.
 - `twcfg.txt` — il file sample di configurazione fornito dall'RPM di Tripwire.
 - `tw.cfg` — Il file di configurazione firmato creato dallo script `twinstall.sh`.
 - `twpol.txt` — Il file sample di policy fornito dall'RPM di Tripwire.
 - `tw.pol` — Il file di policy firmato creato dallo script `twinstall.sh`.
 - File delle chiavi — la chiave per il sito e la chiave locale create dallo script `twinstall.sh` che termina con estensione `.key`.
- Dopo aver eseguito lo script di installazione `twinstall.sh`, nella directory `/var/lib/tripwire/` troverete i file seguenti:
 - Il database di Tripwire — Il database dei file del vostro sistema che terminano con estensione `.td`.
 - Report di Tripwire — nella directory `report/` vengono salvati i report di Tripwire.

La prossima sezione fornisce maggiori spiegazioni sui ruoli che svolgono questi file nel sistema di Tripwire.

11.10.1. Componenti di Tripwire

Qui di seguito trovate una descrizione più dettagliata dei ruoli svolti dai vari file, elencati nella sezione precedente.

`/etc/tripwire/tw.cfg`

Si tratta del file di configurazione criptato di Tripwire, che immagazzina le informazioni specifiche relative al sistema, come per esempio la posizione dei file dei dati di Tripwire. Lo script di installazione `twinstall.sh` e il comando `twadmin` generano questo file servendosi dell'informazione contenuta nella versione di testo del file di configurazione (`/etc/tripwire/twcfg.txt`).

Dopo aver eseguito lo script di installazione, l'amministratore del sistema può cambiare i parametri modificando `/etc/tripwire/twcfg.txt` e rigenerando una copia firmata del file `tw.cfg` tramite il comando `twadmin`. Per maggiori informazioni su come procedere, consultate la Sezione 11.9.

`/etc/tripwire/tw.pol`

Il file di policy di Tripwire attivo è un file criptato contenente commenti, regole, direttive e variabili. Questo file indica a Tripwire il modo in cui eseguire i controlli sul sistema. Ciascuna regola contenuta nel file di policy specifica un oggetto di sistema da monitorare. Le regole descrivono inoltre quali modifiche agli oggetti vanno segnalate e quali vanno ignorate.

Gli oggetti di sistema sono i file e le directory che desiderate monitorare. Ciascun oggetto è contraddistinto da un nome. Una determinata proprietà si riferisce a una singola caratteristica di un oggetto che Tripwire può monitorare. Le direttive controllano l'elaborazione di set di regole all'interno di un file di policy. Durante l'installazione, il file sample di policy (`/etc/tripwire/twpol.txt`) viene utilizzato per generare un file di policy di Tripwire attivo.

Dopo aver eseguito lo script di installazione, l'amministratore del sistema può aggiornare il file di policy di Tripwire modificando `/etc/tripwire/twpol.txt` e rigenerando una copia firmata del file `tw.pol` tramite il comando `twadmin`. Per maggiori informazioni su come procedere, consultate la Sezione 11.8.

`/var/lib/tripwire/host_name.twd`

Alla prima inizializzazione, Tripwire utilizza le regole del file di policy firmato per creare questo file database. Il database di Tripwire è una sorta di "fotografia" stilizzata del sistema nel momento in cui si trova in uno stato "sicuro". Tripwire paragona le linee base che ha disposizione in relazione al sistema attuale per determinare quali modifiche sono state apportate. Questa operazione di confronto è conosciuta con il nome di *controllo dell'integrità*.

`/var/lib/tripwire/report/host_name-data_del_report-ora_del_report.twr`

Quando eseguite un controllo dell'integrità, Tripwire produce dei file di report nella directory `/var/lib/tripwire/report/`. I file di report sintetizzano tutte le modifiche dei file che hanno violato le regole del file di policy durante il controllo. I report di Tripwire vengono nominati in base alla convenzione seguente: `host_name-data_del_report-ora_del_report.twr`. Questi report descrivono in dettaglio le differenze tra il database di Tripwire e i vostri file di sistema effettivi.

11.11. Risorse aggiuntive

Le funzioni di Tripwire sono molto più numerose di quelle trattate in questo capitolo. Per maggiori informazioni su Tripwire, consultate la seguente documentazione.

11.11.1. Documentazione installata

- `/usr/share/doc/tripwire-<numero-versione>` — un eccellente punto di partenza per imparare come personalizzare i file di configurazione e di policy nella directory `/etc/tripwire`.
- Potete inoltre fare riferimento alle pagine man inerenti ai comandi `tripwire`, `twadmin` e `twprint`.

11.11.2. Siti Web utili

- <http://www.tripwire.org> — la pagina principale del progetto Open Source di Tripwire dove potete trovare le ultime novità sulle applicazioni, tra cui un elenco di FAQ.
- http://sourceforge.net/project/showfiles.php?group_id=3130 — il link alla documentazione ufficiale più recente del progetto Tripwire

Servizi di rete

Con Red Hat Linux, tutte le connessioni di rete avvengono fra *interfacce* configurate dispositivi di rete fisici connessi al sistema. Esistono tanti tipi interfacce quanti sono i dispositivi che supportano.

I file di configurazione delle interfacce di rete e gli script che le attivano o disattivano sono contenuti nella directory `/etc/sysconfig/network-scripts`. Mentre la presenza di file di interfaccia può variare da sistema a sistema in funzione dell'utilizzo che ne viene fatto, i tre tipi di file presenti in questa directory, *file di configurazione delle interfacce*, *script di controllo delle interfacce* e *file di funzione della rete*, lavorano insieme per consentire a Red Hat Linux di usare i vari dispositivi di rete disponibili.

Il capitolo spiega il rapporto esistente tra questi file nonché le opzioni che ne regolano l'utilizzo.

12.1. File di configurazione per la rete

Prima di trattare i file di configurazione delle interfacce, parleremo dei file di configurazione primari di cui Red Hat Linux si serve per configurare la rete. Comprendere l'importanza del ruolo di questi file nella configurazione delle impostazioni di rete può essere utile per imparare a personalizzare al meglio il proprio sistema.

I file primary di configurazione di rete sono i seguenti:

- `/etc/hosts` — lo scopo principale di questo file è quello di risolvere i nomi di host che non si possono risolvere in altro modo. Può anche essere utilizzato per risolvere i nomi di host su piccole rete prive di server DNS. Indipendentemente dal tipo di rete su cui si trova il computer, questo file dovrebbe contenere una linea in cui è specificato l'indirizzo IP del dispositivo di loopback (`127.0.0.1`) come `localhost.localdomain`. Per maggiori informazioni, consultate la pagina `man` relativa agli host.
- `/etc/resolv.conf` — specifica l'indirizzo IP dei server DNS e il dominio di ricerca. A meno che la configurazione non preveda altrimenti, questo file viene popolato dagli script di inizializzazione della rete. Per ulteriori informazioni su questo file, consultate la pagina `man` `resolv.conf`.
- `/etc/sysconfig/network` — specifica le informazioni sull'instradamento e sugli host in relazione a tutte le interfacce di rete. Per saperne di più su questo file e su quali direttive accetta, consultate la Sezione 3.7.1.23.
- `/etc/sysconfig/network-scripts/ifcfg-<nome-interfaccia>` — per ciascuna delle interfacce di rete presenti sul sistema esiste uno script di configurazione corrispondente. Ognuno di questi file fornisce informazioni specifiche in merito a una determinata interfaccia di rete. Per ulteriori dettagli su questo tipo di file e sulle direttive che accetta, consultate la Sezione 12.2.



Attenzione

La directory `/etc/sysconfig/networking/` viene utilizzata dal **Network Administration Tool** (`redhat-config-network`) e il suo contenuto non dovrebbe mai essere modificato manualmente. Per maggiori informazioni sulla configurazione delle interfacce di rete tramite **Network Administration Tool**, leggete il capitolo intitolato *Configurazione della rete*, che si trova nella *Official Red Hat Linux Customization Guide*.

12.2. File di configurazione delle interfacce

I file di configurazione delle interfacce controllano il funzionamento di un determinato dispositivo di interfaccia di rete. All'avvio del sistema, Red Hat Linux utilizza i file per determinare quali interfacce attivare automaticamente e come configurarle in modo corretto. Solitamente, i file sono chiamati `ifcfg-<nome>`, dove `<nome>` si riferisce al nome del dispositivo controllato dal file di configurazione.

12.2.1. Interfacce Ethernet

Uno dei file di interfaccia più comuni è `ifcfg-eth0`, che controlla la prima *scheda di interfaccia di rete* o *NIC* nel sistema. Un sistema con più schede NIC contiene diversi file `ifcfg-eth` numerati. Poiché ogni dispositivo ha il proprio file di configurazione, l'utente può controllare da vicino il funzionamento di ogni singola interfaccia.

Di seguito è riportato un esempio di un file `ifcfg-eth0` per un sistema che utilizza un indirizzo IP fisso:

```
DEVICE=eth0
BOOTPROTO=none
ONBOOT=yes
NETWORK=10.0.1.0
NETMASK=255.255.255.0
IPADDR=10.0.1.27
USERCTL=no
```

I valori contenuti nel file di configurazione delle interfacce possono variare in funzione di altri valori. Per esempio, il file `ifcfg-eth0` di un'interfaccia che utilizza DHCP è piuttosto diverso, per via del fatto che le informazioni IP sono ora fornite dal server DHCP:

```
DEVICE=eth0
BOOTPROTO=dhcp
ONBOOT=yes
```

Il più delle volte, vorrete utilizzare un'utility GUI, come **Network Administration Tool** (`redhat-config-network`) o `netconfig`, per apportare modifiche ai file di configurazione delle interfacce. Istruzioni sull'utilizzo di questi tool sono contenute nella *Official Red Hat Linux Customization Guide*.

Potete anche modificare manualmente il file di configurazione per una determinata interfaccia. Di seguito è riportato un elenco dei parametri da configurare nel file di configurazione di un'interfaccia.

In ogni file di configurazione delle interfacce sono presenti i valori seguenti:

- `BOOTPROTO=<protocollo>`, dove `<protocollo>` è uno di questi:
 - `none` — non utilizzare alcun protocollo di avvio.
 - `bootp` — utilizzare il protocollo BOOTP.
 - `dhcp` — utilizzare il protocollo DHCP.
- `BROADCAST=<indirizzo>`, dove `<indirizzo>` è l'indirizzo di broadcast. Questa direttiva è stata disapprovata.
- `DEVICE=<nome>`, dove `<nome>` è il nome del dispositivo fisico (tranne per i dispositivi PPP allocati dinamicamente, dove invece corrisponde al *nome logico*).
- `DNS{1,2}=<indirizzo>`, dove `<indirizzo>` è l'indirizzo di un server dei nomi da inserire in `/etc/resolv.conf` qualora la direttiva `PEERDNS` sia impostata su `yes`.

- IPADDR=<indirizzo>, dove <indirizzo> corrisponde all'indirizzo IP.
- NETMASK=<mascheragt;, dove <maschera> è il valore della maschera di rete.
- NETWORK=<indirizzo>, dove <indirizzo> corrisponde all'indirizzo di rete. Questa direttiva è stata disapprovata.
- ONBOOT=<risposta>, dove <risposta> è una delle seguenti:
 - yes — il dispositivo dovrebbe essere attivato all'avvio.
 - no — il dispositivo non dovrebbe essere attivato all'avvio.
- PEERDNS=<risposta>, dove <risposta> è una della seguenti:
 - yes — modifica /etc/resolv.conf se la direttiva DNS è stata impostata. Se state utilizzando DHCP, allora l'opzione yes è quella predefinita.
 - no — non modifica /etc/resolv.conf.
- SRCADDR=<indirizzo>, dove <indirizzo> è l'indirizzo IP sorgente specificato per i pacchetti in uscita.
- USERCTL=<risposta>, dove <risposta> è una delle seguenti:
 - yes — gli utenti non root sono autorizzati a controllare il dispositivo.
 - no — gli utenti non root non sono autorizzati a controllare il dispositivo.

12.2.2. Interfacce di dialup

Se vi collegate a una rete, per esempio Internet, tramite una connessione di dialup PPP, l'interfaccia necessita di un file di configurazione.

Questo file viene creato automaticamente quando si utilizza **wvdial**, **Network Administration Tool** o **Kppp** per creare un account di dialup. Inoltre, le modifiche apportate ai parametri dell'account di dialup si riflettono in questi file di configurazione delle interfacce. La *Official Red Hat Linux Getting Started Guide* contiene istruzioni sull'utilizzo di questi tool di connessione dialup per l'interfaccia grafica. Il file può anche essere creato e modificato manualmente. I file `ifcfg-ppp0` hanno all'incirca questo aspetto:

```
DEVICE=ppp0
NAME=test
WVDIALSECT=test
MODEMPORT=/dev/modem
LINESPEED=115200
PAPNAME=test
USERCTL=true
ONBOOT=no
PERSIST=no
DEFROUTE=yes
PEERDNS=yes
DEMAND=no
IDLETIMEOUT=600
```

SLIP (Serial Line Internet Protocol) è un'altra interfaccia di dialup, sebbene il suo utilizzo sia meno frequente. I file SLIP hanno nomi di file di configurazione delle interfacce quali `ifcfg-sl0`.

Oltre a quelle già elencate, esistono altre opzioni da utilizzare con questi file:

- DEFROUTE=<risposta>, dove <risposta> è una delle seguenti:
 - yes — imposta l'interfaccia come quella di default.
 - no — non imposta l'interfaccia come quella di default.
- DEMAND=<risposta>, dove <risposta> è una delle seguenti:
 - yes — l'interfaccia consente a pppd di iniziare una connessione quando qualcuno tenta di usarlo.
 - no — per quest'interfaccia deve essere stabilita una connessione in modo manuale.
- IDLETIMEOUT=<valore>, dove <valore> corrisponde al numero di secondi di inattività che precede lo scollegamento dell'interfaccia.
- INITSTRING=<stringa>, dove <stringa> rappresenta la stringa init che viene trasmessa al modem. L'opzione è essenzialmente usata con interfacce SLIP.
- LINESPEED=<valore>, dove <valore> è la frequenza di baud del dispositivo. 57600, 38400, 19200 e 9600 sono alcuni dei valori standard utilizzabili.
- MODEMPORT=<dispositivo>, dove <dispositivo> corrisponde al nome del dispositivo seriale utilizzato per stabilire la connessione per l'interfaccia.
- MTU=<valore>, dove <valore> è il parametro MTU (*Maximum Transfer Unit*) dell'interfaccia. Il parametro MTU si riferisce al numero massimo di byte di dati che un frame può trasportare, senza contare le informazioni di intestazione e coda. In alcune situazioni di dialup, impostando il parametro su 576, migliora leggermente il flusso della connessione e viene ridotto il numero di pacchetti persi.
- NAME=<nome>, dove <nome> è il riferimento al titolo attribuito a un insieme di configurazioni di connessione dialup.
- PAPNAME=<nome>, dove <nome> è il nome utente assegnato durante lo scambio di *Protocolli di autenticazione della password (PAP)*, che avviene perché il collegamento a un sistema remoto si possa verificare.
- PEERDNS=<risposta>, dove <risposta> è una delle seguenti:
 - yes — modifica il file `/etc/resolv.conf` del sistema per usare i server DNS forniti dal sistema remoto durante una connessione.
 - no — il file `/etc/resolv.conf` non viene modificato.
- PERSIST=<risposta>, dove <risposta> è una delle seguenti:
 - yes — l'interfaccia deve rimanere sempre attiva, anche dopo lo scollegamento del modem.
 - no — l'interfaccia non deve rimanere sempre attiva.
- REMIP=<indirizzo>, dove <indirizzo> è l'indirizzo IP del sistema remoto. Solitamente non è specificato.
- WVDIALSECT=<nome>, dove <nome> associa l'interfaccia alla configurazione del dialer in `/etc/wvdial.conf`, che contiene il numero telefonico da comporre e altre informazioni importanti per l'interfaccia.

12.2.3. Altre interfacce

Tra gli altri file di configurazioni comuni per le interfacce che utilizzano queste opzioni ci sono: il file `ifcfg-lo`, che controlla il dispositivo locale di loopback del protocollo IP; il file `ifcfg-irlan0`, che si occupa delle impostazioni del primo dispositivo a infrarossi; il file `ifcfg-plip0`, che controlla il primo dispositivo PLIP e infine il file `ifcfg-tr0`, utilizzato con il primo dispositivo Token Ring.

Spesso, nelle operazioni di verifica, viene utilizzata un'interfaccia di loopback locale, insieme a numerose applicazioni che necessitano di un indirizzo IP che punta allo stesso sistema. Tutti i dati inviati al dispositivo di loopback vengono immediatamente rispediti al layer di rete dell'host.

Avvertenza

Non modificate mai manualmente lo script del loopback di interfaccia `/etc/sysconfig/network-scripts/ifcfg-lo`, poiché il sistema operativo potrebbe cessare di funzionare correttamente.

Un'interfaccia a infrarossi consente il passaggio di informazioni tra dispositivi, come un laptop e una stampante, attraverso una connessione a infrarossi che funziona all'incirca come un dispositivo Ethernet, solo che normalmente si verifica mediante una connessione alla pari.

Una connessione *PLIP* (*Parallel Line Interface Protocol*) funziona praticamente allo stesso modo, solo che utilizza una porta parallela.

Da quando Ethernet ha preso piede, le topologie *Token Ring* non sono più molto utilizzate su reti di tipo LAN.

12.2.4. File alias e cloni

Due tipi di file di configurazione delle interfacce meno frequenti, contenuti nella directory `/etc/sysconfig/network-scripts`, sono i file *alias* e *clone*.

I file di configurazione delle interfacce si chiamano `ifcfg-<if-nome>:<alias-valore>`, e permettono a un alias di puntare su un'interfaccia. Per esempio, un file `ifcfg-eth0:0` può, essere configurato perché specifici `DEVICE=eth0:0` e l'indirizzo IP statico `10.0.0.2`. In questo modo funziona come alias di un'interfaccia Ethernet già configurata per ricevere le informazioni IP tramite DHCP in `ifcfg-eth0`. A questo punto, il dispositivo `eth0` è legato a un indirizzo IP dinamico, ma può sempre essere consultato sul sistema tramite l'indirizzo IP fisso `10.0.0.2`.

Il nome dei file cloni di configurazione delle interfacce dovrebbe essere: `ifcfg-<if-nome>-<nome-clone>`. Mentre il file alias rappresenta un altro modo per fare riferimento a un file di configurazione delle interfacce esistente, un file clone viene usato per specificare altre opzioni durante la definizione di un'interfaccia. Per esempio, se avete un'interfaccia Ethernet DHCP standard chiamata `eth0`, può essere simile a questa:

```
DEVICE=eth0
ONBOOT=yes
BOOTPROTO=dhcp
```

Poiché l'opzione `USERCTL` è impostata su `no` se non specificata, gli utenti non possono attivare e disattivare l'interfaccia. Per dare agli utenti la possibilità di farlo, create un clone copiando `ifcfg-eth0` in `ifcfg-eth0-user` e aggiungendo la riga seguente:

```
USERCTL=yes
```

Quando l'utente attiva l'interfaccia `eth0` con il comando `ifup eth0-user`, le opzioni di configurazione di `ifcfg-eth0` e `ifcfg-eth0-user` vengono combinate. L'esempio riportato è molto semplice. Infatti, questo metodo può essere usato con varie opzioni e interfacce.

Il modo più semplice di creare file di configurazione delle interfacce alias e cloni consiste nell'usare il tool grafico di configurazione della rete **Network Administration Tool**. Per maggiori informazioni su come utilizzare questo tool, consultate il capitolo intitolato *Configurazione della rete* nella *Official Red Hat Linux Customization Guide*.

12.3. Script di controllo delle interfacce

Gli script di controllo delle interfacce attivano e le interfacce di sistema. Esistono due script primari di controllo delle interfacce, `/sbin/ifdown` e `/sbin/ifup`, che usano altri script di controllo contenuti nella directory `/etc/sysconfig/network-scripts`.

I due script primari di controllo delle interfacce sono `ifdown` e `ifup` e sono link simbolici agli script contenuti nella directory `/sbin`. Quando viene chiamato uno di questi script, questo necessita di un valore dell'interfaccia da specificare:

```
ifup eth0
Determining IP information for eth0... done.
```

A questo punto, le funzioni di `/etc/rc.d/init.d/functions` e `/etc/sysconfig/network-scripts/network-functions` vengono usate per eseguire numerosi task. Per maggiori informazioni, consultate la Sezione 12.4.

Dopo aver verificato se è stata specificata un'interfaccia e se l'utente che esegue la richiesta ha il permesso di controllare l'interfaccia, viene chiamato lo script idoneo che, in pratica, si occupa di attivare e disattivare l'interfaccia. Di seguito sono elencati gli script di controllo delle interfacce più frequenti:

- `ifup-aliases` — configura gli alias IP dei file di configurazione delle interfacce quando più indirizzi IP sono associati all'interfaccia.
- `ifdown-cipcb` e `ifup-cipcb` — usati per attivare e disattivare le connessioni *CIPE* (*Crypto IP Encapsulation*).
- `ifdown-ipv6` e `ifup-ipv6` — contengono le chiamate di funzione IPv6 che utilizzano le variabili di ambiente di vari file di configurazione delle interfacce e `/etc/sysconfig/network`.
- `ifup-ipx` — usato per attivare un'interfaccia IPX.
- `ifup-plip` — usato per attivare un'interfaccia PLIP.
- `ifup-plusb` — usato per attivare un'interfaccia USB per le connessioni di rete.
- `ifdown-post` e `ifup-post` — contengono i comandi da eseguire dopo che un'interfaccia è stata attivata o disattivata.
- `ifdown-ppp` e `ifup-ppp` — usati per attivare o disattivare un'interfaccia PPP.
- `ifup-routes` — aggiunge instradamenti statici per un particolare dispositivo quando la sua interfaccia viene attivata.
- `ifdown-sit` e `ifup-sit` — contiene chiamate di funzione relative all'attivazione e alla disattivazione del tunnel IPv6 durante una connessione IPv4.
- `ifdown-sl` e `ifup-sl` — usati per attivare o disattivare un'interfaccia SLIP.

Ricordate che in seguito alla rimozione o alla modifica degli script nella directory `/etc/sysconfig/network-scripts/` varie connessioni di interfaccia possono agire in modo inaspettato. È opportuno che solo gli utenti più esperti si cimentino a modificare gli script collegati a una particolare interfaccia di rete.

Potete anche usare lo script `init /etc/rc.d/init.d/network` per attivare e disattivare tutte le interfacce di rete configurate in modo che si attivino all'avvio del sistema. Per farlo, usate il comando:

```
/sbin/service network azione
```

dove *azione* corrisponde a `start`, a `stop` o a `restart`. Potete anche usare il comando `/sbin/service/network status`, che permette di visualizzare un elenco dei dispositivi configurati e di quelli attivi.

12.4. Funzioni di rete

Red Hat Linux utilizza in vari modi alcuni file contenenti funzioni importanti che servono ad attivare e disattivare le interfacce. Anziché fare in modo che tutti i file di controllo delle interfacce contengano le stesse funzioni, tali funzioni sono riunite in alcuni file che possono essere utilizzati quando necessario.

Il file di funzioni di rete più comune è `network-functions`, contenuto nella directory `/etc/sysconfig/network-scripts`. Il file contiene diverse funzioni IPv4 utili per molti script di controllo delle interfacce. Per esempio, consente di contattare i programmi in esecuzione che hanno richiesto informazioni sui cambiamenti nello stato di in un'interfaccia, impostare i nomi di host, trovare un dispositivo gateway, controllare se un dispositivo è attivo o inattivo e aggiungere un instradamento di default.

Poiché le funzioni richieste per le interfacce IPv6 sono diverse da quelle delle interfacce IPv4, esiste un file, `network-functions-ipv6`, che contiene tutte queste informazioni. Il supporto IPv6 deve essere attivato nel kernel per potere comunicare tramite il protocollo. Nel file è presente una funzione che verifica la presenza del supporto IPv6. Inoltre, il file contiene funzioni che configurano e cancellano gli instradamenti IPv6 statici, aggiungono e rimuovono tunnel, aggiungono e rimuovono indirizzi IPv6 da un'interfaccia e verificano l'esistenza di un indirizzo IPv6 in un'interfaccia.

12.5. Risorse aggiuntive

Per reperire maggiori informazioni sugli script di rete, avete a disposizione le fonti riportate qui sotto.

- `/usr/share/doc/initscripts-<versione>/sysconfig.txt` — Una guida completa alle opzioni disponibili per i file di configurazione della rete, comprese le opzioni IPv6, non trattate in questo capitolo.
- `/usr/share/doc/iproute-<versione>/ip-cref.ps` — questo file Postscript™ rappresenta una ricca fonte di informazione sul comando `ip`, che può essere utilizzato per elaborare, tra le altre cose, tabelle di instradamento. Per consultare questo file, utilizzate `ghostview` o `kghostview`.

Firewall e iptables

Linux è dotato di tool avanzati per il *filtraggio dei pacchetti*, ovvero il processo di controllo dei pacchetti di rete bloccati nel kernel che cercano di accedere, transitare e uscire dalla rete. Le versioni di kernel precedenti alla 2.4 si affidavano al comando `ipchains` per il filtraggio e utilizzavano elenchi di regole applicabili ai pacchetti in ogni fase del processo di filtraggio. Con l'uscita del Kernel 2.4 è stato introdotto il comando `iptables`, (chiamato anche *netfilter*), che è simile al comando `ipchains`, ma permette di ampliare notevolmente le possibilità di controllo in fase di filtraggio dei pacchetti.

Questo capitolo si concentra sugli aspetti più importanti del filtraggio dei pacchetti, evidenzia le differenze tra `ipchains` e `iptables`, descrive le varie opzioni disponibili per i comandi di `iptables` e, infine, spiega come mantenere valide le regole di filtraggio tra un riavvio di sistema e l'altro.

Se necessitate di istruzioni su come creare regole `iptables` o configurare un firewall basato su tali regole, consultate la Sezione 13.5.



Avvertenza

Il meccanismo firewall predefinito per il kernel 2.4 è `iptables`, il quale, tuttavia, non può essere utilizzato se `ipchains` è già in esecuzione. Se `ipchains` viene rilevato all'avvio, il kernel emetterà un messaggio di errore e non sarà in grado di avviare `iptables`.

Questo genere di errori non hanno ripercussioni sul funzionamento di `ipchains`.

13.1. Filtraggio dei pacchetti

Il traffico si muove attraverso una rete sotto forma di *pacchetti*, ossia delle raccolte di dati di determinate dimensioni e determinati formati. Per trasmettere un file via rete, il computer mittente deve prima di tutto suddividere il file in tanti pacchetti, utilizzando le regole del protocollo di rete. Ciascuno di questi pacchetti contiene una piccola parte dei dati del file. L'altro computer riceve i pacchetti, li riassume all'interno del file.

Ogni pacchetto contiene le informazioni che gli servono per spostarsi sulla rete e raggiungere la sua destinazione. Il pacchetto è in grado di dire ai computer che si trovano sul suo percorso, e alla macchina di destinazione, da dove proviene, dove è diretto, che tipo di pacchetto è e altro ancora. In genere, i pacchetti sono progettati per trasportare dati, sebbene alcuni protocolli si servano dei pacchetti anche per altri particolari scopi. Per esempio, il protocollo *TCP (Transmission Control Protocol)*, utilizza un pacchetto SYN, che non contiene alcun dato, per dare avvio a una comunicazione tra due sistemi.

Il kernel di Linux possiede la capacità integrata di filtrare i pacchetti, concedendo o negando loro l'accesso al sistema. Il netfilter del kernel 2.4 ha 3 *tabelle* integrate (dette anche *elenchi di regole*). Eccone una lista:

- `filter` — la tabella di default per la gestione dei pacchetti di rete.
- `nat` — questa tabella altera i pacchetti che creano una nuova connessione.
- `mangle` — Questa tabella viene usata per l'alterazione di pacchetti specifici.

Ciascuna di queste tabelle hanno un gruppo di *catene* integrati che corrispondono alle azioni effettuate da netfilter sul pacchetto.

Le catene integrate per la tabella `filter` sono le seguenti:

- *INPUT* — questa catena si applica ai pacchetti ricevuti mediante un'interfaccia di rete.
- *OUTPUT* — si applica ai pacchetti inviati mediante la stessa interfaccia di rete che ha ricevuto i pacchetti in entrata.
- *FORWARD* — si applica ai pacchetti ricevuti su una data interfaccia di rete e inviati mediante un'altra.

Le catene integrate per la tabella `nat` sono le seguenti:

- *PREROUTING* — questa catena altera i pacchetti ricevuti attraverso un'interfaccia di rete al loro arrivo.
- *OUTPUT* — altera i pacchetti generati in locale prima che vengano instradati attraverso un'interfaccia di rete.
- *POSTROUTING* — questa catena altera i pacchetti prima che vengano inviati attraverso un'interfaccia di rete.

Le catene integrate per la tabella `mangle` sono le seguenti:

- *PREROUTING* — questa catena altera i pacchetti ricevuti attraverso un'interfaccia di rete prima che vengano instradati.
- *OUTPUT* — altera i pacchetti generati in locale prima che vengano instradati attraverso un'interfaccia di rete.

Ogni pacchetto di rete ricevuto da o inviato a un sistema Linux viene elaborato da almeno una tabella.

Ogni pacchetto può subire un controllo basato su numerose regole all'interno di ogni regola prima di raggiungere la fine della catena. La struttura e lo scopo di queste regole possono variare, ma di norma si occupano di identificare un pacchetto proveniente da o diretto verso un determinato indirizzo IP o gruppo di indirizzi tramite un protocollo e un servizio di rete particolari.

Indipendentemente dalla sua destinazione, quando un pacchetto soddisfa una particolare regola contenuta in una delle tabelle, gli viene attribuito un determinato *target* (o azione). Se la regola decide di assegnargli il target `ACCEPT`, il pacchetto può saltare i controlli legati alle restanti regole ed è autorizzato a procedere verso la sua destinazione. Se invece la regola opta per il target `DROP`, il pacchetto viene "abbandonato", ossia gli viene negato l'accesso al sistema, e all'host mittente non viene rispedito indietro nulla. Nel caso in cui la regola decida di applicare al pacchetto il target `REJECT`, il pacchetto viene comunque abbandonato, ma al nodo mittente viene recapitato un pacchetto di errore.

Ogni catena ha una policy di default per accettare (`ACCEPT`), abbandonare (`DROP`) o rifiutare (`REJECT`) il pacchetto, oppure per accodarlo (`QUEUE`) verso lo spazio utente. Quando nessuna delle regole della catena risulta applicabile al pacchetto, è la policy di default a decidere come gestirlo.

Il comando `iptables` vi consente sia di configurare questi elenchi di regole, sia di impostare nuove catene e nuove tabelle che rispondano alle vostre particolari esigenze.

13.2. Differenze tra iptables e ipchains

A prima vista, `ipchains` e `iptables` risultano piuttosto simili. Entrambi i metodi di filtraggio dei pacchetti si servono di catene di regole che operano all'interno del kernel di Linux sia per decidere quali pacchetti far entrare o uscire, sia per stabilire come bisogna procedere quando dei pacchetti soddisfano determinate regole. Tuttavia, `iptables` offre un modo molto più estensibile per il filtrare i pacchetti, poiché fornisce all'amministratore un livello di controllo altamente preciso senza, però, aggiungere eccessiva complessità al sistema.

Se avete familiarità con `ipchains`, prima di avventurarvi nell'utilizzo di `iptables` assicuratevi di avere ben presenti le seguenti importanti differenze tra i due metodi:

- *Sotto iptables, ciascuno dei pacchetti filtrati viene elaborato usando le regole di una sola catena.* Per esempio, un pacchetto FORWARD che entra in un sistema usando ipchains deve passare attraverso le catene INPUT, FORWARD e OUTPUT per poter arrivare a destinazione. iptables, invece, invia i pacchetti alla catena INPUT solo se sono destinati al sistema locale e li invia alla catena OUTPUT solo se sono stati generati dal sistema locale. Pertanto, dovete assicurarvi di collocare la regola creata per catturare un determinato pacchetto nella catena corretta, ovvero quella cui verrà effettivamente recapitato il pacchetto.

Il vantaggio è che ora potete esercitare un controllo molto più preciso sulla disposizione di ciascun pacchetto. Se state cercando di bloccare l'accesso a un particolare sito Web, adesso potete impedire i tentativi di accesso da client in esecuzione su host che utilizzano il vostro host come gateway.

- *Il target DENY è stato cambiato in DROP.* In ipchains, ai pacchetti che soddisfavano una certa regola di una catena poteva essere assegnato il target DENY per fare in modo che venissero abbandonati. Per ottenere lo stesso effetto in iptables è stato necessario cambiare il target in DROP.
- *L'ordine è importante quando si devono collocare delle opzioni a una regola in una catena.* In precedenza, con ipchains, quando si digitava una regola, l'ordine che veniva assegnato alle varie opzioni non aveva molta importanza. Il comando iptables, invece, è un po' più esigente da questo punto di vista. Per esempio, dovete specificare la porta sorgente o di destinazione dopo aver specificato il protocollo (ICMP, TCP o UDP).
- *Quando specificate le interfacce di rete da associare a una regola, dovete usare solo interfacce di ingresso (opzione -i) con le catene INPUT o FORWARD e solo interfacce di uscita (opzione -o) con le catene FORWARD o OUTPUT.* Questo è necessario in quanto le catene OUTPUT non vengono più utilizzate dalle interfacce di ingresso e le catene INPUT non interessano i pacchetti che si muovono attraverso le interfacce di uscita.

Questo elenco presenta solo una parte dei cambiamenti apportati, anche perchè il filtro iptables è stato sostanzialmente riscritto. Per informazioni più specifiche, consultate il *Linux 2.4 Packet Filtering HOWTO* e le fonti che si trovano nella Sezione 13.5.

13.3. Opzioni utilizzate nei comandi di iptables

Per predisporre le regole che consentono al kernel di filtrare i pacchetti, bisogna eseguire il comando iptables. Se si usa il comando iptables occorre specificare le opzioni seguenti:

- *Packet Type* — indica quale tipo di pacchetto filtra il comando.
- *Packet Source or Destination* — indica al comando quali pacchetti filtrare in base alla loro provenienza o destinazione.
- *Target* — indica quali azioni vengono adottate sui pacchetti che rispondono ai criteri elencati sopra.

Percé una regola iptables sia valida, le opzioni a essa applicate devono essere raggruppate in modo logico, in base allo scopo e alle condizioni delle regole in generale.

13.3.1. Tabelle

Un aspetto davvero importante di iptables è la possibilità di utilizzare molteplici tabelle per decidere del destino di un determinato pacchetto, in base al tipo di pacchetto preso in considerazione e a seconda di cosa se ne intende fare. Grazie alla natura estensibile di iptables, si possono creare tabelle specifiche che vengono poi memorizzate nella directory `/etc/modules/<versione-kernel>/kernel/net/ipv4/netfilter` per soddisfare esigenze specifiche. Dovete pensare a iptables come a un comando in grado di eseguire, in catene definite, molteplici gruppi di regole ipchains, ognuno dei quali svolge un ruolo specifico.

La tabella di default, chiamata `filter`, contiene le catene standard integrate INPUT, OUTPUT e FORWARD, che sono pressoché identiche alle catene standard in uso con ipchains. iptables,

però, per default contiene altre due tabelle che si occupano di specifici processi del filtraggio dei pacchetti. La tabella `nat` può essere usata per modificare gli indirizzi di provenienza e di destinazione registrati nei pacchetti; la tabella `mangle` consente di effettuare manipolazioni particolari sui pacchetti.

Ogni tabella contiene catene predefinite che svolgono le operazioni necessarie a seconda dello scopo prefisso; tuttavia, si possono facilmente configurare nuove catene all'interno di ciascuna tabella.

13.3.2. Struttura

Molti comandi di `iptables` presentano la struttura seguente:

```
iptables [-t <nome-tabella>] <comando> <nome-catena> <parametro-1> <opzione-1>
<parametro-n> <opzione-n>
```

In questo esempio, l'opzione `<nome-tabella>` permette all'utente di selezionare una tabella diversa da quella di default (`filter`) da utilizzare con il comando. L'opzione `<comando>` è il centro del comando e ordina un'azione specifica da svolgere, come per esempio aggiungere o cancellare un regola da una determinata catena, la quale viene specificata dall'opzione `<nome-catena>`. Questa opzione è seguita da coppie di parametri e opzioni che determinano come funziona la regola e cosa accade quando un pacchetto la soddisfa.

Per quanto riguarda la struttura di un comando `iptables`, è importante ricordare che, diversamente dalla maggior parte degli altri comandi, la sua lunghezza e la sua complessità possono cambiare a seconda dello scopo prefisso. Un semplice comando per rimuovere una regola da una catena può essere molto corto, mentre un comando creato per filtrare pacchetti da una determinata sottorete per mezzo di parametri e opzioni particolari può raggiungere una discreta lunghezza. Nel creare comandi di `iptables`, è utile tener presente che l'impiego di alcuni parametri e di alcune opzioni può comportare la necessità di utilizzare ulteriori parametri e opzioni per specificare più dettagliatamente la precedente richiesta di opzione. Per creare una regola che sia valida, dovete continuare in questo modo finché non avrete soddisfatto tutti i parametri e tutte le opzioni che richiedono ulteriori gruppi di opzioni.

Per vedere un elenco esauriente delle strutture del comando `iptables`, digitate `iptables -h`.

13.3.3. Comandi

I comandi indicano a `iptables` di svolgere un'azione specifica. Per ogni stringa di comando di `iptables` è consentito un solo comando. Fatta eccezione per il comando `help`, tutti i comandi sono scritti con i caratteri maiuscoli.

I comandi di `iptables` sono i seguenti:

- **-A** — aggiunge la regola di `iptables` alla fine della catena specificata. Questo comando viene utilizzato per aggiungere semplicemente una regola, nei casi in cui l'ordine delle regole nella catena è indifferente.
- **-C** — verifica una particolare regola prima di aggiungerla alla catena specificata dall'utente. Questo comando vi può aiutare nella creazione di regole di `iptables` complicate, indicandovi di volta in volta i parametri e le opzioni da aggiungere.
- **-D** — cancella una regola da una determinata catena in base al numero (per esempio il 5 per cancellare la quinta regola contenuta nella catena). Potete anche digitare l'intera regola e `iptables` cancellerà la regola nella catena corrispondente.
- **-E** — rinomina una catena definita dall'utente, senza intaccare la struttura della tabella. Semplicemente, vi risparmia l'inconveniente di dover cancellare la catena, ricrearla sotto il nuovo nome e riconfigurare tutte le vostre regole per quella catena.

- **-F** — svuota la catena selezionata, cancellando di fatto tutte le regole in essa contenute. Se non viene specificata alcuna catena, questo comando cancella tutte le regole di ogni catena.
- **-h** — fornisce un elenco di strutture di comando molto utili e un breve compendio di parametri e opzioni.
- **-I** — inserisce una nuova regola in un determinato punto di una catena. Assegnando un numero alla regola da inserire, indicherete a `iptables` di collocarla nella posizione corrispondente. Se non specificate alcun numero, `iptables` inserirà il vostro comando all'inizio dell'elenco.



Attenzione

Quando aggiungete una regola, assicuratevi di sapere con certezza quale opzione state utilizzando (`-A` oppure `-I`). L'ordine delle regole può essere molto importante per determinare se un certo pacchetto soddisfa una regola piuttosto che un'altra. Assicuratevi che, aggiungendo una regola all'inizio o alla fine di una catena, non andiate a intaccare altre regole in essa contenute.

- **-L** — elenca tutte le regole contenute nella catena specificata dopo il comando. Per ottenere un elenco di tutte le regole di tutte le catene contenute nella tabella di default `filter`, non specificate alcuna catena o tabella. Per elencare tutte le regole di una specifica catena contenuta in una determinata tabella, dovete utilizzare la seguente sintassi:

```
iptables -L <nome-catena> -t <nome-tabella>
```

Nella Sezione 13.3.7, troverete la descrizione delle interessanti opzioni per il comando `-L` che, tra l'altro, forniscono i numeri delle regole e consentono descrizioni più complesse delle regole.

- **-N** — crea una nuova catena con un nome specificato dall'utente.
- **-P** — imposta la policy di default per una determinata catena: se un pacchetto arriva alla fine di una catena e nessuna delle regole è soddisfatta, la policy di default fa in modo che gli venga attribuito un particolare target, per esempio `ACCEPT` o `DROP`.
- **-R** — sostituisce una regola presente in una certa posizione di una catena. Dopo il nome della catena dovete inserire il numero della regola da sostituire. Alla prima regola presente in una catena corrisponde il numero 1.
- **-X** — cancella una catena specificata dall'utente. In nessun caso è permesso cancellare una catena integrata di una tabella.
- **-Z** — azzerà i contatori dei byte e dei pacchetti in tutte le catene di una determinata tabella.

13.3.4. Parametri

Dopo aver specificato certi comandi di `iptables`, compresi quelli utilizzati per aggiungere, cancellare, inserire o sostituire delle regole all'interno di una determinata catena, vi occorrono alcuni parametri per procedere con la creazione della regola di filtraggio dei pacchetti.

- **-c** — azzerà i contatori per una particolare regola. Questo parametro accetta le opzioni `PKTS` e `BYTES` per specificare quale contatore azzerare.
- **-d** — imposta l'host, l'indirizzo IP o la rete di destinazione del pacchetto che soddisferà la regola. Nello specificare una rete potete usare due diversi metodi per indicare la maschera di rete: `192.168.0.0/255.255.255.0` o `192.168.0.0/24`.
- **-f** — fa sì che questa regola sia valida solo per i pacchetti frammentati.

Se specificate l'opzione `!` dopo questo parametro, la regola varrà solo per i pacchetti non frammentati.

- **-i** — serve per impostare l'interfaccia di rete per i pacchetti in ingresso (per esempio `eth0` o `ppp0`), da utilizzare con una particolare regola. Con `iptables`, questo parametro opzionale può

essere utilizzato solo per le catene INPUT e FORWARD se si opera nella tabella *filter* e con la catena PREROUTING se si opera nelle tabelle *nat* e *mangle*.

Questo parametro presenta varie opzioni molto utili, che si possono anteporre al nome di un'interfaccia:

- **!** — indica al parametro di "non corrispondere", ovvero fa in modo che tutte le interfacce così specificate vengano escluse da questa regola.
- **+** — serve per indicare tutte le interfacce che corrispondono a una determinata stringa. Per esempio, il parametro `-i eth+` applica questa regola a tutte le interfacce ethernet presenti sul vostro sistema, escludendo qualsiasi altra interfaccia (per esempio `ppp0`).

Se utilizzate il parametro `-i` senza specificare alcuna interfaccia, la regola riguarderà tutte le interfacce.

- **-j** — indica a iptables di saltare direttamente a un particolare target quando un pacchetto soddisfa una certa regola. Target validi da utilizzare dopo l'opzione `-j` includono le opzioni standard (ACCEPT, DROP, QUEUE e RETURN) e anche opzioni estese reperibili attraverso i moduli caricati per default con il pacchetto RPM iptables di Red Hat Linux (per esempio, LOG, MARK e REJECT, per citarne alcune). Per maggiori informazioni su questi e altri target e per conoscere le regole concernenti il loro utilizzo (molti target si possono usare solo con una determinata tabella), consultate la pagina man di iptables.

Oltre a specificare le azioni del target, potete anche indirizzare un pacchetto che soddisfa questa regola a una catena definita dall'utente al di fuori della catena attuale. Questo vi permette di utilizzare altre regole per il controllo di tale pacchetto, filtrandolo ulteriormente sulla base di criteri più specifici.

Se non viene specificato alcun target, il pacchetto oltrepassa la regola senza che succeda nulla. Tuttavia, il contatore per questa regola sale di un'unità, poiché è stata comunque soddisfatta dal pacchetto.

- **-o** — serve per impostare l'interfaccia di rete per i pacchetti in uscita, da utilizzare con un regola. Può essere usato solo con le catene OUTPUT e FORWARD se si opera nella tabella *filter* e con la catena POSTROUTING se si opera nelle tabelle *nat* e *mangle*. Le opzioni di questo parametro sono le stesse del parametro relativo all'interfaccia di rete per i pacchetti in ingresso (`-i`).
- **-p** — imposta il protocollo IP per la regola, che può essere `icmp`, `tcp`, `udp` o `all`, per coprire ogni possibile protocollo. Inoltre, si possono utilizzare anche altri protocolli meno diffusi, di cui è disponibile un elenco in `/etc/protocols`. Se non viene specificata questa opzione durante la creazione della regola, l'opzione di default è `all`.
- **-s** — imposta l'indirizzo sorgente per un particolare pacchetto, utilizzando la stessa sintassi del parametro di destinazione (`-d`).

13.3.5. Opzioni estese

Diversi protocolli di rete forniscono speciali opzioni configurabili in modi particolari per corrispondere a un determinato pacchetto che utilizza, appunto, uno di quei protocolli. Naturalmente, bisogna prima specificare il protocollo nel comando iptables, per esempio usando `-p tcp<nome-protocollo>`, per rendere disponibili le opzioni relative a quel protocollo.

13.3.5.1. Protocollo TCP

Le opzioni speciali disponibili per il protocollo TCP (`-p tcp`) sono:

- **--dport** — definisce la porta di destinazione per il pacchetto. Per configurare questa opzione, potete usare il nome di un servizio di rete (come `www` o `smtp`), un numero di porta o un range di numeri di porta. Nel file `/etc/services` potete trovare i nomi e gli alias dei servizi di rete e i

numeri di porta che utilizzano. Per specificare questa opzione potete usare anche `--destination-port`.

Per indicare uno specifico range di numeri di porta, scrivete i due numeri separati dai due punti (:). Per esempio potete specificare `-p tcp --dport 3000:3200`. Il range massimo consentito è 0:65535.

Potete anche utilizzare il punto esclamativo (!) come flag, scrivendolo dopo l'opzione `--dport`, per indicare a iptables di prendere in considerazione tutti i pacchetti che *non* utilizzano quel servizio di rete o quella porta.

- `--sport` — determina la porta sorgente del pacchetto, utilizzando le stesse opzioni di `--dport`. Per specificare questa opzione potete usare anche `--source-port`.
- `--syn` — rende la regola applicabile a tutti i pacchetti TCP adibiti all'apertura di una connessione (più noti come *pacchetti SYN*). Il punto esclamativo (!) posto come flag dopo l'opzione `--syn` indica che devono essere esaminati tutti i pacchetti non SYN.
- `--tcp-flags` — permette di filtrare i pacchetti TCP in base a particolari parti o flag. l'opzione `--tcp-flags` Può essere seguita da due parametri, ovvero i flag delle varie parti separate da una virgola. Il primo parametro è la maschera, che indica i flag da esaminare per quel pacchetto. Il parametro indica quali sono i flag da impostare. I flag possibili sono ACK, FIN, PSH, RST, SYN e URG. Si possono usare anche ALL e NONE per indicare che devono essere esaminati tutti i flag oppure nessuno di essi.

Per esempio, una regola di iptables contenente l'opzione `-p tcp --tcp-flags ACK,FIN,SYN SYN` prende in considerazione solo i pacchetti TCP in cui è impostato il flag SYN, ma non i flag ACK e FIN.

Come per molte altre opzioni, il punto esclamativo (!) posto dopo `--tcp-flags` viene utilizzato per invertire l'effetto, in modo che vengano esaminati i pacchetti in cui sono impostati i flag del secondo parametro.

- `--tcp-option` — cerca la corrispondenza con un particolare pacchetto in cui sono impostate delle opzioni TCP specifiche. Anche in questo caso si può invertire l'effetto mediante il punto esclamativo (!).

13.3.5.2. Protocollo UDP

Le opzioni estese disponibili per il protocollo UDP (`-p udp`) sono:

- `--dport` — specifica la porta di destinazione del pacchetto UDP, usando il nome di un servizio, un numero di porta o un range di numeri di porta. In sostituzione, potete anche usare l'opzione `--destination-port`. Per conoscere i vari modi di impiego relativi all'opzione `--dport`, leggete la Sezione 13.3.5.1.
- `--sport` — specifica la porta sorgente del pacchetto UDP, utilizzando il nome di un servizio, un numero di porta o un range di numeri di porta. In sostituzione, potete anche usare l'opzione `--source-port`. Per conoscere i vari modi di impiego relativi all'opzione `--dport`, leggete la Sezione 13.3.5.1.

13.3.5.3. Protocollo ICMP

I pacchetti che utilizzano il protocollo ICMP (Internet Control Message Protocol), possono essere esaminati sulla base della opzione seguente (se si specifica `-p icmp`):

- `--icmp-type` — determina il nome o il numero del tipo di ICMP che soddisfa la regola. Potete visualizzare un elenco di nomi validi per i tipi di ICMP digitando il comando `iptables -p icmp -h`.

13.3.5.4. Moduli con opzioni estese aggiuntive

Ulteriori opzioni speciali, che non si riferiscono a nessun protocollo in particolare, si possono reperire attraverso i moduli che si caricano quando sono utilizzati dal comando `iptables`. Per usare uno di questi moduli, dovete caricarlo per nome inserendo l'opzione `-m <nome-modulo>` nel comando `iptables` che sta creando una regola.

Molti di questi moduli, ciascuno dei quali con le proprie opzioni estese specifiche, sono disponibili per default. Potete persino creare da voi dei moduli aggiuntivi per ottenere altre opzioni, per esempio relative a specifiche esigenze di rete. Poichè i moduli esistenti sono numerosi, andiamo a presentarvi solo quelli più famosi e diffusi.

Il modulo `limit` vi consente di porre un limite al numero di pacchetti che soddisfano una certa regola. Questo risulta particolarmente utile nelle registrazioni per ridurre la quantità di pacchetti da esaminare, onde impedire l'arrivo di un'ondata di messaggi di log e un consumo eccessivo di risorse di sistema.

- `--limit` — stabilisce il numero massimo di confronti da effettuare sui pacchetti in un determinato lasso di tempo, specificato da un numero e da un'unità di tempo nel formato `<numero>/<tempo>`. Per esempio, `--limit 5/hour` indica che in un'ora sono consentiti al massimo cinque confronti.

Se il numero e l'unità di tempo non vengono specificati, il valore predefinito è `3/hour`.

- `--limit-burst` — impone un limite al numero di pacchetti che possono soddisfare una determinata regola. Questa opzione va associata all'opzione `--limit` e può essere associata a un numero per impostare il limite sopraccitato.

Se non viene specificato alcun numero, soltanto cinque pacchetti potranno soddisfare la regola.

Il modulo `state`, che utilizza l'opzione `--state`, prende in considerazione pacchetti con i seguenti stati di connessione:

- `ESTABLISHED` — il pacchetto è associato ad altri pacchetti in una connessione stabilita.
- `INVALID` — il pacchetto non corrisponde ad alcuna connessione conosciuta.
- `NEW` — il pacchetto sta creando una nuova connessione oppure fa parte di una connessione bidirezionale non vista in precedenza.
- `RELATED` — il pacchetto sta creando una nuova connessione in qualche modo legata a una connessione esistente.

Questi stati di connessione possono essere combinati tra loro e usati in contemporanea. Occorre scriverli separati da una virgola, come in questo esempio: `-m state --state INVALID,NEW`.

Per esaminare uno specifico indirizzo hardware (MAC) di una scheda ethernet, utilizzate il modulo `mac`, con l'opzione `--mac-source` seguita eventualmente da un indirizzo MAC. Per escludere un indirizzo MAC da una regola, fate seguire all'opzione `--mac-source` un punto esclamativo (!).

Per conoscere le altre opzioni aggiuntive reperibili attraverso i moduli, consultate la pagina man di `iptables`.

13.3.6. Opzioni relative ai target

Vi sono molti target attribuibili a un pacchetto dopo che ha soddisfatto una determinata regola. Il loro compito è quello di decidere che cosa fare del pacchetto e di svolgere, se necessario, ulteriori operazioni, come per esempio registrare l'azione. Inoltre, ogni catena ha un suo target di default, che entra in gioco nel caso in cui un pacchetto non soddisfi nessuna delle regole in essa contenute o quando nessuna della regole che il pacchetto ha soddisfatto specificano un target.

I target standard disponibili per decidere cosa fare di un pacchetto non sono molti:

- `><catena-definita-dall'utente>` — nome di una catena creata e definita in precedenza in questa tabella con le cui regole si confronterà questo pacchetto (in aggiunta a eventuali altre regole di altre catene già impostate per controllare il pacchetto).
- `ACCEPT` — autorizza il pacchetto a procedere verso la propria destinazione o verso un'altra catena.
- `DROP` — abbandona il pacchetto. Il sistema che ha inviato il pacchetto non riceve alcun messaggio di mancata accettazione. Il pacchetto viene semplicemente rimosso dalla regola di controllo della catena e scartato.
- `QUEUE` — accoda il pacchetto verso lo spazio utente.
- `RETURN` — arresta il controllo del pacchetto mediante le regole della catena corrente. Se un pacchetto con target `RETURN` soddisfa una regola contenuta in una catena chiamata da un'altra catena, allora viene rimandato alla prima catena, dove la regola riprende il controllo da dove l'aveva lasciato. Se la regola `RETURN` è utilizzata all'interno di una catena integrata e il pacchetto non può tornare alla catena precedente, il target predefinito per la catena corrente decide quale azione intraprendere.

In aggiunta a questi target standard vi sono altri target che si possono utilizzare con delle estensioni chiamate *moduli dei target*. Per maggiori informazioni in merito a questi ultimi, leggete la Sezione 13.3.5.4.

Vi sono numerose estensioni di moduli di target, molte delle quali si adattano solo a specifiche tabelle o situazioni. Quelli che seguono sono due dei moduli di target più diffusi, inclusi per default in Red Hat Linux:

- `LOG` Registra tutti i pacchetti che soddisfano questa regola. Poiché i pacchetti sono registrati dal kernel, il file `/etc/syslog.conf` determina dove vengono memorizzate queste voci di log (per default, nel file `/var/log/messages`).

Dopo il target `LOG` si possono specificare varie opzioni per determinare il modo in cui deve avvenire la registrazione:

- `--log-level` — stabilisce il livello di priorità di una registrazione. Potete trovare un elenco dei livelli di priorità nella pagina `man syslog.conf`.
- `--log-ip-options` — vengono registrate tutte le opzioni impostate nell'intestazione di un pacchetto IP.
- `--log-prefix` — antepone una stringa di testo alla linea del log durante la scrittura. Dopo l'opzione `--log-prefix` si possono inserire fino a 29 caratteri. È utile anche per scrivere filtri per il syslog da usare insieme alla registrazione dei pacchetti.
- `--log-tcp-options` — Vengono registrate tutte le opzioni impostate nell'intestazione di un pacchetto TCP.
- `--log-tcp-sequence` — scrive nel log il numero di sequenza TCP per il pacchetto.
- `REJECT` — rispedisce un pacchetto di errore al sistema che l'ha inviato, dopodiché abbandona il pacchetto (`DROP`). Questo target è utile se desiderate notificare al sistema mittente che si è verificato un problema con il pacchetto inviato.

Il target `REJECT` accetta un'opzione del tipo `--reject-with <tipo>` per aggiungere ulteriori informazioni al pacchetto di errore da inviare. Il messaggio `port-unreachable` è il `<tipo>` di errore fornito per default quando non è stata specificata alcuna opzione. Per ottenere un elenco completo delle opzioni `<tipo>` disponibili, consultate la pagina man di `iptables`.

Sempre nella pagina man di `iptables` potete trovare altre estensioni di target, tra cui alcune molto utili per eseguire il masquerading mediante la tabella `nat` o per alterare i pacchetti mediante la tabella `mangle`.

13.3.7. Elenco delle opzioni

Il comando elenco di default, `iptables -L`, fornisce una panoramica di base delle catene di regole correnti della tabella predefinita `filter`. Altre opzioni consentono di ottenere ulteriori informazioni e di organizzarle in vari modi:

- `-v` — visualizza output complessi, come per esempio il numero di pacchetti e di byte che hanno attraversato ciascuna catena, il numero di pacchetti e di byte che hanno soddisfatto ciascuna regola e le interfacce disponibili per quella regola.
- `-x` — espande i numeri fino a visualizzare il loro valore esatto. Se il sistema è occupato, il contatore di pacchetti e byte relativo a una certa catena mostra un output abbreviato, ponendo `K` (per le migliaia), `M` (per i milioni) o `G` (per i miliardi) alla fine del numero. Questa opzione impone la visualizzazione completa del numero.
- `-n` — visualizza gli indirizzi IP e i numeri di porta in formato numerico invece che nel formato di default (con hostname e servizio di rete).
- `--line-numbers` — elenca le regole contenute in ogni catena vicino al loro ordine numerico nella catena. Questa opzione è utile se si vuole cancellare una determinata regola di una catena o stabilire in quale posizione della catena inserire una certa regola.

13.4. Memorizzare informazioni relative a iptables

Le regole create mediante il comando `iptables` vengono memorizzate solo nella RAM. Se riavviate il sistema dopo aver configurato delle regole di `iptables`, le vostre impostazioni andrebbero perse, dunque se volete renderle effettive a ogni avvio di sistema, dovete salvarle nel file `/etc/sysconfig/iptables`.

Per farlo, digitate il comando `/sbin/service iptables save`. In questo modo, lo script `init` per `iptables` esegue il programma `/sbin/iptables-save` e salva la configurazione corrente nel file `/etc/sysconfig/iptables`. Questo file dovrebbe essere leggibile solo da `root`, così le vostre regole di filtraggio dei pacchetti non potranno essere visualizzate da altri utenti.

È sempre consigliabile testare una nuova regola di `iptables` prima di salvarla nel file `/etc/sysconfig/iptables`; è anche possibile copiare in questo file delle regole di `iptables` provenienti dal file `/etc/sysconfig/iptables` di un altro sistema. Questo vi consente di distribuire vari gruppi di regole `iptables` a molti sistemi diversi.



Importanted

Se distribuite il file `/etc/sysconfig/iptables`, per rendere effettive le nuove regole dovete digitare `/sbin/service iptables restart`.

13.5. Risorse aggiuntive

Per maggiori informazioni circa il filtraggio di pacchetti con `iptables`, consultate le fonti riportate qui di seguito.

13.5.1. Documentazione installata

- La pagina man di `iptables` contiene la descrizione completa di vari comandi, parametri e altre opzioni.

13.5.2. Siti Web utili

- <http://netfilter.samba.org> — contiene informazioni di vario genere riguardo `iptables`, tra cui una sezione FAQ relativa ai problemi specifici che potete incontrare e svariate guide molto utili a cura di Rusty Russel, il responsabile dell'implementazione del firewall IP di Linux. I documenti HOWTO contenuti in questo sito si occupano di argomenti quali i concetti di base relativi al networking, le configurazioni di NAT e del filtraggio dei pacchetti con il kernel 2.4.
- http://www.linuxnewbie.org/nhf/Security/IPtables_Basics.html — una descrizione essenziale e generica di come si muovono i pacchetti attraverso il kernel e un'introduzione su come creare semplici comandi di `iptables`.
- <http://www.redhat.com/support/resources/networking/firewall.html> — contiene link aggiornati a numerose risorse di filtraggio dei pacchetti.

Apache HTTP Server

Apache HTTP Server è un potente server Web open source e in grado di competere con quelli in commercio sviluppato dalla Apache Software Foundation (<http://www.apache.org>). Red Hat Linux 8.0 include la versione 2.0 di Apache HTTP Server oltre a numerosi moduli server progettati per potenziarne le funzionalità.

Il file di configurazione predefinito di Apache HTTP Server si adatta bene alle esigenze tanto che potrebbe non rendersi mai necessario modificare le impostazioni predefinite. Tuttavia, qualora si considerasse farlo, potrebbe rivelarsi utile quanto riportato nel presente capitolo in relazione alla personalizzazione del file di configurazione di Apache HTTP Server (`/etc/httpd/conf/httpd.conf`).



Avvertimento

Se intendete utilizzare l'utility grafica **HTTP Configuration Tool** (`redhat-config-httpd`), *non* modificate il file di configurazione di Apache HTTP Server. L'applicazione **HTTP Configuration Tool** rigenera questo file ogni volta che viene utilizzato.

Per maggiori informazioni sull'utility **HTTP Configuration Tool**, consultate il capitolo relativo alla *configurazione del server HTTP Apache* nella *Official Red Hat Linux Customization Guide*.

14.1. Apache HTTP Server 2.0

Red Hat Linux 8.0 viene distribuito con la versione 2.0 di Apache HTTP Server. Esistono notevoli differenze tra la versione 2.0 e la versione 1.3, disponibile come le versioni precedenti di Red Hat Linux. Questa sezione esamina alcune delle nuove caratteristiche di Apache HTTP Server 2.0 e sottolinea le importanti modifiche. Se dovete migrare un file di configurazione della versione 1.3 al nuovo formato, consultate la Sezione 14.2.

14.1.1. Caratteristiche di Apache HTTP Server 2.0

L'arrivo di Apache HTTP Server 2.0 porta numerose nuove caratteristiche, tra le quali:

- *Nuove API Apache* — Apache HTTP Server dispone di una nuova serie di API (Application Programming Interfaces), più potente, per i moduli.



Attenzione

I moduli creati per Apache HTTP Server 1.3 non funzioneranno se non viene effettuato l'aggiornamento alla nuova API. Se non siete certi del fatto che un modulo particolare sia supportato, consultate chi si occupa della gestione del pacchetto *prima* dell'aggiornamento.

- *Filtro* — i moduli per Apache HTTP Server 2.0 hanno la possibilità di funzionare come filtri del contenuto. Per ulteriori informazioni sul funzionamento dei filtri, consultate la Sezione 14.2.4.
- *Supporto IPv6* — Apache HTTP Server 2.0 supporta la generazione successiva di indirizzi IP.
- *Direttive semplificate* — numerose direttive sono state rimosse, mentre altre sono state semplificate. Per ulteriori informazioni sulle direttive specifiche, consultate la Sezione 14.5.
- *Risposte agli errori in più lingue* — Quando utilizzate documenti *SSI (Server Side Include)*, le pagine di risposta agli errori personalizzabili possono essere distribuite in più lingue.

- *Supporto di più protocolli* — Apache HTTP Server 2.0 ha la capacità di supportare più protocolli.

Un elenco più esauriente di modifiche è disponibile online all'indirizzo <http://httpd.apache.org/docs-2.0/>.

14.1.2. Modifiche ai pacchetti in Apache HTTP Server 2.0

In Red Hat Linux 8.0 il pacchetto Apache HTTP Server è stato rinominato, mentre alcuni pacchetti correlati sono stati anch'essi rinominati, eliminati o incorporati in altri pacchetti.

Di seguito è riportato un elenco delle modifiche ai pacchetti:

- I pacchetti `apache`, `apache-devel` e `apache-manual` sono stati rinominati rispettivamente come `httpd`, `httpd-devel` e `httpd-manual`.
- Il pacchetto `mod_dav` è stato incorporato nel pacchetto `httpd`.
- I pacchetti `mod_put` e `mod_roaming` sono stati rimossi, dato che la loro funzionalità è un sottinsieme di quella fornita da `mod_dav`.
- I pacchetti `mod_auth_any` e `mod_bandwidth` sono stati rimossi.
- Il numero di versione del pacchetto `mod_ssl` è ora sincronizzato con il pacchetto `httpd`. Ciò significa che il pacchetto `mod_ssl` per Apache HTTP Server 2.0 dispone di un numero di versione inferiore rispetto al pacchetto `mod_ssl` per Apache HTTP Server 1.3.

14.1.3. Modifiche al filesystem in Apache HTTP Server 2.0

In seguito all'aggiornamento a Apache HTTP Server 2.0 si verificano le seguenti modifiche al layout del filesystem:

- *È stata aggiunta una nuova directory di configurazione, `/etc/httpd/conf.d/`.* — questa nuova directory viene utilizzata per archiviare i file di configurazione per i moduli in singoli pacchetti, come `mod_ssl`, `mod_perl` e `php`. Al server viene indicato di caricare i file di configurazione da questo percorso tramite la direttiva `Include conf.d/*.conf` che si trova nel file di configurazione di Apache HTTP Server `/etc/httpd/conf/httpd.conf`.



Avvertimento

È importante che questa linea venga inserita al momento della migrazione di una configurazione esistente.

- *I programmi `ab` e `logresolve` sono stati spostati.* — queste utility sono state spostate dalla directory `/usr/sbin/` alla directory `/usr/bin/`. In questo modo gli script con percorsi assoluti per questi file binari non avranno esito positivo.
- *Il comando `dbmmanage` è stato sostituito.* — tale comando è stato sostituito da `htdbm`. Per ulteriori informazioni, consultate la Sezione 14.2.4.4.
- *Il file di configurazione `logrotate` è stato rinominato.* — tale file è stato rinominato da `/etc/logrotate.d/apache` a `/etc/logrotate.d/httpd`.

La sezione successiva spiegherà come eseguire la migrazione di una configurazione Apache HTTP Server 1.3 al nuovo formato 2.0.

14.2. Migrazione dei file di configurazione di Apache HTTP Server 1.3

Se avete aggiornato il vostro server da una versione precedente di Red Hat Linux in cui Apache HTTP Server era già installato, il nuovo file di configurazione per il pacchetto Apache HTTP Server 2.0 verrà installato come `/etc/httpd/conf/httpd.conf.rpmnew` il file della versione 1.3 originale `httpd.conf` rimarrà invariato. Dipende da voi se utilizzare il nuovo file di configurazione e migrare le vecchie impostazioni oppure utilizzare il file esistente come base e modificarlo secondo le necessità. Tuttavia, alcune parti del file sono state modificate più di altre e un approccio variato è in genere la scelta migliore. I file di configurazione per entrambe le versioni 1.3 e 2.0 sono suddivisi in tre sezioni. L'obiettivo di questa guida è quello di suggerire il metodo più semplice.

Se il vostro `httpd.conf` è una versione modificata della versione predefinita di Red Hat e avete salvato una copia dell'originale, potrebbe essere più semplice chiamare il comando `diff`, come nell'esempio riportato di seguito:

```
diff -u httpd.conf.orig httpd.conf | less
```

Questo comando evidenzia le modifiche effettuate. Se non disponete di una copia del file originale, estraetela da un pacchetto RPM mediante i comandi `rpm2cpio` e `cpio` come nell'esempio riportato di seguito:

```
rpm2cpio apache-1.3.23-11.i386.rpm | cpio -i --make
```

È infine utile sapere che Apache HTTP Server dispone di una modalità di verifica degli errori all'intero della configurazione. Per accedervi, digitate il comando riportato di seguito:

```
apachectl configtest
```

14.2.1. Configurazione dell'ambiente globale

La sezione sull'ambiente globale del file di configurazione contiene le direttive che influiscono sul funzionamento globale di Apache HTTP Server, come il numero di richieste simultanee che può gestire e i percorsi dei vari file che utilizza. Questa sezione richiede un grande numero di modifiche in confronto ad altre ed è pertanto consigliabile che vi basiate sul file di configurazione di Apache HTTP Server 2.0 ed eseguite la migrazione delle vecchie impostazioni.

14.2.1.1. Selezione delle interfacce e delle porte di cui eseguire il binding

Le direttive `BindAddress` e `Port` non esistono più. La relativa funzione è ora fornita da una direttiva più flessibile `Listen`.

Se avete impostato `Port 80` nel file di configurazione della versione 1.3, dovete modificare l'opzione in `Listen 80`. Se avete impostato `Port` a un valore *diverso da 80* dovete anche aggiungere il numero di porta al contenuto della direttiva `ServerName`.

Per esempio, quella riportata di seguito è una direttiva di esempio di Apache HTTP Server 1.3:

```
Port 123
ServerName www.example.com
```

Per migrare questa impostazione a Apache HTTP Server 2.0, utilizzate la struttura riportata di seguito:

```
Listen 123
ServerName www.example.com:123
```

Per ulteriori informazioni su questo argomento, consultate la documentazione indicata di seguito nel sito Web di Apache Software Foundation:

- http://httpd.apache.org/docs-2.0/mod/mpm_common.html#listen
- <http://httpd.apache.org/docs-2.0/mod/core.html#servername>

14.2.1.2. Regolazione della dimensione del pool del server

In Apache HTTP Server 2.0 la responsabilità di accettare richieste e inviare processi secondari per gestirle è stata riassunta in un gruppo di moduli denominato *MPM (Multi-Processing Modules)*. A differenza di altri moduli, solo un modulo del gruppo MPM può essere caricato da Apache HTTP Server perché un modulo MPM è responsabile della gestione e della distribuzione delle richieste di base. Con la versione 2.0 sono disponibili tre moduli MPM : prefork, worker e perchild.

Il comportamento originale di Apache HTTP Server 1.3 è stato spostato nel modulo MPM prefork. Attualmente solo prefork è disponibile in Red Hat Linux, anche se altri moduli MPM potranno essere disponibili successivamente.

L'MPM fornito per default in Red Hat Linux è prefork che accetta le stesse direttive di Apache HTTP Server 1.3. Le direttive riportate di seguito possono essere migrate direttamente:

- StartServers
- MinSpareServers
- MaxSpareServers
- MaxClients
- MaxRequestsPerChild

Per ulteriori informazioni su questo argomento, consultate la documentazione indicata di seguito nel sito Web di Apache Software Foundation:

- <http://httpd.apache.org/docs-2.0/mpm.html>

14.2.1.3. Supporto DSO (Dynamic Shared Object)

Sono molte le modifiche necessarie in questo caso ed è consigliabile che chiunque tenti di modificare una configurazione Apache 1.3 perché si adegui ad Apache 2.0 (in contrapposizione alla migrazione delle modifiche nella configurazione di Apache 2.0), copi questa sezione dal file di configurazione di Red Hat Linux Apache HTTP Server 2.0.



Importante

Se decidete di provare a modificare il file originale, dovete tenere presente che è di fondamentale importanza che il file `httpd.conf` contenga la direttiva riportata di seguito:

```
Include conf.d/*.conf
```

L'omissione di questa direttiva porta al fallimento di tutti i moduli contenuti nel pacchetto di RPM, come `mod_perl`, `php` e `mod_ssl`.

Coloro che non desiderano comunque copiare la sezione dal file di configurazione di Apache HTTP Server 2.0 devono notare quanto riportato di seguito:

- Le direttive `AddModule` `ClearModuleList` non esistono più. Queste direttive erano utilizzate per garantire che i moduli potevano essere abilitati nell'ordine corretto. L'API di Apache 2.0 consente ai moduli di specificare l'ordine, eliminando la necessità di queste due direttive.
- L'ordine delle linee `LoadModule` non è più importante.
- Molti moduli sono stati aggiunti, rimossi, rinominati, suddivisi o incorporati l'uno nell'altro.
- Le linee `LoadModule` per i moduli dei pacchetti dei loro RPM (`mod_ssl`, `php`, `mod_perl` e simili) non sono più necessarie perché possono essere disponibili nel file della directory `/etc/httpd/conf.d/`.
- Le varie definizioni di `HAVE_XXX` non sono più definite.

14.2.1.4. Altre modifiche all'ambiente globale

Le direttive riportate di seguito sono state rimosse dalla configurazione di Apache HTTP Server 2.0:

- *ServerType* — Apache HTTP Server può essere eseguito solo come `ServerType standalone` rendendo inutile questa direttiva.
- *AccessConfig* e *ResourceConfig* — queste direttive sono state rimosse dato che rispecchiano la funzione della direttiva `Include`. Se avete impostato le direttive `AccessConfig` e `ResourceConfig` dovete sostituirle con le direttive `Include`.

Per garantire che i file vengano letti nell'ordine stabilito dalle vecchie direttive, `Include` devono essere collocate alla fine del file `httpd.conf`, con la direttiva corrispondente a `ResourceConfig` che precede quella corrispondente a `AccessConfig`. Se avete utilizzato i valori predefiniti, dovete includerli in modo esplicito come `conf/srm.conf` e `conf/access.conf`.

14.2.2. Configurazione del server principale

La sezione relativa alla configurazione del server principale del file di configurazione consente di impostare il server principale, che risponde a qualsiasi richiesta che non venga gestita da una definizione `<VirtualHost>`. I valori in questo caso forniscono inoltre valori predefiniti per qualsiasi contenitore `<VirtualHost>` possiate definire.

Le direttive utilizzate in questa sezione sono state modificate in minima parte tra la versione 1.3 e 2.0 di Apache HTTP Server. Se la configurazione del vostro server principale è stata personalizzata notevolmente potrebbe essere più semplice modificare la configurazione esistente perché si adatti ad Apache 2.0. Gli utenti con sezioni del server principale personalizzate solo in parte devono migrare le proprie modifiche nella configurazione di Apache 2.0.

14.2.2.1. Mappatura di UserDir

La direttiva `UserDir` è utilizzata per abilitare URL come `http://example.com/~jim/` per mappare una sottodirectory all'interno della directory home dell'utente `jim`, come `/home/jim/public_html`. Un effetto collaterale di questa caratteristica può consentire a un potenziale malintenzionato di determinare se un determinato nome utente sia presente nel sistema, pertanto la configurazione predefinita di Apache HTTP Server 2.0 disabilita questa direttiva.

Per abilitare la mappatura di `UserDir`, modificate la direttiva in `httpd.conf` da:

```
UserDir disable
```

a quanto riportato di seguito:

UserDir `public_html`

Per ulteriori informazioni su questo argomento, consultate la documentazione indicata di seguito nel sito Web di Apache Software Foundation:

- http://httpd.apache.org/docs-2.0/mod/mod_userdir.html#userdir

14.2.2.2. Accesso

Le direttive di accesso riportate di seguito sono state rimosse:

- AgentLog
- RefererLog
- RefererIgnore

Tuttavia i log `agent` e `referer` sono ancora disponibili mediante l'utilizzo delle direttive `CustomLog` e `LogFormat`.

Per ulteriori informazioni su questo argomento, consultate la documentazione indicata di seguito nel sito Web di Apache Software Foundation:

- http://httpd.apache.org/docs-2.0/mod/mod_log_config.html#customlog
- http://httpd.apache.org/docs-2.0/mod/mod_log_config.html#logformat

14.2.2.3. Indicizzazione delle directory

La direttiva `FancyIndexing` è stata finalmente rimossa. La stessa funzionalità è disponibile attraverso l'opzione `FancyIndexing` all'interno della direttiva `IndexOptions`.

La nuova opzione `VersionSort` della direttiva `IndexOptions` fa sì che i file contenenti i numeri di versione vengano ordinati in modo naturale, cosicché `httpd-2.0.6.tar` viene visualizzato prima di `httpd-2.0.36.tar` nella pagine dell'indice delle directory.

I valori predefiniti per le direttive `ReadmeName` e `HeaderName` sono stati modificati da `README` e `HEADER` a `README.html` e `HEADER.html`.

Per ulteriori informazioni su questo argomento, consultate la documentazione indicata di seguito nel sito Web di Apache Software Foundation:

- http://httpd.apache.org/docs-2.0/mod/mod_autoindex.html#indexoptions
- http://httpd.apache.org/docs-2.0/mod/mod_autoindex.html#readmename
- http://httpd.apache.org/docs-2.0/mod/mod_autoindex.html#headername

14.2.2.4. Negoziazione del contenuto

La direttiva `CacheNegotiatedDocs` richiede ora l'argomento `on` o `off`. Le istanze esistenti di `CacheNegotiatedDocs` devono essere sostituite con `CacheNegotiatedDocs on`.

Per ulteriori informazioni su questo argomento, consultate la documentazione indicata di seguito nel sito Web di Apache Software Foundation:

- http://httpd.apache.org/docs-2.0/mod/mod_negotiation.html#cachenegotiateddocs

14.2.2.5. Documenti degli errori

Per utilizzare un messaggio hard-coded con la direttiva `ErrorDocument`, il messaggio deve essere compreso tra virgolette doppie, invece che semplicemente preceduto dalle virgolette doppie come in Apache 1.3.

Per migrare un'impostazione `ErrorDocument` a Apache HTTP Server 2.0, utilizzate la struttura riportata di seguito:

```
ErrorDocument 404 "The document was not found"
```

Nell'esempio della direttiva `ErrorDocument` si trovano virgolette doppie alla fine.

Per ulteriori informazioni su questo argomento, consultate la documentazione indicata di seguito nel sito Web di Apache Software Foundation:

- <http://httpd.apache.org/docs-2.0/mod/core.html#errordocument>

14.2.3. Configurazione degli host virtuali

Il contenuto di tutti i contenitori `<VirtualHost>` deve essere migrato nello stesso modo della sezione del server principale come descritto in la Sezione 14.2.2.



Importante

La configurazione dell'host virtuale SSL/TLS è stata spostata all'esterno del file di configurazione del server principale nel file `/etc/httpd/conf.d/ssl.conf`.

Per ulteriori informazioni su questo argomento, consultate la documentazione indicata di seguito nel sito Web di Apache Software Foundation:

- <http://httpd.apache.org/docs-2.0/vhosts/>

14.2.4. Moduli e Apache HTTP Server 2.0

In Apache HTTP Server 2.0 il sistema di moduli è stato modificato per consentire ai moduli di essere collegati o combinati in modo nuovo. Gli script CGI, per esempio, possono generare documenti HTML analizzati dal server che possono poi essere elaborati da `mod_include`. In questo modo si apre una vasta gamma di possibilità in relazione al modo in cui i moduli possono essere combinati per raggiungere un obiettivo specifico.

Questo sistema funziona in base al fatto che ciascuna richiesta viene servita da esattamente un modulo *handler* seguito da zero o più moduli *filtro*.

In Apache 1.3, per esempio, uno script PHP sarebbe stato gestito completamente dal modulo PHP. In Apache 2.0 la richiesta viene inizialmente *gestita* dal modulo principale — relativo ai file statici — e viene poi *filtrata* dal modulo PHP.

L'impiego dettagliato di questa e di altre nuove caratteristiche di Apache 2.0 va oltre l'ambito di questo documento. Tuttavia, la modifica ha ramificazioni se è stato utilizzato `PATH_INFO`, che contiene informazioni sul percorso finale dopo il nome di file true in un documento che è gestito da un modulo che viene ora implementato come filtro. Il modulo principale, che gestisce inizialmente la richiesta, non comprende per default `PATH_INFO` e restituirà errori 404 `Not Found` per le richieste

che contengono tali informazioni. Potete utilizzare la direttiva `AcceptPathInfo` per obbligare il modulo principale ad accettare le richieste con `PATH_INFO`. Di seguito è riportato un esempio di questa direttiva:

```
AcceptPathInfo on
```

Per ulteriori informazioni su questo argomento, consultate la documentazione indicata di seguito nel sito Web di Apache Software Foundation:

- <http://httpd.apache.org/docs-2.0/mod/core.html#acceptpathinfo>
- <http://httpd.apache.org/docs-2.0/handler.html>
- <http://httpd.apache.org/docs-2.0/filter.html>

14.2.4.1. Il modulo `mod_ssl`

La configurazione per `mod_ssl` è stata spostata da `httpd.conf` nel file `/etc/httpd/conf.d/ssl.conf`. Perché questo file venga caricato e perché `mod_ssl` funzioni, dovete disporre dell'istruzione `Include conf.d/*.conf` nel vostro file `httpd.conf` come descritto in la Sezione 14.2.1.3.

Le direttive `ServerName` negli host virtuali SSL devono specificare in modo esplicito il numero di porta.

Per esempio, quella riportata di seguito è una direttiva di esempio di Apache HTTP Server 1.3:

```
##
## SSL Virtual Host Context
##

<VirtualHost _default_:443>
    # General setup for the virtual host
    ServerName ssl.host.name
    ...
</VirtualHost>
```

Per migrare questa impostazione a Apache HTTP Server 2.0, utilizzate la struttura riportata di seguito:

```
##
## SSL Virtual Host Context
##

<VirtualHost _default_:443>
    # General setup for the virtual host
    ServerName ssl.host.name:443
    ...
</VirtualHost>
```

Per ulteriori informazioni su questo argomento, consultate la documentazione indicata di seguito nel sito Web di Apache Software Foundation:

- http://httpd.apache.org/docs-2.0/mod/mod_ssl.html
- <http://httpd.apache.org/docs-2.0/vhosts/>

14.2.4.2. Il modulo `mod_proxy`

Le istruzioni di controllo dell'accesso proxy sono ora collocate nel blocco `<Proxy>` invece di `<Directory proxy:>`.

La funzionalità di caching del vecchio file `mod_proxy` è stata suddivisa nei tre moduli riportati di seguito:

- `mod_cache`
- `mod_disk_cache`
- `mod_file_cache`

Questi moduli utilizzano in genere direttive uguali o simili come le versioni più vecchie del modulo `mod_proxy`.

Per ulteriori informazioni su questo argomento, consultate la documentazione indicata di seguito nel sito Web di Apache Software Foundation:

- http://httpd.apache.org/docs-2.0/mod/mod_proxy.html

14.2.4.3. Il modulo `mod_include`

Il modulo `mod_include` è ora implementato come filtro (per ulteriori informazioni sui filtri, consultate la Sezione 14.2.4) ed è pertanto abilitato in modo diverso.

Per esempio, quella riportata di seguito è una direttiva di esempio di Apache HTTP Server 1.3:

```
AddType text/html .shtml
AddHandler server-parsed .shtml
```

Per migrare questa impostazione a Apache HTTP Server 2.0, utilizzate la struttura riportata di seguito:

```
AddType text/html .shtml
AddOutputFilter INCLUDES .shtml
```

Come in precedenza, la direttiva `Options +Includes` è ancora necessaria per la sezione `<Directory>` o in un file `.htaccess`.

Per ulteriori informazioni su questo argomento, consultate la documentazione indicata di seguito nel sito Web di Apache Software Foundation:

- http://httpd.apache.org/docs-2.0/mod/mod_include.html

14.2.4.4. I moduli `mod_auth_dbm` e `mod_auth_db`

Apache HTTP Server 1.3 supportava due moduli di autenticazione, `mod_auth_dbm` e `mod_auth_db`, che utilizzavano rispettivamente database Berkeley e DBM. Questi moduli sono stati combinati in un singolo modulo denominato `mod_auth_dbm` in Apache HTTP Server 2.0, che è in grado di accedere a numerosi formati diversi di database. Per migrare da `mod_auth_db` alla versione 1.3, i file di configurazione devono essere modificati sostituendo `AuthDBUserFile` e `AuthDBGroupFile` con gli equivalenti `mod_auth_dbm`: `AuthDBMUserFile` and `AuthDBMGroupFile`. Dovete inoltre aggiungere la direttiva `AuthDBMType DB` per indicare il tipo di file di database utilizzato.

L'esempio riportato di seguito mostra una configurazione `mod_auth_db` per Apache 1.3:

```
<Location /private/>
  AuthType Basic
```

```
AuthName "My Private Files"
AuthDBUserFile /var/www/authdb
require valid-user
</Location>
```

Per migrare questa impostazione alla versione 2.0 di Apache HTTP Server, utilizzate la struttura riportata di seguito:

```
<Location /private/>
AuthType Basic
AuthName "My Private Files"
AuthDBMUserFile /var/www/authdb
AuthDBMType DB
require valid-user
</Location>
```

La direttiva `AuthDBMUserFile` può anche essere utilizzata nei file `.htaccess`.

Lo script Perl `dbmmanage`, utilizzato per manipolare database di nomi e utenti e password, è stato sostituito da `htdbm` in Apache HTTP Server 2.0. Il programma `htdbm` offre funzionalità equivalenti e come `mod_auth_dbm` può supportare un'ampia serie di formati di database. L'opzione `-T` può essere utilizzata nella riga di comando per specificare il formato da utilizzare.

La Tabella 14-1 mostra come migrare da un database in formato DBM al formato `htdbm` mediante `dbmmanage`.

Azione	Comando <code>dbmmanage</code> (Apache 1.3)	Comando <code>htdbm</code> equivalente (Apache 2.0)
Aggiungere l'utente al database (mediante la password)	<code>dbmmanage authdb add username password</code>	<code>htdbm -b -TDB authdb username password</code>
Aggiungere l'utente al database (richiesta della password)	<code>dbmmanage authdb adduser username</code>	<code>htdbm -TDB authdb username</code>
Rimuovere l'utente dal database	<code>dbmmanage authdb delete username</code>	<code>htdbm -x -TDB authdb username</code>
Elencare gli utenti nel database	<code>dbmmanage authdb view</code>	<code>htdbm -l -TDB authdb</code>
Verificare una password	<code>dbmmanage authdb check username</code>	<code>htdbm -v -TDB authdb username</code>

Tabella 14-1. Migrazione da `dbmmanage` a `htdbm`

Le opzioni `-m` e `-s` funzionano con `dbmmanage` e `htdbm`, attivando l'utilizzo degli algoritmi MD5 o SHA1 per l'hashing delle password.

Quando create un nuovo database con `htdbm`, deve essere utilizzata l'opzione `-c`.

Per ulteriori informazioni su questo argomento, consultate la documentazione indicata di seguito nel sito Web di Apache Software Foundation:

- http://httpd.apache.org/docs-2.0/mod/mod_auth_dbm.html

14.2.4.5. Il modulo `mod_perl`

La configurazione per `mod_perl` è stata spostata da `httpd.conf` nel file `/etc/httpd/conf.d/perl.conf`. Perché questo file venga caricato e perché `mod_perl` funzioni,

doвете disporre dell'istruzione `Include conf.d/*.conf` nel file `httpd.conf` come descritto in la Sezione 14.2.1.3.

Le occorrenze di `Apache::` nel file `httpd.conf` devono essere sostituite con `ModPerl::`. È stato inoltre modificato il modo in cui vengono registrati gli handler.

Quella riportata di seguito è una configurazione Apache HTTP Server 1.3 `mod_perl` di esempio:

```
<Directory /var/www/perl>
    SetHandler perl-script
    PerlHandler Apache::Registry
    Options +ExecCGI
</Directory>
```

Quella riportata di seguito è l'equivalente `mod_perl` per Apache HTTP Server 2.0:

```
<Directory /var/www/perl>
    SetHandler perl-script
    PerlModule ModPerl::Registry
    PerlHandler ModPerl::Registry::handler
    Options +ExecCGI
</Directory>
```

La maggior parte dei moduli per `mod_perl` 1.x deve funzionare senza alcuna modifica con `mod_perl` 2.x. I moduli XS richiedono la ricompilazione e possono necessitare di modifiche Makefile minori.

14.2.4.6. Il modulo `mod_python`

La configurazione per `mod_python`; è stata spostata da `httpd.conf` nel file `/etc/httpd/conf.d/python.conf`. Perché questo file venga caricato e perché `mod_python`; funzioni, doвете disporre dell'istruzione `Include conf.d/*.conf` nel file `httpd.conf` come descritto in la Sezione 14.2.1.3.

14.2.4.7. PHP

La configurazione per PHP è stata spostata da `httpd.conf` nel file `/etc/httpd/conf.d/php.conf`. Perché questo file venga caricato, doвете disporre dell'istruzione `Include conf.d/*.conf` nel file `httpd.conf` come descritto in la Sezione 14.2.1.3.

Il modulo PHP verrà ora implementato come filtro e dovrà pertanto essere abilitato in modo diverso. Per ulteriori informazioni sui filtri, consultate la Sezione 14.2.4.

In Apache HTTP Server 1.3 PHP è stato implementato mediante le direttiva riportate di seguito:

```
AddType application/x-httpd-php .php
AddType application/x-httpd-php-source .phps
```

In Apache HTTP Server 2.0 utilizzate invece le seguenti direttive:

```
<Files *.php>
    SetOutputFilter PHP
    SetInputFilter PHP
</Files>
```

In PHP 4.2.0 e versioni successive la serie prestabilita di variabili predefinite che sono disponibili nell'ambito globale è stata modificata. Il singolo input e le variabili del server non sono più, per default, collocate direttamente in questo punto. Questa modifica può fare in modo che gli script si

interrompano. Potete tornare al comportamento precedente impostando `register_globals` a `On` nel file `/etc/php.ini`.

Per ulteriori informazioni su questo argomento, consultate l'URL indicato di seguito per dettagli relativi alle modifiche di ambito globale:

- http://www.php.net/release_4_1_0.php

14.3. Dopo l'installazione

Dopo avere installato il pacchetto `httpd`, la documentazione relativa a Apache HTTP Server sarà disponibile installando il pacchetto `httpd-manual` e indirizzando un browser Web all'indirizzo `http://localhost/manual/` oppure potete esaminare la documentazione di Apache disponibile nel sito Web all'indirizzo `http://httpd.apache.org/docs-2.0/`.

La documentazione relativa a Apache HTTP Server contiene un elenco e descrizioni complete di tutte le opzioni di configurazione. Per vostra praticità, questo capitolo fornisce brevi descrizioni delle direttive di configurazione utilizzate da Apache HTTP Server 2.0.

La versione di Apache HTTP Server inclusa in Red Hat Linux comprende la possibilità di impostare server Web sicuri utilizzando la potente cifratura SSL fornita dai pacchetti `mod_ssl` e `openssl`. Mentre esaminate i file di configurazione, fate attenzione che sia incluso un server Web sicuro e un server Web sicuro. Quest'ultimo esegue un host virtuale che viene configurato nel file `/etc/httpd/conf.d/ssl.conf`. Per ulteriori informazioni sugli host virtuali, consultate la Sezione 14.8. Per informazioni sulla configurazione di un host virtuale del server sicuro, consultate la Sezione 14.8.2. Per informazioni sulla configurazione del server sicuro HTTP di Apache, consultate il capitolo relativo alla *configurazione del server sicuro HTTP Apache* nella *Official Red Hat Linux Customization Guide*.



Nota Bene

Red Hat, Inc. non include le estensioni di FrontPage, poiché la licenza Microsoft™ vieta l'inserimento delle estensioni in prodotti di terza parte. Per maggiori informazioni sulle estensioni di FrontPage, consultate <http://www.rtr.com/fpsupport/>.

14.4. Avvio e chiusura di `httpd`

L'RPM `httpd` installa lo script `/etc/rc.d/init.d` a cui è possibile accedere tramite il comando `sbin/service`.

Per avviare il server, digitate il comando come root:

```
/sbin/service httpd start
```



Nota Bene

Se state eseguendo Apache HTTP Server come server sicuro, vi viene richiesto di inserire la password.

Per chiudere il server, digitate il comando:

```
/sbin/service httpd stop
```

Il comando `restart` è un modo veloce per chiudere e riavviare il server. Il comando `restart` arresta il server per poi farlo ripartire. Se state eseguendo Apache HTTP Server come server sicuro, vi viene richiesto di inserire la password. Il comando `restart` è simile all'esempio seguente:

```
/sbin/service httpd restart
```

Se avete appena terminato di effettuare delle modifiche nel file `httpd.conf`, non vi occorre chiudere e riavviare il server, ma potete utilizzare il comando `reload`.



Nota Bene

Se state eseguendo Apache HTTP Server come server sicuro, *non* sarà necessario digitare la password per l'utilizzo dell'opzione `reload` dato che la password rimane memorizzata durante il ricaricamento.

Il comando `reload` è simile all'esempio seguente:

```
/sbin/service httpd reload
```

Per default, il processo `httpd` *non* verrà avviato automaticamente all'avvio del computer. Sarà necessario configurare il servizio `httpd` per l'avvio in fase iniziale mediante un'utility `initscript`, come `/sbin/chkconfig`, `/sbin/ntsysv` o il programma **Services Configuration Tool**.

Per ulteriori informazioni relative a questi strumenti, consultate il capitolo relativo al *controllo dei servizi di accesso Services* nella *Official Red Hat Linux Customization Guide*.



Nota Bene

Se state eseguendo Apache HTTP Server come server sicuro, dopo l'avvio vi viene richiesta la password del server, a meno che non abbiate generato una chiave specifica per il server sicuro.

Per informazioni sulla configurazione del server sicuro HTTP di Apache, consultate il capitolo relativo alla *configurazione del server sicuro HTTP Apache* nella *Official Red Hat Linux Customization Guide*.

14.5. Direttive di configurazione in `httpd.conf`

Il file di configurazione di Apache HTTP Server è `/etc/httpd/conf/httpd.conf`, un file ben commentato che, in un certo senso, si spiega da solo. La sua configurazione di default si adatta bene a molteplici situazioni, anche se è comunque necessario acquisire familiarità con alcune delle più importanti opzioni di configurazione.



Avvertimento

Con la versione 2.0 di Apache HTTP Server, molte opzioni di configurazione sono state modificate. Se dovete migrare un file di configurazione della versione 1.3 al nuovo formato, consultate la Sezione 14.2.

Se avete necessità di configurare Apache HTTP Server, modificate `httpd.conf` e ricaricate, riavviate oppure spegnete e riavviate, il processo `httpd`. Per informazioni su come ricaricare, chiudere e avviare Apache HTTP Server, consultate la Sezione 14.4.

Prima di modificare `httpd.conf` si consiglia di fare una copia del file originale, chiamandola per esempio `httpd.conf-old`. Creando un backup, infatti, è possibile rimediare a eventuali errori commessi durante la modifica del nuovo file di configurazione.

Qualora, in seguito a un errore, il server Web non dovesse funzionare correttamente, dovete anzitutto andare a rivedere le modifiche che avete effettuato in `httpd.conf`, e assicurarvi di non aver commesso errori di battitura. Dopodiché controllate il file di log degli errori del server Web (`/var/log/httpd/error_log`). Se non avete molta esperienza, vi può risultare difficile interpretare il log degli errori. Tuttavia, se vi siete appena imbattuti in un problema, le ultime voci elencate in questo file dovrebbero fornire sufficienti informazioni su ciò che può essere avvenuto.

Nelle sezioni che seguono, troverete una breve descrizione delle direttive incluse in `httpd.conf`. Tali descrizioni non sono approfondite. Per reperire maggiori informazioni potete consultare la relativa documentazione, disponibile in formato HTML all'indirizzo <http://localhost/manual/> oppure nel sito Apache all'indirizzo <http://httpd.apache.org/docs-2.0/>. Per ulteriori dettagli sulle direttive `mod_ssl`, consultate la documentazione disponibile all'indirizzo http://localhost/mod/mod_ssl.html o lo *User Manual* oppure nel sito Apache all'indirizzo http://httpd.apache.org/docs-2.0/mod/mod_ssl.html.

14.5.1. ServerRoot

`ServerRoot` è la directory di livello superiore che contiene i file del server. Entrambi i server (sicuro e non sicuro) sono impostati per utilizzare `ServerRoot` di `"/etc/httpd"`.

14.5.2. LockFile

`LockFile` imposta il percorso per il file di lock utilizzato quando viene compilato `httpd` con `USE_FCNTL_SERIALIZED_ACCEPT` o `USE_FLOCK_SERIALIZED_ACCEPT`. È consigliabile lasciare `LockFile` al suo valore di default.

14.5.3. PidFile

`PidFile` è il file nel quale è memorizzato il pid (ID del processo) del server Web. Secondo la configurazione il pid viene memorizzato nella directory `/var/run/httpd.pid`.

14.5.4. ScoreBoardFile

`ScoreBoardFile` immagazzina le informazioni interne del processo server, utilizzate per la comunicazione tra il processo padre e i processi figli. Red Hat Linux si serve della memoria condivisa per memorizzare il comando `ScoreBoardFile`; mentre il valore predefinito di `/etc/httpd/logs/apache_runtime_status` viene utilizzato solo come ripiego.

14.5.5. Timeout

`Timeout` definisce, in secondi, quanto tempo aspetta il server per la ricezione e la trasmissione durante la comunicazione. In particolare, definisce quanto tempo trascorre prima che il server riceva una richiesta GET e prima che riceva pacchetti TCP su richiesta POST o PUT, nonché il tempo di attesa perché gli ACK rispondano ai pacchetti TCP. `Timeout` è impostato per attendere 300 secondi, ossia il tempo adeguato per la maggior parte delle situazioni.

14.5.6. KeepAlive

`KeepAlive` stabilisce se il server consentirà più di una richiesta per connessione o, in altri termini, le connessioni persistenti. `KeepAlive` può essere usato per evitare che un client utilizzi troppe risorse del server.

Per default, `KeepAlive` è impostato su `off`. Se è impostato su `on` e il server è particolarmente occupato, il server può produrre rapidamente il numero più elevato di processi figli. In tal caso, il server diventa piuttosto lento. Se abilitate `KeepAlive`, dovete impostare il `KeepAliveTimeout` (vedere la Sezione 14.5.8) su valori bassi e controllare il server `/var/log/httpd/error_log`. Questo file di log vi consente di sapere se il server è carente di processi figli.

14.5.7. MaxKeepAliveRequests

La direttiva imposta il numero massimo di richieste accettate su ogni connessione persistente. Il team di sviluppo di Apache consiglia di impostare un valore alto, al fine di migliorare le prestazioni del server. `MaxKeepAliveRequests` è impostato per default su 100, un valore che si adatta alla maggior parte delle possibili situazioni.

14.5.8. KeepAliveTimeout

`KeepAliveTimeout` imposta (in secondi) il tempo durante il quale il server attende una nuova richiesta prima di chiudere la connessione. Una volta ricevuta la richiesta, si applica invece la direttiva `Timeout`.

14.5.9. MinSpareServers e MaxSpareServers

Apache HTTP Server si adatta dinamicamente al carico di lavoro mantenendo un numero appropriato di processi di riserva del server, a seconda del traffico. Viene eseguita la verifica del numero di server in attesa di richiesta che vengono eliminati se superano il valore di `MaxSpareServers` oppure vengono creati se sono inferiori al valore di `MinSpareServers`.

Il valore di default di `MinSpareServers` è 5, mentre quello di `MaxSpareServers` è 20. Queste impostazioni di default sono adatte alla maggior parte delle possibili situazioni. Si consiglia di non aumentare troppo il valore di `MinSpareServers`, poiché si rischia di sovraccaricare il server quando il traffico non è eccessivo.

14.5.10. StartServers

`StartServers` imposta il numero di processi server che devono essere creati all'avvio del servizio. Poiché il server Web elimina o crea dinamicamente i processi server in funzione del traffico, non è necessario cambiare questo parametro. Il vostro server Web è impostato per far partire 8 processi all'avvio.

14.5.11. MaxClients

`MaxClients` imposta un limite per il numero totale di processi server (e quindi anche di client connessi contemporaneamente) in esecuzione. Si consiglia di mantenere alto il valore di `MaxClients` (di default 150), in modo che, una volta raggiunto il limite massimo di client connessi in contemporanea, non saranno possibili ulteriori connessioni. Non è possibile impostare un valore superiore a 256 senza ricompilare Apache. La ragione principale per cui conviene limitare il numero di connessioni simultanee è quella di evitare che si verifichi un crash del sistema operativo.

14.5.12. MaxRequestsPerChild

`MaxRequestsPerChild` imposta il numero massimo di richieste che ogni processo figlio può gestire prima che sia eliminato. Lo scopo principale di `MaxRequestsPerChild` è quello di evitare che un processo rimanga in esecuzione troppo a lungo, occupando un'eccessiva quantità di memoria. Il valore di default è 1000.

14.5.13. Listen

Il comando `Listen` specifica su quale porta il server Web riceve le richieste. Per default il server Web attende le richieste sulla porta 80 per la comunicazione Web non sicura e (nei tag relativi all'host virtuale che definiscono il server sicuro) sulla porta 443 per la comunicazione sicura.

Se impostate Apache HTTP Server perché attenda su una porta inferiore alla 1024, dovete eseguire il processo `httpd` come `root`. Per le porte superiori alla 1024, potete eseguirlo come un utente normale.

`Listen` può inoltre essere usato per specificare particolari indirizzi IP dai quali il server accetta le connessioni.

14.5.14. Include

`Include` consente ad altri file di configurazione di essere inclusi in fase di runtime.

Per percorso per questi file di configurazione può essere assoluto o relativo per il comando `ServerRoot`.



Importante

Perché il server utilizzi moduli in singoli pacchetti, come `mod_ssl`, `mod_perl` e `php`, la direttiva riportata di seguito deve trovarsi in `Section 1: Global Environment` del file `http.conf`:

```
Include conf.d/*.conf
```

14.5.15. LoadModule

`LoadModule` è utilizzata per caricare i moduli Dynamic Shared Object (DSO, oggetti condivisi dinamicamente). Maggiori informazioni sul supporto DSO di Apache HTTP Server e sull'esatto modo di impiego della direttiva `LoadModule` sono disponibili nella Sezione 14.7. L'ordine di caricamento dei moduli *non è più importante* in Apache HTTP Server 2.0. Per ulteriori informazioni sul supporto di Apache HTTP Server 2.0 per il DSO, consultate la Sezione 14.2.1.3.

14.5.16. IfDefine

I tag `<IfDefine> c </IfDefine>` utilizzano le direttive di configurazione specificate al loro interno se nel primo tag la definizione "test" risulta essere vera. Le direttive vengono ignorate se il test è falso.

Il test nei tag `<IfDefine>` è il nome di un parametro (per esempio, `HAVE_PERL`). Se il parametro è definito (ossia viene fornito come argomento del comando dell'avvio del server), allora il test è "vero". In questo caso il server Web viene attivato e le direttive contenute nei tag `IfDefine` vengono applicate.

I tag `<IfDefine HAVE_SSL>` contengono per default la configurazione dell'host virtuale per il server Web sicuro. I tag `<IfDefine HAVE_SSL>` contengono anche le direttive `LoadModule` e `AddModule` per il modulo `ssl_module`.

14.5.17. `ExtendedStatus`

La direttiva `ExtendedStatus` verifica se Apache genera le informazioni di base (`off`) o dettagliate (`on`) sullo stato del server quando viene chiamato l'handler `server-status` mediante il tag `Location`. Per maggiori informazioni consultate la Sezione 14.5.65.

14.5.18. `User`

La direttiva `User` imposta lo `userid` utilizzato dal server per rispondere alle richieste. L'impostazione della direttiva `User` determina gli accessi al server. Qualunque file non accessibile all'utente specificato non sarà distribuito via Web. L'utente di default per `User` è `apache`.

L'utente `User` dovrebbe avere i privilegi per accedere solamente ai file visibili via Web. I processi CGI vengono eseguiti con i diritti di `User`. L'utente `User` non dovrebbe essere autorizzato a eseguire alcun codice che non sia inteso come risposta alle richieste HTTP.



Nota Bene

Per ragioni di sicurezza, Apache HTTP Server non funzionerà come `User root`. Impostare `User` come `root` ridurrà notevolmente la sicurezza del vostro server Web.

Il processo padre `httpd` viene dapprima eseguito come `root` durante le normali operazioni, ma viene poi immediatamente rinvio all'utente `apache`. Il server deve essere eseguito con i diritti di `root` per collegarsi a una porta inferiore alla 1024 (la comunicazione standard sicura per il Web è sulla porta 443, mentre la comunicazione standard non sicura è sulla porta 80). Le porte inferiori alla 1024 sono riservate all'uso del sistema, perciò sono accessibili solo dall'utente `root`. Quando il server si è collegato alla propria porta, comunque, rinvia il processo al comando `User` prima di accettare qualsiasi richiesta di connessione.

14.5.19. `Group`

La direttiva `Group` è analoga alla direttiva `User`. `Group` imposta il gruppo in base al quale il server risponderà alle richieste. Il gruppo di default è `apache`.

14.5.20. `ServerAdmin`

`ServerAdmin` è l'indirizzo di posta elettronica dell'amministratore del server Web. Questo indirizzo di posta elettronica appare nei messaggi di errore delle pagine Web generate dal server, in modo che gli utenti possano riferire eventuali problemi inviando un messaggio all'amministratore del server. `ServerAdmin` è impostato per default su `root@localhost`.

Normalmente alla direttiva `ServerAdmin` viene attribuito il valore `webmaster@your_domain.com`. Assegnate dunque l'alias `webmaster` alla persona responsabile del server Web nel file `/etc/aliases`. Infine, eseguite `/usr/bin/newaliases` per aggiungere il nuovo alias.

14.5.21. ServerName

Potete usare la direttiva `ServerName` per attribuire un nome host e un numero di porta (corrispondenti alla direttiva `Listen`) per il server. Il nome non deve corrispondere al nome reale del computer host. Per esempio, potete usare `www.your_domain.com` se il nome reale del vostro server è `foo.your_domain.com`. `ServerName` deve essere un nome valido per il vostro DNS (Domain Name Service) affinché tutto funzioni correttamente (evitate di inserire nomi inventati). Per esempio:

```
ServerName www.your_domain.com:80
```

Se specificate un `ServerName`, accertatevi che nel file `/etc/hosts` esista una corrispondenza tra indirizzo IP e nome del server.

14.5.22. DocumentRoot

`DocumentRoot` è la directory che contiene la maggior parte dei file HTML in risposta alle richieste. La directory `DocumentRoot` di default per entrambi i server Web (sicuro e non sicuro) è `/var/www/html`. Per esempio, il server Web può ricevere una richiesta per il documento seguente:

```
http://vostro_dominio/foo.html
```

Il server cercherà il file riportato di seguito nella directory di default:

```
/var/www/html/foo.html
```

Se desiderate modificare la direttiva `DocumentRoot` in modo che non sia condivisa dal server Web non sicuro e da quello sicuro, consultate la Sezione 14.8.

14.5.23. Directory

I tag `<Directory /path/to/directory> e </Directory>` sono usati per raggruppare un insieme di direttive di configurazione da applicare solo a una particolare directory e a tutte le sue sottodirectory. Tutte le direttive applicabili a una directory possono essere usate all'interno dei tag `<Directory>`. Allo stesso modo, i tag `<File>` possono essere utilizzati in applicazione a uno o più file specifici.

Per default, alla directory root vengono applicati i parametri più restrittivi, tramite le direttive `Options` (consultate la Sezione 14.5.24) e `AllowOverride` (consultate la Sezione 14.5.25). Con questa configurazione, alle directory che necessitano di impostazioni meno restrittive devono essere assegnate in modo esplicito quelle impostazioni.

Usando i tag `Directory`, `DocumentRoot` avrà parametri meno restrittivi, in modo tale da poter effettuare le richieste HTTP.

La directory `cgi-bin` è impostata in modo da permettere l'esecuzione degli script CGI tramite l'opzione `ExecCGI`. Se dovete eseguire degli script CGI presenti in un'altra directory, impostate la direttiva `ExecCGI` per quella directory. Per esempio, se la vostra directory `cgi-bin` è `/var/www/cgi-bin`, ma volete eseguire anche gli script CGI presenti in `/home/my_cgi_directory`, aggiungete una direttiva `ExecCGI` a una serie di direttive `Directory` come quelle che seguono nel file `httpd.conf`:

```
<Directory /home/my_cgi_directory>
    Options +ExecCGI
</Directory>
```

Per poter eseguire gli script CGI presenti in `/home/my_cgi_directory`, oltre a impostare `ExecCGI` è necessario compiere ulteriori operazioni. Innanzitutto, eliminate il commento dalla direttiva `AddHandler` per identificare i file con estensione `.cgi` come script CGI. Per sapere come impostare

`AddHandler` consultate la Sezione 14.5.59. I permessi per gli script CGI e l'intero percorso vanno impostati su `0755`.

14.5.24. Options

La direttiva `Options` verifica le caratteristiche del server disponibili in una particolare directory. Per esempio, con i parametri restrittivi specificati per la directory root, la direttiva `Options` è impostata solo per `FollowSymLinks`: quindi non sono autorizzate altre caratteristiche, tranne quella che consente al server di seguire i link simbolici nella directory root.

Per default, nella directory `DocumentRoot`, la direttiva `Options` è configurata per contenere `Indexes`, `Includes` e `FollowSymLinks`. `Indexes` permette al server di generare un elenco di directory per una directory, se non è specificata alcuna direttiva `DirectoryIndex` (per esempio: `index.html`). `Includes` indica che sono autorizzati gli include lato server. `FollowSymLinks` consente al server di seguire i link simbolici in questa directory.

È inoltre necessario inserire la direttiva `Options` per le directory degli host virtuali, se desiderate che questi riconoscano quelle particolari caratteristiche.

Mediante la linea `Options Includes`, contenuta all'interno della sezione delle direttive `<Directory "/var/www/html">`, gli include lato server sono già abilitati. Tuttavia, se desiderate che un host virtuale riconosca l'autorizzazione degli include lato server, dovrete inserire nei tag del vostro host virtuale una sezione simile alla seguente:

```
<Directory /var/www/html>
Options Includes
</Directory>
```

14.5.25. AllowOverride

La direttiva `AllowOverride` consente di stabilire se le `Options` possono essere ridefinite dalle dichiarazioni presenti in un file `.htaccess`. Per default, sia nella directory root, sia nella directory `DocumentRoot` non è permesso sovrascrivere il file `.htaccess`.

14.5.26. Order

La direttiva `Order` controlla semplicemente l'ordine di valutazione delle direttive `allow` e `deny`. Il server è configurato per valutare prima le direttive `Allow` per la directory `DocumentRoot`.

14.5.27. Allow

`Allow` specifica quali richieste possono accedere a una determinata directory. Il campo richiedente può essere `all`, un nome di dominio, un indirizzo IP, una parte dell'indirizzo IP, una coppia rete/netmask e così via. La directory `DocumentRoot` è configurata per permettere l'accesso da `all` (ossia da tutti i client).

14.5.28. Deny

`Deny` funziona esattamente come `allow`, ma specifica quali accessi negare. Per default, la vostra `DocumentRoot` non è configurata per negare alcuna richiesta.

14.5.29. UserDir

UserDir è il nome di una sottodirectory della home di ciascun utente, in cui vanno collocati i file HTML personali che il server Web utilizza.

Per default, la sottodirectory è `public_html`. Per esempio, il server può ricevere la richiesta seguente:

```
http://vostro_dominio/~nomeutente/foo.html
```

Il server cerca il file:

```
/home/username/public_html/foo.html
```

Nell'esempio precedente `/home/username` è la directory home dell'utente (ovviamente il percorso di default della directory home può essere diverso sul vostro sistema).

Assicuratevi che i permessi della directory home dell'utente siano corretti. L'impostazione esatta è 0711. È necessario attivare i bit di lettura (r) e di esecuzione (x) nella directory `public_html` (anche 0755 funziona correttamente). I file presenti in questa directory devono essere impostati almeno su 0644.

Questa direttiva è impostata a `disable` per default.

14.5.30. DirectoryIndex

La direttiva `DirectoryIndex` è la pagina predefinita che viene restituita al client quando un utente richiede l'indice di una directory, specificando uno slash (/) dopo il nome della directory.

Quando un utente richiede la pagina `http://vostro_dominio/questa_directory/`, riceve la pagina `DirectoryIndex`, se presente, o un elenco di directory generato dal server. La configurazione di default di `DirectoryIndex` è `index.html index.htm index.shtml index.php index.php4 index.php3 index.cgi`. Il server cerca di individuare uno di questi file e restituisce il primo file che trova. Se non trova nessun file, e per questa directory è impostata la direttiva `Options Indexes`, il server genera e restituisce un elenco delle sottodirectory e dei file contenuti nella directory in formato HTML.

14.5.31. AccessFileName

`AccessFileName` attribuisce un nome al file che il server utilizza per accedere alle informazioni di controllo in ogni directory. In genere, il server Web è impostato di default per utilizzare `.htaccess`, se disponibile, per le informazioni sul controllo degli accessi di ciascuna directory.

Immediatamente dopo la direttiva `AccessFileName`, una serie di tag `Files` controlla l'accesso ai file che iniziano con `.ht`. Per ragioni di sicurezza queste direttive negano l'accesso Web a qualunque file `.htaccess` (o a qualunque altro file che inizi con `.ht`).

14.5.32. CacheNegotiatedDocs

Per default, il server Web chiede ai server proxy di non conservare nella cache i documenti trasmessi in base al contenuto (ovvero quei documenti che potrebbero essere modificati mediante l'inserimento del richiedente). Eliminando il commento dalla direttiva `CacheNegotiatedDocs` viene disabilitata questa funzione e i server proxy sono autorizzati a conservare i documenti nella cache.

14.5.33. UseCanonicalName

Per default, `UseCanonicalName` è impostato su `on`. `UseCanonicalName` permette a un server di creare un URL che faccia riferimento a sé stesso, utilizzando `ServerName` e `Port`. Quando il server fa riferimento a sé stesso per rispondere alle richieste di un client, utilizza questo URL. Se, però, impostate `UseCanonicalName` su `off`, il server utilizza il valore inviato dal client per fare riferimento a sé stesso.

14.5.34. TypesConfig

`TypesConfig` definisce il nome del file che imposta le mappature dell'elenco predefinito dei tipi MIME (le estensioni dei file per i tipi di contenuto). Il file `TypesConfig` di default è `/etc/mime.types`. Invece di modificare questo file, si consiglia di aggiungere i tipi MIME tramite la direttiva `AddType`.

Per maggiori informazioni su questa direttiva, consultate la Sezione 14.5.58.

14.5.35. DefaultType

`DefaultType` definisce il tipo MIME di default per i documenti non riconosciuti. Secondo la configurazione di default, se il server Web non riconosce un tipo di file, lo considera come un file in formato testo.

14.5.36. IfModule

I tag `<IfModule>` e `</IfModule>` raggruppano le direttive condizionali. Le direttive presenti all'interno dei tag `IfModule` vengono elaborate solo a una di queste due condizioni: se il modulo specificato all'interno del tag `<IfModule>` è caricato in `httpd`, se il simbolo "!" (punto esclamativo) è inserito prima del nome del modulo oppure solo se il modulo *non* è compilato nel primo tag `<IfModule>`.

Il file `mod_mime_magic.c` è incluso nei tag `IfModule`. Il modulo `mod_mime_magic` è simile al comando `file` di UNIX, che legge i primi byte del contenuto di un file e utilizza "numeri magici" e altre istruzioni per stabilirne il tipo MIME.

Se il modulo `mod_mime_magic` viene compilato per Apache HTTP Server, i tag `IfModule` indicano al modulo dove trovare il file delle istruzioni: in questo caso `/usr/share/magic`.

Il modulo `mod_mime_magic` non è compilato per default. Se desiderate utilizzarlo, consultate le informazioni su come aggiungere i moduli al vostro server contenute in la Sezione 14.7.

14.5.37. HostnameLookups

`HostnameLookups` può essere impostato su `on`, `off` oppure `double`. Se autorizzate la direttiva `HostnameLookups` (impostandola su `on`), il vostro server risolve in modo automatico l'indirizzo IP per ogni connessione che richiede un documento dal server Web. Risolvere l'indirizzo IP significa che il vostro server si connette una o più volte al DNS per individuare il nome dell'host che corrisponde a un particolare indirizzo IP. Se impostate `HostnameLookups` su `double`, il server esegue un DNS inverso doppio. In altre parole, dopo un lookup "inverso" viene eseguito un lookup "diritto". Almeno uno degli indirizzi IP nel lookup diritto deve coincidere con l'indirizzo derivante dal primo lookup inverso.

Di norma dovrete lasciare `HostnameLookups` impostato su `off`, poiché le richieste del DNS comportano un carico eccessivo che può causare un rallentamento del server. Se il server è occupato, gli effetti di `HostnameLookups` sono evidenti.

`HostnameLookups` coinvolge anche Internet nel suo complesso perché si sommano tutte le connessioni individuali create per cercare ogni nome host. Dunque, per evitare problemi con il server Web e per il bene di Internet in generale, è consigliabile lasciare la direttiva `HostnameLookups` impostata su `off`.

Se desiderate vedere i nomi host nei vostri file di log, dovrete eseguire uno dei tanti tool di analisi dei log in grado di effettuare lookup DNS in modo più efficiente e su scala più ampia nel momento in cui ruotate i file di log.

14.5.38. ErrorLog

`ErrorLog` specifica il nome del file dove vengono registrati tutti gli errori del server. Per default, il file di log degli errori è `/var/log/httpd/error_log`.

Il file di log degli errori è il miglior posto dove cercare eventuali errori generati dal server Web o per capire la causa dei malfunzionamenti.

14.5.39. LogLevel

`LogLevel` definisce il livello di dettaglio dei messaggi d'errore registrati nel file di log. `LogLevel` può essere impostato (dal più dettagliato al meno dettagliato) `suemerg`, `alert`, `crit`, `error`, `warn`, `notice`, `info` o `debug`. Per default, è impostato su `warn`.

14.5.40. LogFormat

Le direttive `LogFormat` nel file `httpd.conf` definiscono il formato dei messaggi nel log di accesso. La direttiva `LogFormat` che verrà utilizzata dipende dalle impostazioni attribuite nella direttiva `CustomLog` (consultate la Sezione 14.5.41).

14.5.41. CustomLog

`CustomLog` indica il file di log e il suo formato. Nella configurazione di default del vostro server Web, `CustomLog` definisce il file di log in cui sono registrati gli accessi al server Web: `/var/log/httpd/access_log`. Se desiderate generare delle statistiche di accesso al sito Web, dovete sapere dove è memorizzato questo file.

`CustomLog` definisce anche il formato del file di log da combinare. Il formato standard è:

```
remotehost rfc931 authuser [date] "request" status bytes referer
user-agent
```

remotehost

Contiene il nome dell'host remoto. Se il nome non è disponibile tramite il DNS oppure se `HostnameLookups` è impostato su `Off`, il campo `remotehost` conterrà l'indirizzo IP dell'host remoto.

rfc931

Non è usato. In questo punto del file di log viene inserito il carattere `-`.

authuser

Se viene richiesta l'autenticazione, allora indica il nome che identifica l'utente. Di solito non viene usato e al suo posto viene inserito il carattere `-`.

[date]

Data e ora della richiesta.

"request"

Stringa contenente la stessa richiesta inviata dal browser o dal client.

status

Codice dello stato HTTP restituito al browser o al client.

bytes

Dimensione del documento.

referer

Fornisce l'URL della pagina Web che si è collegata alla richiesta corrente.

user-agent

Fornisce il nome del browser o del client che effettua la richiesta.

14.5.42. *ServerSignature*

La direttiva *ServerSignature* aggiunge una linea contenente la versione del server Apache e il *ServerName* del server host a qualsiasi documento generato dal server (per esempio, i messaggi di errore rispediti ai client). Per default, *ServerSignature* è impostato su *on*. Se non volete aggiungere queste informazioni, impostatelo su *off*. Potete anche impostarlo su *Email*. In tal caso verrà aggiunto un tag HTML `mailto:ServerAdmin`.

14.5.43. *Alias*

L'impostazione *Alias* consente alle directory di trovarsi al di fuori di *DocumentRoot* pur essendo comunque accessibili dal server Web. Qualunque URL che termina con l'alias viene automaticamente risolto nel percorso dell'alias. Nella configurazione di default è già presente un alias. Il server Web può accedere a una directory *icons*, anche se la directory non si trova in *DocumentRoot*. La directory *icons* è in realtà un alias, e non `/var/www/html/icons/`.

14.5.44. *ScriptAlias*

L'impostazione *ScriptAlias* definisce dove sono localizzati gli script CGI (o gli altri tipi di script). Normalmente è meglio non lasciare gli script CGI all'interno di *DocumentRoot*, poiché potrebbero essere visualizzati come documenti di testo. Se anche non vi interessasse il fatto che qualcuno veda (e usi) i vostri script CGI, qualche malintenzionato potrebbe scoprire come funzionano e approfittarsi di eventuali "falle" negli script, mettendo a repentaglio la sicurezza del vostro server. La directory *cgi-bin* è, di default, uno *ScriptAlias* della directory `/cgi-bin/` e si trova in `/var/www/cgi-bin/`. Per la directory `/var/www/cgi-bin` è stata impostata la direttiva *Options ExecCGI*, che abilita l'esecuzione degli script presenti nella directory.

Per maggiori informazioni sull'esecuzione di script CGI in directory diverse da *cgi-bin*, consultate la Sezione 14.5.59 e la Sezione 14.5.23.

14.5.45. Redirect

Quando una pagina Web viene spostata, la direttiva `Redirect` può essere utilizzata per rimappare il vecchio URL con quello nuovo. Il formato è il seguente:

```
Redirect /percorso/foo.html
http://nuovo_dominio/percorso/foo
.html
```

Se si riceve una richiesta HTTP per la pagina `http://vostro_dominio/percorso/foo.html`, il server restituisce la pagina `http://nuovo_dominio/percorso/foo.html` al client, che di solito cerca di richiamare il documento dal nuovo URL.

Per operazioni più avanzate, potete utilizzare il modulo `mod_rewrite` incluso nel server.

14.5.46. IndexOptions

`IndexOptions` controlla l'aspetto degli elenchi delle directory generati dal server aggiungendo icone e descrizioni dei file e così via. Se è impostata la direttiva `Options Indexes` (consultate la Sezione 14.5.24), il server Web genera un elenco delle directory quando riceve una richiesta HTTP come la seguente:

```
http://vostro_dominio/questa_directory/
```

Innanzitutto, il server Web cerca nella directory uno dei file specificati con la direttiva `DirectoryIndex` (di solito, `index.html`). Se il server non trova uno di quei file, crea un elenco HTML dei file e delle sottodirectory presenti nella directory. Potete modificare l'aspetto di questa directory usando determinate direttive contenute nel file `httpd.conf`, tra cui la direttiva `IndexOptions`.

La configurazione di default imposta `FancyIndexing` su `on`. Se `FancyIndexing` è attivato, facendo clic sull'intestazione della colonna potrete organizzare l'ordinamento della visualizzazione secondo quell'intestazione. Con un altro clic sulla stessa intestazione si inverte l'ordine da ascendente a discendente. `FancyIndexing` visualizza un'icona diversa per ogni tipo di file, a seconda dell'estensione. Se utilizzate la direttiva `AddDescription` e activate `FancyIndexing`, viene visualizzata una breve descrizione dei file nell'elenco delle directory generato dal server.

`IndexOptions` ha vari parametri per selezionare l'aspetto degli elenchi delle directory generate dal server. I parametri possibili sono `IconHeight` e `IconWidth`, che permettono al server di includere i tag `HEIGHT` e `WIDTH` per le icone delle pagine Web generate dal server. `IconsAreLinks` gli consente invece di utilizzare le icone come parte del link HTML, insieme al nome del file e così via.

14.5.47. AddIconByEncoding

Questa direttiva associa un'icona a un particolare tipo di file secondo la codifica MIME negli elenchi di directory generati dal server. Per esempio, il server Web mostra, per default, l'icona `compressed.gif` accanto ai file di tipo `x-compress` e `x-gzip` con codifica MIME.

14.5.48. AddIconByType

Questa direttiva specifica il nome dell'icona da visualizzare accanto al file con il tipo MIME negli elenchi delle directory generati dal server. Per esempio, il vostro server è impostato per visualizzare l'icona `text.gif` accanto al file con il tipo MIME di testo.

14.5.49. AddIcon

`AddIcon` specifica l'icona da visualizzare negli elenchi delle directory generati dal server per certi tipi di file o per i file che hanno determinate estensioni. Per esempio, il vostro server Web è impostato in modo da mostrare l'icona `binary.gif` per i file con estensione `.bin` o `.exe`.

14.5.50. DefaultIcon

`DefaultIcon` specifica l'icona da visualizzare negli elenchi delle directory generati dal server per i file che non hanno altre icone specificate. La direttiva `DefaultIcon` predefinita per quei file è `unknown.gif`.

14.5.51. AddDescription

Potete utilizzare la direttiva `AddDescription` per visualizzare il testo specificato per certi file negli elenchi delle directory generati dal server (dovrete anche abilitare `FancyIndexing` come una direttiva `IndexOptions`). Per specificare i file a cui bisogna applicare questa direttiva, potete nominare file specifici, espressioni con asterisco o estensioni di file. Per esempio potete utilizzare la linea seguente:

```
AddDescription "A file that ends in .ni" .ni
```

In ogni elenco generato dal server Web, tutti i file con l'estensione `.ni` avranno la descrizione `A file that ends in .ni` dopo il nome del file. È inoltre necessario attivare la direttiva `FancyIndexing`.

14.5.52. ReadmeName

`ReadmeName` definisce il nome del file (se presente nella directory) che viene aggiunto alla fine degli elenchi delle directory generati dal server. Il server Web cercherà prima di includere il file come documento HTML, poi come file di testo. Nella configurazione di default, `ReadmeName` è impostato su `README`.

14.5.53. HeaderName

`HeaderName` specifica il nome del file (se presente nella directory) che viene inserito all'inizio degli elenchi delle directory generati dal server. Come per `ReadmeName`, il server cerca, se possibile, di includere il file in formato HTML o altrimenti in formato testo.

14.5.54. IndexIgnore

`IndexIgnore` elenca estensioni di file, parti di nomi di file, espressioni con asterisco o nomi di file completi. Il server Web non include nessun file che corrisponda a quei parametri negli elenchi delle directory generati dal server.

14.5.55. AddEncoding

`AddEncoding` definisce le estensioni dei file che hanno una particolare codifica. `AddEncoding` può essere utilizzato anche per indicare ai browser (non a tutti) di decomprimere certi file mentre vengono scaricati.

14.5.56. AddLanguage

`AddLanguage` associa l'estensione di un file a una particolare lingua. Questa direttiva è particolarmente utile quando il server restituisce un documento in base alla lingua di preferenza del client specificata nel browser.

14.5.57. LanguagePriority

`LanguagePriority` vi permette di stabilire una lingua prioritaria in cui trasmettere i file se il client non ha specificato alcuna preferenza nel browser.

14.5.58. AddType

Usate la direttiva `AddType` per associare un tipo MIME a una estensione di file. Per esempio, se state utilizzando il linguaggio PHP4, il vostro server Web sta utilizzando la direttiva `AddType` per riconoscere i file con estensione PHP (`.php4`, `.php3`, `.phtml` o `.php`) come tipi MIME PHP. La seguente direttiva indica a Apache HTTP Server di riconoscere l'estensione `.shtml`:

```
AddType text/html .shtml
AddHandler server-parsed .shtml
```

Dovete includere questa linea all'interno dei tag dell'host virtuale per ogni host virtuale che deve autorizzare gli include lato server.

14.5.59. AddHandler

`AddHandler` mappa l'estensione di un file per handler specifici. Per esempio, l'handler `cgi-script` può essere utilizzato in associazione con l'estensione `.cgi` per trattare un file che termina con `.cgi` automaticamente come script CGI. Questo metodo funziona anche al di fuori della directory `ScriptAlias`, purché seguiate attentamente le istruzioni qui riportate.

Nel file `httpd.conf` è contenuta una linea `AddHandler CGI`:

```
AddHandler cgi-script .cgi
```

È necessario innanzitutto eliminare il commento dalla linea, dopodiché Apache eseguirà gli script CGI per i file che terminano in `.cgi`, anche se sono al di fuori di `ScriptAlias`, impostata di default per localizzare la vostra directory `/cgi-bin/` in `/var/www/cgi-bin/`.

Dovete inoltre configurare `ExecCGI` come `Options` per ogni directory che contiene uno script CGI. Consultate la Sezione 14.5.23 per maggiori informazioni sulla configurazione della directory `ExecCGI`. Accertatevi che i permessi siano impostati correttamente per gli script CGI e per le directory che li contengono. Gli script CGI e l'intero percorso della directory devono essere impostati su 0755.

`AddHandler` deve essere inserito nella configurazione del vostro `VirtualHost`, se state utilizzando host virtuali e volete che riconoscano anche gli script CGI che si trovano al di fuori di `ScriptAlias`.

Oltre agli script CGI, il vostro server Web utilizza anche `AddHandler` per elaborare gli HTML e i file `imagemap` analizzati dal server.

14.5.60. Action

`Action` vi permette di associare un contenuto MIME a uno script CGI. In questo modo, quando viene richiesto un file di quel tipo viene eseguito un particolare script CGI.

14.5.61. MetaDir

MetaDir specifica il nome della directory in cui il vostro server Web deve cercare i file che contengono meta informazioni (ulteriori intestazioni HTTP extra) da includere nella preparazione di documenti.

14.5.62. MetaSuffix

MetaSuffix specifica il suffisso del file contenente meta informazioni (ulteriori intestazioni HTTP), che dovrebbe trovarsi nella directory MetaDir.

14.5.63. ErrorDocument

Per default, in caso di problemi o errori, il vostro server Web mostra un messaggio di errore semplice (e di solito criptato) al client richiedente. Al posto dell'impostazione di default, potete utilizzare la direttiva ErrorDocument per personalizzare il messaggio da visualizzare in caso di errore o reindirizzare il client verso una URL locale o esterno. ErrorDocument associa semplicemente il codice HTTP di risposta a un URL o a un messaggio che verrà restituito al client.



Importante

È *necessario* includere il messaggio di errore tra virgolette doppie perché sia valido.

14.5.64. BrowserMatch

La direttiva BrowserMatch consente al server di definire le variabili di ambiente e/o le azioni sulla base del campo dell'intestazione HTTP User-Agent, che identifica il browser del client. Per default, il vostro server Web utilizza BrowserMatch per negare le connessioni a determinati browser con problemi noti e anche per disabilitare i keepalive e i comandi di annullamento delle intestazioni HTTP per i browser che hanno problemi con queste azioni.

14.5.65. Location

I tag <Location> e </Location> vi consentono di specificare il controllo dell'accesso in base all'URL.

Se desiderate consentire agli utenti di connettersi dal vostro dominio per visualizzare i report sullo stato del server, dovete eliminare i commenti della sezione di direttive riportata di seguito:

```
#<Location /server-status>
#   SetHandler server-status
#   Order deny,allow
#   Deny from all
#   Allow from .your_domain.com
#</Location>
```

Dovete sostituire `.your_domain.com` con il nome del dominio di secondo livello.

Se desiderate fornire report sulla configurazione del server (inclusi i moduli installati e le direttive di configurazione) da richiedere all'interno del vostro dominio, dovete eliminare i commenti dalle linee riportate di seguito:

```
#<Location /server-info>
#   SetHandler server-info
#   Order deny,allow
#   Deny from all
#   Allow from .your_domain.com
#</Location>
```

Ancora una volta dovete inserire `.your_domain.com`.

14.5.66. ProxyRequests

Se eliminate il commento dai tag `IfModule` delle direttive `ProxyRequests`, il vostro Apache HTTP Server svolgerà anche le funzioni di un server proxy. È inoltre necessario caricare il modulo `mod_proxy`. Per informazioni su come caricare i moduli, consultate la Sezione 14.7.

14.5.67. ProxyVia

Il comando `ProxyVia` controlla se la linea relativa all'intestazione di HTTP Via: viene inviata con le richieste o le risposte attraverso il server proxy Apache. L'intestazione Via: mostra il nome dell'host se `ProxyVia` è impostato su `On` e il nome dell'host e la versione di Apache HTTP Server se è impostato su `Full`. Qualsiasi linea Via: viene trasmessa senza essere modificata se è impostato su `Off` e viene invece rimossa se è impostato su `Block`.

14.5.68. Direttive della cache

Una serie di direttive relative alla cache sono commentate nei tag `IfModule` del proxy menzionate sopra. Se state utilizzando la funzionalità server proxy e desiderate abilitare anche la cache proxy, dovete eliminare il commento dalle direttive della cache come descritto. La configurazione di default per le vostre direttive cache dovrebbe adattarsi alla maggior parte delle configurazioni.

`CacheRoot` configura il nome della directory che conterrà i file memorizzati nella cache. La direttiva `CacheRoot` di default è `/var/cache/httpd`.

`CacheSize` specifica quanti KB di spazio può utilizzare la cache. Il limite predefinito è 5 KB.

`CacheGcInterval` indica un limite di ore oltre il quale i file nella cache vengono cancellati qualora la cache superasse la quantità massima di spazio consentita (secondo quanto stabilito nella configurazione di `CacheSize`). Il valore di default per `CacheGcInterval` è 4 ore.

In assenza di una nuova richiesta del server, i documenti HTML rimangono nella cache per un numero massimo di ore configurato in `CacheMaxExpire`. Il valore di default è 24 ore.

`CacheLastModifiedFactor` si occupa della creazione di una data di scadenza per un documento che non ne possiede una propria. Il valore di default per `CacheLastModifiedFactor` è 0.1. Questo significa che la data di scadenza per questi documenti è pari a 1/10 della quantità di tempo trascorso dall'ultima modifica.

`CacheDefaultExpire` specifica (in ore) la scadenza di un documento ricevuto tramite un protocollo che non supporta la data di scadenza. La configurazione di default è 1 ora.

I documenti ricevuti da un host o dominio conforme alla configurazione in `NoCache` non vengono memorizzati nella cache. Se conoscete host o domini dai quali non volete memorizzare documenti, eliminate il commento dalla direttiva `NoCache` e inserite il nome di tali host o domini.

14.5.69. NameVirtualHost

È necessario utilizzare la direttiva `NameVirtualHost` per l'indirizzo IP (e, se necessario, il numero di porta) di ogni host virtuale che state configurando. La configurazione degli host virtuali basati sul nome viene utilizzata quando volete configurare diversi host virtuali per altrettanti domini, ma non potete (o non volete) usare indirizzi IP diversi per ogni singolo nome di dominio.



Nota Bene

Ogni host virtuale basato sul nome funzionerà solo con connessioni HTTP non sicure, dato che non potete utilizzare host virtuali basati sui nomi con un server sicuro. In questo secondo caso, dovrete utilizzare host virtuali basati su indirizzi IP.

Se usate un host virtuale basato sul nome, eliminate il commento dalla direttiva `NameVirtualHost` e aggiungete l'indirizzo IP corretto. Poi inserite le informazioni relative ai differenti domini utilizzando i tag `VirtualHost` vicino al `ServerName` per ogni host virtuale e per qualsiasi altra direttiva di configurazione applicabile solo a quell'host virtuale.

14.5.70. VirtualHost

I tag `<VirtualHost>` e `</VirtualHost>` si trovano accanto alle direttive di configurazione da applicare a un host virtuale. La maggior parte delle direttive di configurazione può essere utilizzata all'interno dei tag dell'host virtuale e viene poi applicata solo a quel particolare host virtuale.

In alcune direttive di configurazione e in alcuni segnaposto si trovano una serie di tag `VirtualHost` commentati. Per maggiori informazioni sugli host virtuali, consultate la Sezione 14.8.



Nota Bene

Tutto il contesto degli host virtuali SSL è stato spostato nel file `/etc/httpd/conf.d/ssl.conf`.

14.5.71. SetEnvIf

La direttiva di configurazione `SetEnvIf` può essere utilizzata per impostare variabili di ambiente basate sulle intestazioni nella richiesta. Nel file `httpd.conf` è usata per disabilitare i keepalive HTTP e per consentire al protocollo SSL di chiudere la connessione senza un messaggio di avvertimento da parte del browser client. Questa impostazione è necessaria per alcuni browser che non chiudono in modo affidabile la connessione SSL.

14.5.72. Direttive di configurazione per SSL

Le direttive SSL incluse nel file `/etc/httpd/conf.d/ssl.conf` possono essere configurate per abilitare comunicazioni Web sicure utilizzando SSL e TLS.

Per maggiori informazioni sulle direttive SSL visitate il sito http://localhost/manual/mod/mod_ssl/ oppure la documentazione relativa ad Apache `mod_ssl` all'indirizzo http://httpd.apache.org/docs-2.0/mod/mod_ssl.html.

Per informazioni sulla configurazione di un server sicuro HTTP Apache, consultate il capitolo relativo alla *configurazione di un server sicuro HTTP Apache* nella *Official Red Hat Linux Customization Guide*.



Nota Bene

Non modificate le vostre direttive SSL se non siete più che certi di quello che state facendo. Nella maggior parte dei casi le direttive di default sono quelle più indicate.

14.6. Moduli predefiniti

In Apache HTTP Server sono disponibili alcuni moduli. Per default i moduli indicati di seguito sono installati e abilitati con il pacchetto `httpd` in Red Hat Linux:

```
mod_access
mod_auth
mod_auth_anon
mod_auth_dbm
mod_auth_digest
mod_include
mod_log_config
mod_env
mod_mime_magic
mod_cern_meta
mod_expires
mod_headers
mod_usertrack
mod_unique_id
mod_setenvif
mod_mime
mod_dav
mod_status
mod_autoindex
mod_asis
mod_info
mod_cgi
mod_dav_fs
mod_vhost_alias
mod_negotiation
mod_dir
mod_imap
mod_actions
mod_speling
mod_userdir
mod_alias
mod_rewrite
```

Sono inoltre disponibili i moduli seguenti installando pacchetti aggiuntivi:

```
mod_auth_mysql
mod_auth_pgsq1
mod_perl
mod_python
mod_ssl
php
```

```
squirrelmail
```

14.7. Aggiungere moduli al server

Poiché Apache HTTP Server 2.0 supporta i DSO, è possibile caricare i moduli di Apache o compilare i vostri moduli. Il supporto DSO indica inoltre che i moduli vengono caricati solo se necessario, senza sprecare risorse di memoria.

Il team di sviluppo di Apache fornisce una documentazione completa all'indirizzo <http://httpd.apache.org/docs-2.0/dso.html>. Dopo aver installato il pacchetto `httpd-manual`, potete anche reperire la documentazione sui moduli di Apache all'indirizzo <http://localhost/manual/mod/>.

Affinché Apache HTTP Server utilizzi dinamicamente un modulo condiviso, tale modulo deve disporre di una linea `LoadModule` nel file `httpd.conf`.

Una linea di esempio `LoadModule` è simile a:

```
LoadModule access_module modules/mod_access.so
```

Se aggiungete o eliminate moduli dal file `httpd.conf`, dovete ricaricare o riavviare Apache, come descritto in la Sezione 14.4.

Se disponete di un modulo personale, potete aggiungerlo al file `httpd.conf` per fare in modo che venga compilato e caricato come DSO. Per questo motivo deve essere installato il pacchetto `httpd-devel`, perché contiene i file `include`, i file dell'intestazione e l'applicazione *APache eXtension* (`apxs`). Il comando `apxs` utilizza i file `include` e quelli di intestazione per compilare i moduli Apache.

Se avete scritto da voi il modulo o state utilizzando un modulo di terza parte, dovrete essere in grado di usare `apxs` per compilare i file sorgenti esterni all'albero delle directory di Apache. Se avete bisogno di maggiori informazioni su `apxs`, consultate la documentazione di Apache all'indirizzo <http://httpd.apache.org/docs-2.0/dso.html> e la pagina `man` di `apxs`.

Dopo aver compilato il vostro modulo, inseritelo nella directory `/usr/lib/httpd`. Inserite poi una linea `LoadModule` nel file `httpd.conf`. In fondo all'elenco `LoadModule` in `httpd.conf`, aggiungete una linea per l'oggetto condiviso, simile alla seguente:

```
LoadModule <foo_module> moduli/<mod_foo.so>
```

È necessario cambiare il nome del modulo e il nome di `<foo_module>` e di `<mod_foo.so>` in modo appropriato.

Concluse le fasi precedenti, riavviate il vostro server Web come indicato nella Sezione 14.4. Se avete eseguito tutto correttamente, il server Web dovrebbe trovare il modulo e caricarlo all'avvio.

14.8. Utilizzare gli host virtuali

Apache HTTP Server ha la possibilità di usare gli host virtuali in modo da permettere a server diversi di funzionare con indirizzi IP differenti, nomi di host o porte differenti sulla stesso server. Se siete interessati a utilizzare gli host virtuali, potete trovare la documentazione sul vostro computer, oppure online all'indirizzo <http://httpd.apache.org/docs-2.0/vhosts/>.



Nota Bene

Non potete usare gli host virtuali con il vostro Red Hat Linux Advanced Server, poiché la handshake SSL si attiva prima della richiesta HTTP che identifica l'host virtuale corretto. Se volete usare gli host virtuali basati sul nome, tenete presente che funzioneranno solo con il server Web non sicuro.

Gli host virtuali sono configurati nel file `httpd.conf` come descritto in la Sezione 14.5. Prima di modificare la configurazione degli host virtuali, leggete la relativa documentazione.

14.8.1. Host virtuali del server Web sicuro

La configurazione di default del vostro server Web esegue un server sicuro e uno non sicuro. Entrambi i server utilizzano lo stesso indirizzo IP e lo stesso nome host, ma attendono su porte differenti e il server sicuro è un host virtuale. Questa configurazione vi permette di abilitare documenti da un server sicuro e da uno non sicuro nel modo più efficiente possibile. Le trasmissioni HTTP di tipo sicuro impiegano più tempo, poiché il traffico è più intenso e, di conseguenza, si possono servire molte meno pagine al secondo. Occorre tenerne conto al momento di decidere quali informazioni includere sul server sicuro e quali far gestire al server non sicuro.

Le direttive di configurazione per il server sicuro sono contenute nei tag degli host virtuali nel file `/etc/httpd/conf.d/ssl.conf`. Se dovete modificare la configurazione del server dovete cambiare le direttive di configurazione contenute nei tag dell'host virtuale.

Per default, i server Web sicuri e non sicuri condividono la stessa `DocumentRoot`. Per cambiare la `DocumentRoot` in modo tale che non venga condivisa dai server sicuri e non sicuri, cambiate una delle direttive `DocumentRoot`. La `DocumentRoot` all'interno o all'esterno dei tag degli host virtuali in `httpd.conf` definisce la `DocumentRoot` per il vostro server Web non sicuro. Quella all'interno dei tag degli host virtuali in `conf.d/ssl.conf` definisce il documento root per il server sicuro.

Il server Apache HTTP Server sicuro è in attesa sulla porta 443 mentre il vostro server non sicuro rimane in attesa sulla porta 80. Affinché il server Web non sicuro non accetti le connessioni, cercate la seguente linea:

Quindi, commentate tutte le linee di `httpd.conf` che contengono `Listen 80`.

14.8.2. Configurazione degli host virtuali

Per creare un host virtuale, dovete modificare le relative linee, fornite come esempio, nel file `httpd.conf` oppure creare una sezione nuova.

Le linee d'esempio per l'host virtuale sono simili alle seguenti:

```
#<VirtualHost *>
#   ServerAdmin webmaster@dummy-host.example.com
#   DocumentRoot /www/docs/dummy-host.example.com
#   ServerName dummy-host.example.com
#   ErrorLog logs/dummy-host.example.com-error_log
#   CustomLog logs/dummy-host.example.com-access_log common
#</VirtualHost>
```

Eliminate il commento da tutte le linee e aggiungete poi le informazioni corrette per l'host virtuale.

Nella prima linea, modificate `*` con i dati relativi al vostro indirizzo IP. Cambiate la direttiva `ServerName` con un nome del DNS *valido* da utilizzare per l'host virtuale.

È inoltre necessario eliminare il commento a una delle seguenti linee `NameVirtualHost`:

```
NameVirtualHost *
```

Successivamente, modificate l'indirizzo IP e, se necessario, il numero di porta per l'host virtuale. Al termine otterrete un risultato simile al seguente:

```
NameVirtualHost 192.168.1.1:80
```

Se state configurando un host virtuale e volete che rimanga in attesa su una porta che non sia quella di default, dovete configurare un host virtuale per quella porta e aggiungere una direttiva `Listen` corrispondente.

In seguito, aggiungete il numero di porta nella prima linea della configurazione dell'host virtuale come riportato nell'esempio seguente:

```
<VirtualHost ip_address_of_your_server:12331>
```

Questa linea crea un host virtuale in attesa sulla porta 12331.

Dovete riavviare `httpd` per avviare un nuovo host virtuale. Per maggiori informazioni su come avviare e chiudere il file `httpd`, consultate la Sezione 14.4.

Informazioni più dettagliate sulla creazione e configurazione degli host virtuali basati sul nome e sull'indirizzo sono contenute nella pagina Web <http://httpd.apache.org/docs-2.0/vhosts/>.

14.9. Risorse aggiuntive

Per saperne di più su Apache HTTP Server, consultate le fonti seguenti.

14.9.1. Siti Web utili

- <http://httpd.apache.org> — il sito Web ufficiale per il server Web Apache, con tutta la documentazione relativa alle direttive e ai moduli caricati per default.
- <http://www.modssl.org> — il sito Web ufficiale di `mod_ssl`.
- <http://www.apacheweek.com> — pubblicazione online settimanale su tutto ciò che riguarda Apache.

14.9.2. Libri correlati

- *Apache Desktop Reference* di Ralf S. Engelschall; Addison Wesley

Scritto da Ralf EngelSchall, membro dell'ASF e autore di `mod_ssl`, *Apache Desktop Reference* rappresenta una guida di riferimento concisa ma esauriente su come usare Apache durante le fasi di compilazione, configurazione ed esecuzione. È possibile scaricare la versione online di questo libro all'indirizzo <http://www.apacheref.com/>.

- *Professional Apache* di Peter Wainwright; Wrox Press Ltd

Professional Apache è uno dei numerosi libri della collana Wrox Press Ltd's "Programmer to Programmer" e si rivolge agli utenti esperti e agli amministratori dei server Web che si apprestano a utilizzare Apache per la prima volta.

- *Administering Apache* di Mark Allan Arnold; Osborne Media Group

Questo libro si rivolge a quei provider di servizi Internet che desiderano fornire servizi più sicuri.

- *Apache Server Unleashed* di Richard Bowen, et al; SAMS BOOKS

Una risorsa enciclopedica su Apache.

- *Apache Pocket Reference* di Andrew Ford, Gigi Estabrook; O'Reilly

È l'ultimo aggiornamento della collana O'Reilly Pocket.

La posta elettronica è uno dei servizi più usati su Internet. Che siate utenti desktop o amministratori di sistema, Red Hat Linux vi offre molti modi per l'utilizzo e l'accesso dei messaggi.

Questo capitolo tratta i protocolli e-mail attualmente più diffusi e alcuni programmi ideati per le operazioni di posta elettronica.

15.1. Protocolli

La posta elettronica, come altri servizi di rete, utilizza vari protocolli che consentono a computer diversi, spesso con sistemi operativi e programmi e-mail differenti, di comunicare fra loro tramite messaggi di posta elettronica.

I seguenti protocolli sono fra i più comunemente utilizzati per il trasferimento di e-mail da un sistema all'altro.

15.1.1. IMAP

L'*Internet Message Access Protocol (IMAP)* è un metodo usato dalle applicazioni client di e-mail per accedere ai messaggi archiviati in remoto. Quando si utilizza l'IMAP i messaggi di posta elettronica rimangono nell'e-mail server remoto, dove l'utente può leggerli o cancellarli, e creare, rinominare o eliminare caselle di posta per archiviare le e-mail.

Inoltre, l'IMAP è totalmente compatibile con importanti standard di Internet messaging, fra cui MIME (Multipurpose Internet Mail Extensions), per consentire la ricezione di allegati. Molti client di e-mail che utilizzano l'IMAP possono essere anche configurati per il caching locale di una copia dei messaggi: in questo modo potrete consultare messaggi letti in precedenza anche se non siete direttamente connessi al server IMAP.

L'IMAP è utilizzato principalmente da utenti che possono accedere alle e-mail mediante più computer. Gli utenti connessi a Internet o a una rete privata mediante connessione a banda bassa utilizzano spesso questo protocollo, poiché inizialmente visualizza le informazioni di testa del messaggio. Ciò consente di diffirere il download dei messaggi con allegati ampi finché la larghezza di banda limitata non è in uso. Allo stesso modo, l'utente può cancellare i messaggi e-mail indesiderati senza visualizzare il corpo del messaggio, evitando addirittura di scaricarlo durante la connessione.

I documenti *Request for Comment (RFC)* che coprono l'IMAP contengono diversi dettagli e specifiche sul funzionamento del protocollo. RFC-1730 definisce le modalità d'utilizzo di IMAP nella versione 4, mentre RFC-2060 tratta l'implementazione IMAP corrente usata da numerosi server IMAP, la versione IMAP4rev1.

Il pacchetto `imap` contenuto in Red Hat Linux consente agli utenti di connettersi al vostro sistema e di ricevere le e-mail personali usando IMAP. Le connessioni IMAP sicure sono supportate dalla tecnologia SSL (Secure Socket Layer) integrata nel demone `/usr/sbin/imapd` e consentono l'utilizzo del file certificato `/usr/share/ssl/certs/imapd.pem`. Il sistema di crittografia SSL per le connessioni IMAP non necessita del programma `stunnel`, sebbene sia supportato. Per ulteriori informazioni riguardo alle opzioni di crittografia, consultate la Sezione 15.6.2.

Sono disponibili anche altri client e server IMAP, commerciali e gratuiti, molti dei quali estendono il protocollo IMAP e dispongono di funzionalità aggiuntive. Potete trovare un elenco completo nel sito <http://www.imap.org/products/longlist.htm>.

15.1.2. POP

Il *Post Office Protocol (POP)* consente ai clienti di e-mail di scaricare i messaggi di posta elettronica dal server remoto e salvarli sul computer locale. La maggior parte dei client POP di e-mail è configurata per l'eliminazione automatica del messaggio sul mail server dopo che questo è stato trasferito con successo sul sistema del client, anche se questo solitamente viene modificato.

Per connettersi a un server POP, il client di e-mail apre una connessione TCP sulla porta 110 del server. Una volta effettuata la connessione, il server POP si presenta al client POP e i due cominciano a scambiarsi comandi e risposte specificati nel protocollo. Parte di questa comunicazione prevede l'autenticazione del client POP in quella che viene definita *Fase di autenticazione*, in cui il nome utente e la password dell'utente sono inviati al server POP. Effettuata l'autenticazione, il client POP passa alla successiva, *Fase di transazione*, in cui è possibile utilizzare comandi quali LIST, RETR e DELE rispettivamente per elencare, scaricare ed eliminare i messaggi dal server. I messaggi da eliminare non sono, di fatto, rimossi dal server finché il client POP non invia il comando QUIT per terminare la sessione. A questo punto, il server POP entra nella *Fase di aggiornamento*, dove elimina i messaggi per i quali era stata richiesta la cancellazione ed elimina qualsiasi risorsa residua dalla sessione.

Il POP è un protocollo molto più semplice rispetto all'IMAP, in quanto i comandi che possono essere inviati tra client e server sono più limitati. POP è il protocollo ideale per gli utenti che dispongono di un unico sistema per leggere le e-mail, dato che scaricano i messaggi. Questo protocollo mostra un buon funzionamento anche quando la connessione a Internet o alla rete contenente il mail server non è costante.

Vi sono diversi RFC per coprire il protocollo POP, ma RFC-1939 definisce le linee principali di POP3, la versione corrente.

Occasionalmente, potreste incorrere in varianti meno diffuse del protocollo POP:

- *APOP* — POP3 con autenticazione MDS, in cui il client e-mail invia al server la password criptata e non in chiaro.
- *KPOP* — POP3 con autenticazione Kerberos. Per ulteriori informazioni sull'autenticazione Kerberos, consultate il capitolo 10.
- *RPOP* — POP3 con autenticazione RPOP, utilizza un'ID per ogni utente, simile a una password, per autenticare le richieste del POP. L'ID non è tuttavia criptato, quindi RPOP non è più sicuro di un POP standard.

Red Hat Linux supporta molti server e client POP, oltre ad altre applicazioni. Se preferite un client di e-mail grafico, **Mozilla Mail** o **Ximian Evolution** costituiscono un'ottima scelta. La vostra posta elettronica può essere reperita via POP anche attraverso altre utility, come Fetchmail. Se state utilizzando il sistema Red Hat Linux come server mail, il pacchetto `imap` installa, nella directory `/usr/sbin`, i demoni POP2 (`ipop2`) e POP3 (`ipop3`).

15.1.3. SMTP

Se i protocolli IMAP e POP permettono all'utente di ricevere le e-mail, il *Simple Mail Transfer Protocol (SMTP)* serve per inviarle. La posta in uscita utilizza l'SMTP (Simple Mail Transfer Protocol) per spostarsi dal client al server e avvicinarsi alla destinazione finale. L'SMTP può essere usato anche da due server per trasportare posta e quindi comunicare.

Per la comunicazione, l'SMTP utilizza la porta 25 del server. Un primo scambio SMTP inizia quando il sistema di connessione invia un comando MAIL FROM: `<indirizzo-email>` per iniziare lo scambio. Il sistema ricevente risponde con un messaggio 250 di avvenuta ricezione del primo comando. Il sistema di connessione indica quindi gli indirizzi e-mail all'altro sistema per ricevere il messaggio, seguiti dal messaggio DATA. Questo informa il sistema ricevente che la parte successiva della comunicazione sarà il corpo del messaggio. Quando il sistema di connessione ha terminato questa fase, posiziona un punto singolo (.) su una riga e il messaggio si considera inviato.

L'SMTP gestisce anche i casi in cui le e-mail devono essere inoltrate da un sistema all'altro, quando il sistema ricevente conosce la destinazione del messaggio. Il protocollo è in grado di verificare se alcuni utenti utilizzano un particolare server mail (il comando `VERFY` command) oppure di espandere una mailing list (il comando `EXPN`). L'e-mail può anche essere scambiata tra due server SMTP, se entrambi i sistemi consentono questo tipo di attività.

A differenza dell'IMAP e del POP, la forma più elementare di SMTP non richiede alcuna autenticazione. Ciò significa che i server SMTP possono permettere a qualunque utente di Internet di utilizzare il vostro sistema per inviare o trasmettere la posta a un numero imprecisato di destinatari. Questo ha reso possibili numerosi spam. Le applicazioni SMTP moderne minimizzano ormai questo comportamento, riducendo la ritrasmissione e consentendo l'invio di posta solo agli host conosciuti.

RFC-821 evidenzia il comportamento di base dell'SMTP. Tuttavia, negli anni, molte estensioni SMTP possibili grazie a RFC-1869 hanno aggiunto ulteriori funzioni a questo protocollo, rendendo disponibili nuovi comandi. Inizializzando una conversazione con un server SMTP con il comando `EHLO` piuttosto che `HELO`, il server per la connessione identifica se stesso come un server che supporta le estensioni SMTP. Il server ricevente risponde con una riga 250 contenente le diverse estensioni SMTP supportate. Il server di connessione può quindi usare le estensioni supportate per la comunicazione.

Un'importante estensione riguarda l'inserimento dell'Autenticazione SMTP mediante il comando `AUTH` come evidenziato in RFC-2554. Un'altra estensione molto diffusa viene descritta nel dettaglio in RFC-2034, la quale tratta l'utilizzo di codici d'errore standardizzati e separati dal punto, necessari tra applicazioni SMTP. La lettura dei vari RFC riguardanti l'SMTP fornisce informazioni di base sui percorsi Internet dei messaggi di posta elettronica. È inoltre possibile connettersi a un server SMTP via telnet specificando la porta 25, per esempio `telnet localhost 25`. L'esecuzione di alcuni comandi e l'invio di posta manualmente rappresentano un buon metodo per acquisire familiarità con il funzionamento delle comunicazioni SMTP.

Red Hat Linux 8.0 utilizza Sendmail (`/usr/sbin/sendmail`) come programma SMTP predefinito. Tuttavia è disponibile anche un'applicazione più semplice denominata Postfix (`/usr/sbin/postfix`). Per ulteriori informazioni, consultate la Sezione 15.2.2.

15.2. Diversi tipi di programmi e-mail

In generale, esistono tre tipi di programmi e-mail, ciascuno dei quali svolge un ruolo specifico nel processo di trasferimento e gestione dei messaggi di posta elettronica. La maggior parte degli utenti conosce solo il programma e-mail usato specificatamente per ricevere e inviare i messaggi, ma perché il messaggio arrivi al corretto destinatario sono fondamentali tutti e tre.

15.2.1. Mail User Agent

Il *Mail User Agent (MUA)* è un programma che consente all'utente almeno di leggere e scrivere messaggi e-mail. Spesso ci si riferisce al MUA come a *client di e-mail*. Molti MUA permettono all'utente di svolgere altri compiti, fra cui il reperimento di messaggi attraverso i protocolli POP o IMAP, l'impostazione di mailbox per archiviare i messaggi e il passaggio delle nuove e-mail a un programma Mail Transfer Agent affinché questo le consegna al destinatario finale.

I programmi MUA possono essere grafici, come **Mozilla Mail**, oppure possono avere un'interfaccia semplice, basata sul testo, per esempio **Mutt**.

15.2.2. Mail Transfer Agent

Il *Mail Transfer Agent (MTA)* trasferisce i messaggi e-mail tra computer che utilizzano l'SMTP. Durante il trasferimento verso la destinazione finale, un messaggio può interessare diversi MTA. La

maggior parte degli utenti non è consapevole della loro presenza, anche se ogni e-mail è inviata grazie ad almeno un MTA.

La consegna dei messaggi tra computer sembra abbastanza diretta, ma in realtà l'intero processo necessario per decidere se un particolare MTA può o dovrebbe accettare un messaggio per la consegna, è piuttosto complicato. Inoltre, a causa di problemi dovuti allo spamming, l'uso di un MTA specifico è in genere limitato dalla sua stessa configurazione o dall'accesso alla rete del sistema che lo esegue.

Molti MUA più ampi e complessi possono essere usati anche per inviare e-mail, ma quest'azione non deve essere confusa con i compiti di un MTA vero e proprio. Affinché gli utenti che eseguono il proprio MTA possano inviare i messaggi in uscita verso un computer remoto per la loro consegna, è necessario usare la funzione del MUA che trasferisce il messaggio a un MTA autorizzato dagli utenti stessi. La consegna del messaggio al server di posta non avviene in modo diretto: questo compito è riservato all'MTA.

Red Hat Linux utilizza Sendmail come MTA predefinito, sebbene supporti anche tipi differenti.



Suggerimento

Per informazioni su come impostare Postfix invece di Sendmail per l'MTA predefinito, consultate il capitolo relativo alla *configurazione dell'MTA (Mail Transport Agent)* nella *Official Red Hat Linux Customization Guide*.

15.2.3. Mail Delivery Agent

Il *Mail Delivery Agent (MDA)* è utilizzato dall'MTA per consegnare le e-mail a una mailbox utente specifica. In molti casi, l'MDA è di fatto un *Local Delivery Agent (LDA)*, come `/bin/mail` o `procmail`. Tuttavia, Sendmail può svolgere anche le funzioni di un MDA, per esempio quando accetta un messaggio per un utente locale e lo inserisce nel file spool delle e-mail. Tutti i programmi che gestiscono un messaggio da consegnare fino al punto in cui può essere letto da un MUA possono essere considerati MDA. È importante ricordare che gli MDA non trasportano messaggi tra sistemi o interfacce con l'end user.

Molti utenti non utilizzano direttamente gli MDA in quanto per ricevere e inviare messaggi sono sufficienti MTA e MUA. In ogni caso, è possibile usare alcuni MDA per selezionare i messaggi prima che l'utente li legga, e ciò risulta estremamente utile quando il flusso di messaggi è elevato.

15.3. Sendmail

Red Hat Linux utilizza Sendmail come MTA predefinito, sia che questi siano destinati a utenti sullo stesso sistema o a destinazioni remote. Esistono anche altri MTA, ma la maggior parte degli amministratori sceglie di usare Sendmail come MTA per la potenza, la scalabilità e la conformità a importanti standard Internet, fra cui SMTP.

Il compito principale di Sendmail, come di altri MTA, è il trasferimento sicuro della posta tra host, in genere mediante il protocollo SMTP. Sendmail offre elevate possibilità di configurazione e vi consente di controllare praticamente ogni aspetto riguardante la gestione della posta elettronica, inclusi i protocolli da usare.

15.3.1. Storia

Le origini di Sendmail risalgono alla nascita della posta elettronica, una decina d'anni prima dell'apparizione di ARPANET, precursore di Internet. Allora ogni mailbox utente corrispondeva a un

file la cui lettura era riservata al solo titolare e le applicazioni mail aggiungevano solo il testo al file. Gli utenti dovevano passare al vaglio il file mail per trovare la vecchia posta mentre la lettura di quella nuova risultava difficile. Il primo trasferimento effettivo di un file di messaggio e-mail da un host all'altro avvenne nel 1972, quando per la posta elettronica si iniziò a usare l'FTP sul protocollo di rete NCP. Questo metodo di comunicazione semplificato si diffuse molto rapidamente, fino al punto da assorbire la maggior parte del traffico ARPANET in meno di un anno.

Tuttavia, la mancanza di standardizzazione tra i protocolli concorrenti ha complicato notevolmente l'invio dei messaggi a opera di alcuni sistemi, e la situazione rimase tale fino alla standardizzazione di ARPANET sul TCP/IP nel 1982. Apparve poi un nuovo protocollo, l'SMTP, per il trasporto dei messaggi. Questi sviluppi, insieme alla sostituzione dei file HOSTS con il DNS, hanno consentito la materializzazione degli MTA e delle loro caratteristiche complete. Con l'espansione di Internet, Sendmail, evoluzione di un precedente sistema di consegna di posta elettronica (Delivermail), è diventato ben presto lo standard.

15.3.2. Scopi e limiti

Conoscere Sendmail e i suoi possibili usi diventa molto importante per gli utenti. Al giorno d'oggi, per via della preponderanza di applicazioni monolitiche che svolgono compiti molteplici, l'utente potrebbe erroneamente pensare a Sendmail come l'unica applicazione necessaria per un server mail all'interno della propria organizzazione. Tecnicamente, ciò corrisponde a verità, poiché Sendmail indirizza la posta verso le directory dei diversi utenti e accetta nuovi messaggi mediante la linea di comando. La maggior parte degli utenti richiede, tuttavia, qualcosa di più che la semplice consegna della posta. Vuole interagire con il servizio di posta elettronica usando un MUA che utilizza POP o IMAP per scaricare i messaggi sul computer locale o preferire un interfaccia Web per accedere alla propria mailbox. Queste applicazioni aggiuntive funzionano insieme a Sendmail e SMTP, ma il loro scopo è differente e possono operare separatamente le une dalle altre.

Questa sezione non prevede l'approfondimento di Sendmail, di come potrebbe o dovrebbe essere configurato. Vi sono centinaia di opzioni e regole differenti e interi volumi che illustrano le potenzialità di Sendmail e le soluzioni ai problemi. Si consiglia di consultare le fonti informative presenti online e offline per impostarlo secondo le vostre specifiche.

È comunque importante capire quali file vengono installati per default con Sendmail sul vostro sistema, come effettuare modifiche di base alla sua configurazione o come bloccare la posta indesiderata (spam) e ampliare Sendmail mediante il protocollo *Lightweight Directory Access Protocol (LDAP)*.

15.3.3. Installazione predefinita di Sendmail

Sebbene sia possibile scaricare il codice sorgente per Sendmail e sviluppare la propria copia, molti utenti preferiscono utilizzare la versione di Sendmail installata di default con il sistema Red Hat Linux. Successivamente, è possibile utilizzare i CD-ROM di Red Hat Linux per reinstallare l'RPM `sendmail`. Ricordatevi di modificare il file di configurazione predefinito, in modo che Sendmail possa utilizzarlo come server di posta per più host locali. Per maggiori dettagli, consultate la Sezione 15.3.4.

Dopo l'installazione, l'eseguibile `sendmail` viene posizionato nella directory `/usr/sbin`.

Il lungo e dettagliato file di configurazione di Sendmail (`sendmail.cf`) viene installato in `/etc/mail`. Evitate di modificare direttamente il file `sendmail.cf`. Invece, per apportare delle modifiche a Sendmail è preferibile modificare il file `/etc/mail/sendmail.mc` eseguire un backup del file originale `/etc/mail/sendmail.cf` e usare il macro processore `m4` per creare un nuovo `/etc/sendmail.cf`. Ulteriori informazioni sulla configurazione di Sendmail sono disponibili in la Sezione 15.3.4.

Diversi file di configurazione di Sendmail sono installati in `/etc/mail`,

- `access` — specifica quali sistemi possono utilizzare Sendmail per la posta elettronica.
- `domaintable` — consente di fornire la mappatura del nome di dominio.
- `local-host-names` — il punto in cui si indicano tutti gli alias per il computer.
- `mailertable` — specifica le istruzioni per superare il routing per particolari domini.
- `virtusertable` — consente di eseguire una forma di aliasing specifica per dominio e quindi posizionare domini virtuali multipli su un computer.

Diversi file di configurazione in `/etc/mail`, fra cui `access`, `domaintable`, `mailertable` e `virtusertable`, devono archiviare le informazioni nei file del database prima che Sendmail possa adottare qualsiasi modifica della configurazione. Per includere tali modifiche nei file del database, è necessario eseguire il comando `makemap hash /etc/mail/<nome> < /etc/mail/<nome>`, dove `<nome>` è il nome del file di configurazione da convertire.

Se, per esempio, volete indirizzare tutte le e-mail a un account `domain.com` perché siano consegnate a `<bob@otherdomain.com>`, sarà necessario aggiungere una riga simile a quella riportata di seguito al file `virtusertable`:

```
@domain.com      bob@otherdomain.com
```

Quindi, per aggiungere queste nuove informazioni al file `virtusertable.db`, eseguite `makemap hash /etc/mail/virtusertable < /etc/mail/virtusertable come root`. In questo modo creerete un nuovo `virtusertable.db` contenente la nuova configurazione.

15.3.4. Modifiche comuni alla configurazione di Sendmail

Un file `sendmail.cf` verrà installato di default in `/etc/mail/` durante il processo di installazione di Red Hat Linux. Tuttavia, per poter utilizzare alcune delle caratteristiche più avanzate del programma dovete modificare tale file.

Quando modificate il file di configurazione di Sendmail, è meglio creare un file `/etc/mail/sendmail.cf` completamente nuovo, piuttosto che modificarne uno già esistente.



Attenzione

Prima di effettuare modifiche al file `sendmail.cf`, è consigliabile fare un backup della versione predefinita.

Per aggiungere le funzionalità Sendmail desiderata, modificate il file `/etc/mail/sendmail.mc`. Fatto ciò, utilizzate il macro processore `m4` per generare un nuovo file `sendmail.cf`, eseguendo il comando `m4/etc/mail/sendmail.mc > /etc/mail/sendmail.cf`. Dopo la creazione di un nuovo `/etc/mail/sendmail.cf`, riavviate Sendmail affinché le modifiche siano effettive. Il modo più semplice è digitare il comando `service sendmail restart` come root.

Per default, il macro processore `m4` viene installato con Sendmail ed è incluso nel pacchetto `sendmail-cf`.



Importante

La versione di default di `sendmail.cf` non consente a sendmail di accettare connessioni di rete da host diversi dal computer locale. Se desiderate configurare sendmail come server per altri client, dovete modificare `/etc/mail/sendmail.mc` e impostare `DAEMON_OPTIONS` in modo che stia in ascolto sui dispositivi di rete oppure commentate l'intera opzione. Successivamente, ricreate `/etc/mail/sendmail.cf` eseguendo:

Questa configurazione dovrebbe funzionare per la maggior parte dei siti SMTP, ma *non* risulterà valida per i siti UUCP (UNIX to UNIX Copy). Dovrete, dunque, creare un nuovo `sendmail.cf` se dovete utilizzare trasferimenti di posta con UUCP.

Si consiglia di consultare il file `/usr/share/sendmail-cf/README` prima di modificare i file contenuti nelle sottodirectory di `/usr/share/sendmail-cf`, poiché possono influire sulle future configurazioni dei file `/etc/mail/sendmail.cf`.

15.3.4.1. Masquerading

Una comune configurazione di Sendmail vede un singolo computer fungere da mail gateway per tutti i computer della rete. Per esempio, una società potrebbe chiamare `mail.bigcorp.com` il computer che si occupa della gestione della posta elettronica e dell'assegnazione di un indirizzo del mittente coerente per tutta la posta in uscita.

In questa situazione il server di sendmail deve mascherare il nome del computer sulla rete dell'azienda per fare in modo l'indirizzo sia `user@bigcorp.com` invece di `user@devel.bigcorp.com`.

Per effettuare questa operazione, aggiungete le righe riportate di seguito al file `/etc/mail/sendmail.mc`.

```
FEATURE(always_add_domain)dnl
FEATURE('masquerade_entire_domain')
FEATURE('masquerade_envelope')
FEATURE('allmasquerade')
MASQUERADE_AS('bigcorp.com.')
MASQUERADE_DOMAIN('bigcorp.com.')
MASQUERADE_AS(bigcorp.com)
```

Dopo avere generato un nuovo file `sendmail.cf` mediante `m4`, la configurazione consentirà a tutta la posta della rete di essere visualizzata come se fosse inviata da `bigcorp.com`.

15.3.5. Misure anti-spam con Sendmail

Lo *spam* delle e-mail può essere identificato come la definizione della posta non necessaria e indesiderata ricevuta da un utente che non ha mai richiesto tale comunicazione. Si tratta di un ampio abuso, spiacevole e costoso, degli standard comunicativi di Internet.

Sendmail ha semplificato (relativamente) il blocco delle più recenti tecniche di spamming impiegate per inviare posta indesiderata attraverso il vostro sistema. Sendmail è addirittura predisposto per bloccare per default molti dei comuni metodi di spamming. Sarà necessario imparare ad attivare l'anti-spamming modificando il file `/etc/mail/sendmail.mc` in modo da abilitare il sistema.

Per esempio, l'inoltro dei messaggi SMTP, definito anche *SMTP relaying*, è stato disabilitato di default a partire dalla versione 8.9 di Sendmail. Prima di ciò, Sendmail istruiva il mail host (`x.org`) ad accettare i messaggi provenienti da una parte (`y.com`) e a inviarli a un'altra parte (`z.net`). Oggi, invece, è necessario specificare a Sendmail di permettere a un dominio di rilevare la posta attraverso il vostro dominio. È sufficiente modificare il file `/etc/mail/relay-domains` e riavviare Sendmail digitando il comando `service sendmail restart` come root per attivare le modifiche.

Accade spesso, tuttavia, che i vostri utenti siano bombardati da spam provenienti da altri server su Internet, che sfuggono al vostro controllo. In questi casi, utilizzate le caratteristiche di controllo agli accessi di Sendmail disponibili mediante il file `/etc/mail/access`. Come root, aggiungete i domini che desiderate bloccare o cui si vuole specificatamente consentire l'accesso come nell'esempio riportato di seguito:

```
badspammer.com      550 Go away and do not spam us anymore
tux.badspammer.com  OK
10.0                RELAY
```

Poiché `/etc/mail/access` è un database, utilizzate `makemap` per ricreare la mappa del database e attivare le modifiche. Per fare ciò, eseguite il comando `makemap hash /etc/mail/access </etc/mail/access come root.`

Questo esempio mostra che qualsiasi e-mail inviata da `badspammer.com` sarà bloccata con un codice d'errore 550 conforme alla specifica RFC-821 e rimandata allo spammer, a eccezione della posta inviata dal sottodominio `tux.badspammer.com`, la quale verrebbe accettata. L'ultima riga indica che qualsiasi e-mail inviata dalla rete `10.0.*.*` può essere rilevata dal mail server.

È facilmente intuibile che quest'esempio si limita alla superficie delle potenzialità di Sendmail in termini di consenso o blocco dell'accesso. Consultate il file `/usr/share/doc/sendmail/README.cf` per informazioni dettagliate e alcuni esempi.

15.3.6. Utilizzo di Sendmail con LDAP

Il protocollo *Lightweight Directory Access Protocol* (LDAP) fornisce un metodo veloce e potente per trovare informazioni su un utente specifico proveniente da un gruppo più ampio. È possibile, per esempio, utilizzare un server LDAP per cercare un indirizzo e-mail specifico in una directory comune aziendale partendo dal cognome dell'utente. In questo tipo d'implementazione il protocollo LDAP, ampiamente separato da Sendmail, archivia le informazioni gerarchiche sull'utente mentre Sendmail riceve solo il risultato delle query LDAP in messaggi e-mail pre-indirizzati.

Sendmail supporta un'integrazione maggiore con LDAP, in cui utilizza il protocollo per sostituire file mantenuti separatamente, fra cui `aliases` e `virtusertables`, su mail server differenti che funzionano insieme per supportare un'impresa che opera a livello medio-grande. In breve, il protocollo LDAP può essere utilizzato per analizzare il livello di routing della posta di Sendmail e i file di configurazione separati in un potente cluster LDAP, potenziato da differenti applicazioni.

La versione attuale di Sendmail include il supporto per LDAP. Per ampliare il server Sendmail usando LDAP, procurarsi prima un server LDAP, come **OpenLDAP**, funzionante e correttamente configurato. Modificate quindi `/etc/mail/sendmail.mc` per includere:

```
LDAPROUTE_DOMAIN('vostro_dominio.com')dnl
FEATURE('ldap_routing')dnl
```



Nota Bene

La presente è solo una configurazione elementare di Sendmail con LDAP. La vostra configurazione sarà sicuramente differente, poiché dipenderà dal tipo d'implementazione del protocollo LDAP, soprattutto se desiderate configurare diversi computer Sendmail per un server LDAP comune.

Consultate `/usr/share/doc/sendmail/README.cf` per istruzioni dettagliate riguardo alla configurazione di routing LDAP ed esempi.

Successivamente, ricreate il file `/etc/mail/sendmail.cf` eseguendo il comando `m4` e riavviando Sendmail. Per informazioni, consultate la Sezione 15.3.4.

Per ulteriori informazioni sul protocollo LDAP, consultate il Capitolo 18.

15.4. Fetchmail

Fetchmail è un programma per rilevare la posta elettronica da server remoti per connessioni TCP/IP su richiesta. Molti utenti apprezzano la capacità di separare il processo di download dei messaggi sul server remoto da quello di lettura e organizzazione dei messaggi in un MUA. Ideato specificatamente per gli utenti dial-up, Fetchmail è in grado di connettersi e scaricare velocemente tutti i messaggi e-mail nel file spool di posta, utilizzando un numero indefinito di protocolli, inclusi POP3 e IMAP. Questo programma, può anche inoltrare i vostri messaggi a un server SMTP, qualora fosse necessario.

Prima di usare Fetchmail, assicuratevi che sia stato installato sul sistema. In caso contrario, procedete all'installazione mediante l'RPM `fetchmail` sui CD-ROM Red Hat Linux.

Fetchmail viene configurato per ciascun utente mediante il file `.fetchmailrc` nella directory home dell'utente. Per una configurazione di base del file `.fetchmailrc`, personalizzabile secondo le necessità, si consiglia di utilizzare `fetchmailconf`, un programma molto utile fornito in insieme a Fetchmail.

Le preferenze contenute nel file `.fetchmailrc` consentono a Fetchmail di controllare la posta su un server remoto e di rinviarla nel tentativo di consegnarla alla porta 25 sul computer locale, utilizzando l'MTA locale per posizionare le e-mail nel corretto file spool utente. Se Procmail è disponibile, può essere usato per filtrare i messaggi e archivarli in una mailbox affinché siano letti da un MUA.

15.4.1. Opzioni di configurazione per Fetchmail

È possibile passare tutte le opzioni sulla linea di comando necessaria a controllare la posta su un server remoto durante l'esecuzione di Fetchmail, tuttavia si consiglia di semplificare l'operazione usando il file `.fetchmailrc`. Tutte le preferenze di configurazione vengono inserite nel file `.fetchmailrc`, che è comunque possibile escludere quando Fetchmail è in esecuzione, specificando l'opzione sulla riga di comando.

Il file `.fetchmailrc` si divide in tre tipi particolari di opzioni di configurazione:

- *opzioni globali* — fornisce a Fetchmail le istruzioni che controllano l'esecuzione del programma o le impostazioni per le connessioni di controllo dei messaggi.
- *opzioni server* — specifica le informazioni necessarie per il polling del server, fra cui il nome host o le preferenze che potreste voler applicare con un server mail specifico, per esempio la porta da controllare oppure i secondi d'attesa prima del time out. Queste opzioni riguardano ogni opzione utente utilizzata con quel server.
- *opzioni utente* — contiene informazioni, quali id utente e password, necessarie per l'autenticazione e il controllo della posta mediante un server mail particolare.

Le opzioni globali sono in cima al file `.fetchmailrc`, seguite da una o più opzioni server, ciascuna delle quali designa un server di posta differente che sarà controllato da Fetchmail. Le opzioni utente sono successive alle opzioni server e servono per gli account utente che si desidera controllare su un server di posta. Come le opzioni server, quelle multiple utente possono essere specifiche per un server particolare, e si usano, per esempio, per controllare account e-mail multipli sullo stesso server.

Le opzioni server nel file `.fetchmailrc` sono attivate da una speciale opzione verbale, `poll` oppure `skip`, che precede ogni informazione del server. L'azione di `poll` dice a Fetchmail di usare quest'opzione server quando è in esecuzione, in pratica si tratta del controllo della posta che utilizza le varie opzioni utente. Dopo un'azione di `skip`, invece, le opzioni server non sono verificate, a meno che non si specifichi il nome host del server quando si avvia Fetchmail. L'opzione `skip` consente di impostare configurazioni prova in `.fetchmailrc` e controlla il server solo se specificatamente indicato, senza interferire sulle configurazioni correnti.

Un file di esempio `.fetchmailrc` appare così:

```
set postmaster "user1"
```

```

set bouncemail

poll pop.domain.com proto pop3
    user 'user1' there with password 'secret' is user1 here

poll mail.domain2.com
    user 'user5' there with password 'secret2' is user1 here
    user 'user7' there with password 'secret3' is user1 here

```

In questo esempio le impostazioni globali sono impostate in modo tale che l'utente riceva la posta come ultima risorsa (opzione `postmaster`) e tutti gli errori di e-mail siano inviati al `postmaster` invece che al mittente (opzione `bouncemail`). L'azione `set` dice a Fetchmail che questa linea contiene un'opzione globale. Vengono quindi specificati due server di posta, uno impostato per rilevare la posta usando POP3 e l'altro per utilizzare diversi protocolli e consentire al primo di eseguire le proprie funzioni. Due utenti sono controllati mediante l'opzione relativa al secondo server, ma tutta la loro posta rilevata viene inviata al mail spool `user1`. Ciò consente il controllo delle mailbox multiple su server multipli, mentre compaiono in un'unica inbox MUA. Le informazioni specifiche per ogni utente sono attivate con l'azione `user`.



Nota Bene

Non è necessario inserire la password personale nel file `.fetchmailrc`. Potete pertanto tralasciare la sezione `with password '<password>'`. Fetchmail vi chiederà la password dopo l'avvio con il comando `fetchmail`.

Il file `.fetchmailrc` può essere impostato manualmente, ma sarà più semplice lasciare che il programma `fetchmailconf` incluso in Fetchmail esegua la configurazione per voi. Quando invece si verificano le configurazioni, è consigliabile modificare direttamente il file `.fetchmailrc`.

Come si conviene a un programma che fornisce un servizio di rete diffuso come la posta elettronica e che utilizza numerosi protocolli, Fetchmail contiene una varietà di opzioni globali, server e locali. Molte di queste opzioni sono usate di rado oppure si applicano solo a situazioni molto specifiche. La pagina `man fetchmail` spiega in dettaglio queste opzioni. Troverete qui di seguito un elenco delle più comuni.

15.4.1.1. Opzioni globali

Ogni opzione globale dovrebbe essere posizionata su una singola riga dopo un'azione di `set`.

- `daemon <secondi>` — indica a Fetchmail di usare in automatico la modalità demone, in cui resterà in esecuzione in background e rileverà la posta a intervalli specifici.
- `postmaster` — assegna a Fetchmail un utente locale per inviare la posta in caso di problemi di consegna.
- `syslog` — istruisce Fetchmail sull'attivazione dei messaggi di stato o di errore di accesso nel file log del sistema. Per default, il file è `/var/log/maillog`.

15.4.1.2. Opzioni server

Posizionate le opzioni server sulla riga specifica in `.fetchmailrc` dopo un'azione di `poll` oppure di `skip`.

- `auth <tipo-aut>` — specifica il tipo di autenticazione da utilizzare. Per default, viene utilizzata l'autenticazione `password`, ma alcuni protocolli supportano altri tipi di autenticazione, fra cui `kerberos_v5`, `kerberos_v4`, e `ssh`. Quando viene usata l'autenticazione di tipo `any`, Fetchmail tenterà prima i metodi che non richiedono la password, quindi quelli che usano il masquerading e infine cercherà di inviare la vostra password in chiaro per l'autenticazione presso il server.
- `interval <numero>` — indica a Fetchmail di interrogare il server ogni `<numero>` di volte che controlla la posta sui server configurati. Questa opzione può essere usata con i server di posta che ricevono i messaggi solo occasionalmente.
- `port <numero-porta>` — esclude il numero della porta di default per un protocollo specifico.
- `proto <protocollo>` — istruisce Fetchmail su un protocollo specifico da usare, fra cui `pop3` oppure `imap`, per controllare i messaggi su questo server.
- `timeout <secondi>` — configura Fetchmail per interrompere l'operazione dopo un certo intervallo di inattività del server. Se questo valore non viene impostato, si presuppone il valore predefinito di 300.

15.4.1.3. Opzioni utente

Le opzioni utenti possono essere posizionate sulle loro linee specifiche sotto un'opzione `server` o sulla stessa riga dell'opzione `server`. In entrambi i casi le opzioni utente seguono l'opzione `user` (definita qui sotto).

- `fetchall` — ordina a Fetchmail di scaricare tutti i messaggi in coda, compresi quelli che sono già stati visualizzati. Per default, Fetchmail scarica solo i messaggi nuovi.
- `fetchlimit <numero>` — consente solo di scaricare un certo numero di messaggi prima di terminare.
- `flush` — indica a Fetchmail di cancellare tutti i messaggi precedentemente visualizzati nella coda prima di scaricare i nuovi messaggi.
- `limit <numero-max-byte>` — consente di specificare le dimensioni massime ammesse per scaricare i messaggi. Quest'opzione risulta utile in caso di connessioni di rete lente, quando l'intervallo di tempo necessario a scaricare il messaggio è eccessivo.
- `password '<password>'` — specifica la password da usare per quell'utente.
- `preconnect "<comando>"` — indica a Fetchmail di eseguire il comando specifico prima di rilevare i messaggi per quell'utente.
- `postconnect "<comando>"` — indica a Fetchmail di eseguire il comando specifico dopo il rilevamento dei messaggi per quell'utente.
- `ssl` — consente a Fetchmail di raccogliere il messaggio mediante connessione criptata SSL, se il server la supporta.
- `user "<nome-utente>"` — imposta il nome utente usato da Fetchmail per rilevare i messaggi. Quest'opzione dovrebbe figurare in elenco prima di qualsiasi altra opzione utente.

15.4.2. Opzioni di comando di Fetchmail

La maggior parte delle opzioni Fetchmail utilizzabili sulla riga di comando, quando si esegue il comando `fetchmail`, rispecchia le opzioni di configurazione `.fetchmailrc`. In questo modo, Fetchmail può essere utilizzato con o senza un file di configurazione. La maggior parte degli utenti non userà queste opzioni sulla riga di comando, poiché è più semplice lasciarle nel file `.fetchmailrc` affinché siano attivate quando Fetchmail è in esecuzione.

A volte, tuttavia, si potrebbe desiderare di eseguire il comando `fetchmail` con altre opzioni per scopi particolari. Poiché ogni opzione specificata sulla linea di comando elimina le opzioni del file di configurazione, è possibile attivare le opzioni di comando per escludere temporaneamente l'impostazione `.fetchmailrc` che sta provocando un errore.

15.4.2.1. Opzioni informative o di debugging

Certe opzioni usate dopo il comando `fetchmail` possono fornire informazioni importanti.

- `--configdump` — visualizza ogni possibile opzione basata su informazioni provenienti da `.fetchmailrc` e dai valori predefiniti di Fetchmail. Quando viene usata quest'opzione, non vi è la possibilità di rilevare la posta per alcun utente.
- `-s` — esegue Fetchmail in modalità "silent", impedendo la visualizzazione di qualsiasi messaggio che non sia di errore dopo il comando `fetchmail`.
- `-v` — esegue Fetchmail in modalità "verbose", visualizzando ogni comunicazione tra Fetchmail e i server remoti.
- `-V` — fa sì che Fetchmail visualizzi informazioni dettagliate sulla versione, elenca le opzioni globali e mostra le impostazioni da usare per ogni utente, compreso il protocollo e-mail e il metodo di autenticazione. Quando questa opzione è attiva, la posta non viene rilevata per alcun utente.

15.4.2.2. Opzioni speciali

Queste opzioni vengono usate occasionalmente per escludere i valori di default spesso inclusi nel file `.fetchmailrc`.

- `-a` — indica a Fetchmail di scaricare tutti i messaggi, sia nuovi che precedentemente visualizzati, presenti sul server remoto. Di default, Fetchmail scarica solo i messaggi nuovi.
- `-k` — fa sì che Fetchmail lasci i messaggi sul server remoto dopo averli scaricati. Quest'opzione esclude la cancellazione di default dei messaggi dopo il download.
- `-l <numero-max-byte>` — indica a Fetchmail di non scaricare i messaggi che superano determinate dimensioni lasciandoli sul server remoto.
- `--quit` — abbandona il procedimento demone di Fetchmail.

Ulteriori comandi e opzioni `.fetchmailrc` sono disponibili sulla pagina `man fetchmail`.

15.5. Procmail

Procmail consente di filtrare i messaggi di posta elettronica al momento del rilevamento da un server di posta remoto o di posizionarli nel vostro file spool su un server remoto o locale. Procmail è potente, ampiamente utilizzato e non intrusivo. Comunemente identificato come un *Local Delivery Agent (LDA)*, svolge un ruolo minimo nella consegna della posta elettronica perché sia letta da un MUA.

Per utilizzare Procmail, è prima necessario installarlo. Digitare il comando `rpm -q procmail` per verificare l'installazione del pacchetto `procmail`. Se per qualche ragione Procmail non fosse presente nel sistema, installatelo usando i CD di installazione Red Hat Linux.

Procmail può essere richiamato in molti modi differenti. Dato che la posta elettronica viene posizionata nel file spool, Procmail può essere configurato per avviare, filtrare la posta nelle posizioni configurate per l'uso con il MUA e lasciare l'applicazione. In alternativa il MUA potrebbe essere

configurato per richiamare Procmail ogni volta che si riceve un messaggio, così che i messaggi siano archiviati nelle corrette mailbox. In molti casi la presenza di un file `.procmailrc` nella home directory utente richiamerà Procmail, nel caso in cui Sendmail sia in uso.

Le azioni eseguite da Procmail con i messaggi e-mail dipendono da istruzioni incluse in particolari regole, che predispongono il confronto dei messaggi a opera del sistema. Se un messaggio soddisfa i requisiti necessari, verrà posizionato in un certo file, eliminato o elaborato.

Una volta avviato, Procmail legge il messaggio e-mail e separa il corpo del testo dall'informazione dell'intestazione. Successivamente, Procmail cerca, di default, il file `/etc/procmailrc` e i file `rc` nella directory `/etc/procmailrcs` per le regole e le variabili di sistema dell'ambiente Procmail. Procmail cerca quindi un file `.procmailrc` nella directory home per cercare le regole specifiche per quell'utente. Molti utenti creano anche file `rc` aggiuntivi personali cui fa riferimento il file `.procmailrc` e che sono velocemente abilitati o disabilitati al verificarsi di problemi relativi al filtraggio della posta.

Per default, non esiste alcun file `rc` nella directory `/etc` e non vi sono nemmeno file utente `.procmailrc`. Per iniziare a usare Procmail, sarà necessario creare un file `.procmailrc` con particolari variabili e regole di ambiente che spieghino ciò che si desidera fare con determinati messaggi.

Nella maggior parte dei casi, la scelta di configurare Procmail per filtrare la posta elettronica dipende dall'esistenza di un file utente `.procmailrc`. Per disabilitare Procmail, salvando però il lavoro eseguito sul file `.procmailrc`, spostatelo in un file dal nome simile usando il comando `mv ~/.procmailrc ~/.procmailrcSAVE`. Quando sarete pronti a verificare nuovamente Procmail, cambiate il nome del file in `.procmailrc`. Procmail sarà pronto per il funzionamento immediato.

15.5.1. Configurazione di Procmail

I file di configurazione di Procmail, molto probabilmente il file `.procmailrc` dell'utente, contengono molte variabili di ambiente importanti. Tali variabili istruiscono Procmail sullo smistamento dei messaggi, sul suo comportamento nei confronti dei messaggi che non soddisfano le regole e così via.

Queste variabili di ambiente compaiono in genere all'inizio del file `.procmailrc`, nel seguente formato:

```
<variabile-amb>=<valore>"
```

In questo esempio, `<variabile-amb>` è il nome della variabile e `<valore>` la sezione che la definisce.

La maggior parte degli utenti Procmail non utilizza molte variabili, anche se molte delle più importanti sono già definite come valore di default. Spesso vi troverete di fronte le variabili seguenti:

- **DEFAULT** — imposta la mailbox di default in cui saranno posizionati i messaggi che non soddisfano le regole.

Il valore di default **DEFAULT** è uguale a `$ORGMAIL`.

- **INCLUDERC** — specifica i file `rc` aggiuntivi che contengono ulteriori regole per i messaggi da controllare. Ciò consente di separare gli elenchi delle regole in due singoli file che svolgono funzioni differenti, per esempio il blocco degli spam o la gestione delle mailing list, le quali sono attivabili o disattivabili utilizzando i caratteri commento nel file utente `.procmailrc`.

Il seguente potrebbe essere un esempio di due righe in un file utente `.procmailrc`:

```
MAILDIR=$HOME/Msgs
INCLUDERC=$MAILDIR/lists.rc
INCLUDERC=$MAILDIR/spam.rc
```

Se l'utente desidera disabilitare il filtro Procmail sugli elenchi di posta ma lasciare attivo il controllo degli spam, può semplicemente inserire il carattere `#` nella riga **INCLUDERC**.

- **LOCKSLEEP** — imposta l'intervallo di tempo, in secondi, che intercorre tra un tentativo e l'altro di Procmail di utilizzare un particolare file di blocco. Il tempo predefinito è di otto secondi.
- **LOCKTIMEOUT** — imposta l'intervallo di tempo, in secondi, che deve trascorrere dopo l'ultima modifica di un file di blocco perché Procmail supponga che tale file sia vecchio e possa essere eliminato. Il tempo predefinito è di 1024 secondi.
- **LOGFILE** — posizione e file che contengono i messaggi informativi e di errore di Procmail.
- **MAILDIR** — imposta la directory correntemente in uso per Procmail. Se impostata, tutti gli altri percorsi di Procmail sono legati a questa directory.
- **ORGMAIL** — specifica la mailbox originale o qualsiasi altro luogo dove posizionare i messaggi qualora non fosse possibile usare la posizione di default o quella richiesta dalle regole.

Per default, è usato un valore di `/var/spool/mail/$LOGNAME`.

- **SUSPEND** — imposta l'intervallo temporale, in secondi, che Procmail utilizzerà come pausa se una risorsa necessaria, come per esempio lo spazio swap, non è disponibile.
- **SWITCHRC** — consente all'utente di specificare un file esterno contenente regole Procmail aggiuntive, molto simile all'opzione **INCLUDERC**, tranne per la verifica delle regole, che di fatto termina sul file della configurazione di riferimento, e poiché sono utilizzate solo le regole sul file **SWITCHRC** specificato.
- **VERBOSE** — consente a Procmail di registrare una maggiore quantità d'informazioni. Questa opzione si rivela utile per il debugging.

Altre importanti variabili di ambiente sono prelevate dalla shell, fra queste **LOGNAME**, che corrisponde al vostro nome di login, **HOME**, ovvero la posizione della vostra directory home e **SHELL**, la shell di default.

Spiegazioni esaurienti su tutte le variabili di ambiente e sui valori di default sono disponibili nella pagina `man procmailrc`.

15.5.2. Regole Procmail

Per molti nuovi utenti, la costruzione delle regole rappresenta la parte più complicata di usare Procmail. In un certo senso, questo è comprensibile, poiché le regole utilizzano *espressioni regolari* per far sì che i messaggi soddisfino i requisiti richiesti. Queste espressioni regolari identificano un particolare formato utilizzato per specificare gli attributi delle stringhe corrispondenti. Il costruito delle espressioni regolari non è, tuttavia, complicato e la loro lettura ne rende ancora più semplice la comprensione. Inoltre, la consistenza del modo in cui le regole sono scritte, indipendentemente dalle espressioni regolari, facilita la comprensione del processo in corso.

Una spiegazione approfondita delle espressioni regolari esula dagli obiettivi di questo capitolo e pertanto non verrà trattata. La struttura delle regole di Procmail è molto più importante. Su internet sono disponibili utili esempi di regole Procmail. Fra i vari siti si consiglia <http://www.iki.fi/era/procmail/links.html>. L'uso appropriato e l'adattamento delle espressioni regolari trovate negli esempi dipendono dalla comprensione della struttura delle regole Procmail. Specifiche informazioni introduttive alle espressioni regolari di base sono disponibili nella pagina `man grep`.

Una regola Procmail ha la seguente forma:

```
:0<flag>: <nome-file-lock;
* <carattere-condizione-speciale> <condizione-1>
* <carattere-condizione-speciale> <condizione-2>
* <carattere-condizione-speciale> <condizione-N>

<carattere-condizione-speciale><azione-da-svolgere>
```

I primi due caratteri di una regola Procmail sono i due punti e lo zero. Dopo lo zero possono anche esserci diversi flag per il controllo dell'attività di Procmail durante l'elaborazione di quella regola. La presenza dei due punti dopo la sezione `<flag>` indica che per quel messaggio sarà creato un file di blocco. L'utente dovrà specificare il nome del file nello spazio `<nome-file-lock>`.

Se non vi sono condizioni, ogni messaggio sarà conforme alla regola. Le espressioni regolari sono comprese in alcune condizioni al fine di facilitare una corrispondenza con il messaggio. Se vengono utilizzate delle condizioni multiple, queste devono corrispondere tutte affinché sia possibile eseguire un'azione. Le condizioni sono controllate in base ai flag impostati nella prima riga delle regole. Speciali caratteri opzionali posizionati dopo il carattere `*` aggiungono un ulteriore controllo sulla condizione.

L'`<azione-da-svolgere>` specifica cosa succede a un messaggio che soddisfa una delle condizioni. È consentita una sola azione per regola. In molti casi, qui viene utilizzato il nome di una mailbox per dirigere i messaggi conformi in quel file, smistando efficacemente la posta. Prima che venga specificata l'azione, possono essere utilizzati caratteri per azioni speciali.

15.5.2.1. Regole di distribuzione e non distribuzione

L'azione eseguita se la regola si applica a un particolare messaggio determina se la regola stessa è considerata per la distribuzione o meno. Una *regola per la distribuzione* contiene un'azione che scrive un messaggio su un file, lo invia a un altro programma oppure lo inoltra a un altro indirizzo e-mail. Una *regole non per la distribuzione* copre qualsiasi altra azione, come il blocco del nesting. Il *blocco di nesting* è un'azione contenuta fra parentesi graffe `{ }` che designa azioni aggiuntive da eseguire sui messaggi che soddisfano le condizioni della regola. I blocchi di nesting possono essere gerarchizzati, garantendo così un maggiore controllo per l'identificazione e l'esecuzione di azioni sui messaggi.

Le regole di distribuzione che si applicano ai messaggi consentono a Procmail di eseguire l'azione specificata e di terminare il confronto del messaggio con altre regole. I messaggi che soddisfano le condizioni delle regole di non distribuzione continueranno a essere confrontati con altre regole nei successivi file `rc` corenti. In altre parole, queste ultime regole consentono al messaggio di continuare il controllo delle regole dopo il verificarsi dell'azione specificata.

15.5.2.2. Flag

I flag sono molto importanti per determinare se e come le condizioni della regola sono applicate al messaggio. I seguenti esempi mostrano alcuni flag comunemente usati:

- **A** — specifica che la regola sarà usata solo se anche l'ultima regola applicata senza un flag **A** o **a** è stata applicata al messaggio.
Prima di consentire un controllo con la regola corrente, usate il flag **a** per assicurarvi che l'azione sull'ultima regola applicata sia stata completata con successo.
- **B** — analizza il corpo del messaggio e cerca il soddisfacimento delle condizioni.
- **b** — utilizza il corpo in ogni azione risultante, come la scrittura del messaggio su un file o il suo inoltro. Tale comportamento viene eseguito per default.
- **c** — genera una copia carbone della posta. Ciò risulta utile con le regole di distribuzione, poiché l'azione richiesta può essere eseguita sul messaggio e la copia dello stesso può continuare a essere elaborata nei file `rc`.
- **D** — fa in modo che il confronto `egrep` preveda il riconoscimento dei caratteri maiuscoli. Per default, il processo di confronto non lo prevede.
- **E** — simile al flag **A**, tranne per il fatto che le condizioni in questa regola sono confrontate con il messaggio solo se la regola immediatamente precedente senza un flag **E** non soddisfa le condizioni. Ciò è paragonabile a un'azione *else*.

Utilizzate, invece, il flag `e` se desiderate solo sottoporre al controllo la regola nel caso in cui quella precedente fosse soddisfacente ma l'azione fosse fallita.

- `f` — utilizza pipe come filtro.
- `H` — analizza l'intestazione del messaggio e cerca il soddisfacimento delle condizioni. Ciò viene eseguito per default.
- `h` — utilizza l'intestazione in un'azione risultante. Ciò viene eseguito per default.
- `w` — indica a Procmail di aspettare il filtro specificato o il programma per terminare ed eseguire il report sull'esito dell'operazione prima di considerare filtrato il messaggio.

Usate, invece, l'opzione `w` se desiderate ignorare i messaggi "Program failure" quando decidete se un filtro o un'azione sono riusciti.

Flag aggiuntivi sono disponibili nella pagina man `procmailrc`.

15.5.2.3. Specificare un file di lock locale

I file di lock sono molto utili perché evitano il contemporaneo verificarsi di processi di alterazione su certi messaggi. È possibile specificare un file di lock locale posizionando un carattere `:` dopo ogni flag sulla prima linea della regola. In questo modo verrà creato un file di lock locale basato sul filename di destinazione comprensivo di tutto ciò che è stato impostato nella variabile globale `LOCKEXT`.

In alternativa, è possibile specificare il nome del file di lock locale da utilizzare con la regola dopo il carattere `:`.

15.5.2.4. Condizioni e azioni speciali

Caratteri specifici utilizzati prima delle condizioni delle regole e delle azioni di Procmail cambiano la loro interpretazione.

I seguenti caratteri possono essere utilizzati dopo il carattere `*` all'inizio di una linea delle condizioni delle regole:

- `!` — inverte le condizioni, provocando un controllo solo se le condizioni non sono applicabili al messaggio.
- `<` — controlla che il messaggio non superi un determinato numero di byte.
- `>` — controlla che il messaggio superi un determinato numero di byte.

I seguenti caratteri sono utilizzati per eseguire azioni speciali:

- `!` — indica a Procmail di inoltrare il messaggio all'indirizzo e-mail specificato
- `$` — si riferisce a una variabile precedentemente impostata nel file `rc`. Questo carattere è utilizzato per impostare una comune mailbox cui faranno riferimento varie regole.
- `|` — il carattere pipe indica a Procmail di avviare un programma specifico per il messaggio.
- `{ e }` — crea un blocco di nesting, usato per contenere regole aggiuntive da applicare ai messaggi che soddisfano le condizioni.

Se non sono utilizzati caratteri speciali all'inizio della linea dell'azione, Procmail presume che quest'ultima stia specificando la mailbox in cui scrivere il messaggio.

15.5.2.5. Esempi di regole

Procmail è un programma estremamente flessibile che consente agli utenti di confrontare i messaggi con delle condizioni molto specifiche e di eseguire su di essi azioni dettagliate. Come conseguenza di questa flessibilità, tuttavia, vi è il fatto che comporre una regola per raggiungere un determinato scopo può risultare difficile per i non esperti.

Il miglior modo per sviluppare le abilità per creare le condizioni delle regole Procmail deriva da una profonda comprensione delle espressioni regolari e dall'osservazione degli esempi forniti. I seguenti esempi di semplice comprensione servono come dimostrazione della struttura delle regole Procmail e possono fornire le basi per costrutti più complicati.

Le regole più elementari non contengono nemmeno le condizioni, come illustrato nell'esempio riportato di seguito.

```
:0:
new-mail.spool
```

La prima linea inizia la regola specificando che deve essere creato un file di lock locale senza indicarne il nome, lasciando che Procmail utilizzi il filename di destinazione e il `LOCKEXT` per nominarlo. Poiché non viene specificata alcuna condizione, ogni messaggio potrà soddisfare questa regola e, pertanto, sarà posizionato in un unico file di spool chiamato `new-mail.spool` e nella directory indicata dalla variabile di ambiente `MAILDIR`. I messaggi in questo file possono quindi essere visualizzati da un MUA.

Questa regola di base può posizionarsi alla fine di tutti i file `rc` per indirizzare i messaggi in una posizione predefinita. Un altro esempio, più complesso, è in grado di afferrare i messaggi provenienti da un particolare indirizzo e-mail ed eliminarli, come si può osservare nell'esempio seguente.

```
:0
* ^From: spammer@domain.com
/dev/null
```

In questo esempio, ogni messaggio inviato da `spammer@domain.com` viene immediatamente spostato in `/dev/null`, con conseguente eliminazione.



Attenzione

Assicurarsi che una regola funzioni correttamente prima di spostare i messaggi che la soddisfano in `/dev/null`, poiché la cancellazione è definitiva. Se le condizioni della vostra regola trovano inavvertitamente un messaggio da non sottoporre al controllo, non sarete nemmeno informati della perdita del messaggio, a meno che non vi provveda il mittente.

La soluzione migliore è indirizzare l'azione verso una mailbox speciale che potrete controllare di volta in volta per cercare *falsi positivi*, oppure messaggi che soddisfano inavvertitamente le condizioni. Una volta certi che non vi siano messaggi controllati inavvertitamente, potete cancellare la mailbox e dirigere l'azione per inviare i messaggi a `/dev/null`.

Procmail è usato principalmente come filtro per le e-mail, poiché è in grado di assegnare loro la posizione appropriata così da non doverle smistare manualmente. La regola riportata nell'esempio seguente individua le e-mail inviate da una mailing list particolare e le mette nella cartella corretta per voi.

```
:0:
* ^(From|CC|To). *tux-lug
tuxlug
```


15.6.2. Mail server sicuri

L'offerta della cifratura SSL per gli utenti IMAP e POP sul mail server è piuttosto semplice, poiché Red Hat Linux include anche il pacchetto `stunnel`, un wrapper di cifratura SSL che agisce per alcuni servizi sul traffico di rete standard, non sicuro, ed evita l'intercettazione delle comunicazioni tra client e server.

Il programma `stunnel` utilizza librerie SSL esterne, fra cui le librerie OpenSSL incluse in Red Hat Linux, per fornire un alto livello di crittografia e protezione alle vostre connessioni. È possibile richiedere a una *Certificate Authority (CA)* un certificato SSL, oppure creare un'autocertificazione per usufruire del vantaggio della comunicazione criptata SSL.

Per creare un certificato SSL "auto-firmato", eseguite la modifica nella directory `/usr/share/ssl/certs`, digitate il comando `make stunnel.pem` e rispondete alle domande. Usate quindi `stunnel` per avviare il demone mail che desiderate utilizzare.

Per esempio, i seguenti comandi possono essere utilizzati per avviare il server IMAP incluso in Red Hat Linux:

```
/usr/sbin/stunnel -d 993 -l /usr/sbin/imapd imapd
```

Ora dovrete essere in grado di aprire un client di posta IMAP e di connettervi al server utilizzando la cifratura SSL. Sicuramente, vorrete anche configurare il server IMAP cui applicate lo `stunnel` per avviarsi in automatico sui corretti livelli d'esecuzione.

Per ulteriori informazioni sull'utilizzo di `stunnel`, consultate la pagina `man stunnel` o fate riferimento ai documenti nella directory `/usr/share/doc/stunnel-<numero-versione>`.

In alternativa, il pacchetto `imap` fornito con Red Hat Linux contiene la funzione per disporre la cifratura SSL sul proprio `stunnel`. Per connessioni IMAP sicure, create il certificato SSL modificando la directory `/usr/share/ssl/certs` ed eseguendo il comando `make imapd.pem`. Impostate quindi il servizio `imaps` perché venga eseguito ai runlevel corretti.

È anche possibile utilizzare il pacchetto `ipop3` fornito con Red Hat Linux per fornire la cifratura SSL indipendentemente da `stunnel`.

15.7. Risorse aggiuntive

Molti utenti incontrano qualche difficoltà nel configurare i programmi e-mail. Ciò è dovuto principalmente al vasto numero di opzioni disponibili. Di seguito vi proponiamo un elenco di documenti aggiuntivi utili che vi può aiutare a configurare correttamente le vostre applicazioni di posta.

15.7.1. Documentazione installata

- I pacchetti `sendmail` e `sendmail-cf` comprendono informazioni sulle modalità di configurazione di Sendmail.
 - `/usr/share/doc/sendmail/README.cf` — contiene informazioni su `m4`, i percorsi dei file per Sendmail, i mailer supportati, l'accesso alle caratteristiche potenziate e altro ancora.
 - `/usr/share/doc/sendmail/README` — contiene informazioni sulla struttura della directory Sendmail, il supporto del protocollo IDENT, i dettagli sui permessi alla directory e i più comuni problemi provocati da una configurazione non appropriata dei permessi.

Inoltre, le pagine `man sendmail` e `aliases` contengono, rispettivamente, informazioni utili riguardanti varie opzioni di Sendmail e la corretta configurazione del file `/etc/mail/aliases`.

- `/usr/share/doc/fetchmail-<numero-versione>` — contiene un elenco completo delle caratteristiche Fetchmail contenute nel file `FEATURES` e un documento introduttivo `FAQ`.
- `/usr/share/doc/procmail-<numero-versione>` — contiene un file `README` che fornisce una panoramica di Procmail, un file `FEATURES` che esplora le caratteristiche di ciascun programma, e un file `FAQ` con le risposte a domande comuni sulla configurazione.

Quando si familiarizza con il funzionamento di Procmail e la creazione di nuove regole, queste pagine man di Procmail assumono un valore notevole:

- `procmail` — fornisce una panoramica sul funzionamento di Procmail e le fasi di filtraggio della posta elettronica.
- `procmailrc` — spiega il formato di file usato per creare le regole.
- `procmailex` — fornisce vari esempi utili e già applicati delle regole Procmail.
- `procmailsc` — spiega la tecnica di score utilizzata da Procmail per verificare che una particolare regola è applicabile a un certo messaggio.

15.7.2. Siti Web utili

- <http://www.redhat.com/mirrors/LDP/HOWTO/Mail-Administrator-HOWTO.html> — fornisce una panoramica sul funzionamento delle e-mail ed esamina possibili soluzioni di posta e la configurazione lato client e server.
- <http://www.redhat.com/mirrors/LDP/HOWTO/Mail-User-HOWTO> — considera la posta elettronica dal punto di vista dell'utente, interroga varie applicazioni client e dispone di un'introduzione ad argomenti specifici come alias, inoltro, auto-replying, mailing list, filtri di posta e spam.
- <http://www.redhat.com/mirrors/LDP/HOWTO/mini/Secure-POP+SSH.html> — mostra come rilevare la posta POP usando SSH con il port forwarding in modo che le password e i messaggi possano essere trasferiti in modo sicuro.
- <http://www.sendmail.net> — contiene novità, interviste e articoli su Sendmail, oltre a un'ampia panoramica sulle opzioni disponibili.
- <http://www.sendmail.org> — offre un'approfondita analisi delle caratteristiche di Sendmail ed esempi di configurazione.
- <http://tuxedo.org/~esr/fetchmail> — la home page di Fetchmail, con un manuale online e FAQ approfondite.
- <http://www.procmail.org> — la home page di Procmail, con link a diverse mailing list dedicate a Procmail e FAQ.
- <http://www.ling.helsinki.fi/users/erikss/procmail/mini-faq.html> — un eccellente documento FAQ su Procmail, con suggerimenti per la risoluzione dei problemi e dettagli sul blocco dei file e l'uso di caratteri jolly.
- <http://www.uwasa.fi/~ts/info/proctips.html> — fornisce numerosi suggerimenti che, in molti casi, facilitano l'uso di Procmail, incluso il metodo per verificare i file `.procmailrc` e l'utilizzo dello scoring Procmail per decidere se eseguire una particolare azione.

15.7.3. Libri correlati

- *Sendmail* di Bryan Costales ed Eric Allman e al; O'Reilly & Associates — un valido riferimento su Sendmail scritto con l'assistenza dell'autore originale di Delivermail e Sendmail.

- *Removing the Spam: Email Processing and Filtering* di Geoff Mulligan; Addison-Wesley Publishing Company — un volume che illustra diversi metodi usati dagli amministratori di posta elettronica che utilizzano tool affermati, quali Sendmail e Procmal, per gestire i problemi di spam.
- *Internet Email Protocols: A Developer's Guide* di Kevin Johnson; Addison-Wesley Publishing Company — fornisce una rivisitazione approfondita dei principali protocolli di posta elettronica e la sicurezza che garantiscono.
- *Managing IMAP* di Dianna Mullet e Kevin Mullet; O'Reilly & Associates — illustra in modo dettagliato le fasi necessarie per la configurazione di un server IMAP.

BIND (Berkeley Internet Name Domain)

Oggi giorno, Internet e quasi tutte le reti locali dipendono da una *DNS (Domain Name Service)* funzionante e affidabile, utilizzato per convertire i nomi dei sistemi in indirizzi IP e viceversa.

Per facilitare il DNS sulla vostra rete è necessario un *server dei nomi* che converta questi nomi negli indirizzi IP necessari per stabilire la connessione. Inoltre, un server dei nomi è in grado di riconvertire gli indirizzi IP nel nome di un sistema: questa operazione è comunemente definita *ricerca inversa*.

In questo capitolo verrà trattato BIND, la struttura dei suoi file di configurazione e il modo in cui può essere amministrato in modo locale o remoto.

Per istruzioni sulla configurazione di BIND tramite **Bind Configuration Tool**, `redhat-config-bind`, consultate il capitolo sulla *configurazione di BIND* nella *Official Red Hat Linux Customization Guide*.



Avvertenza

Se usate **Bind Configuration Tool**, ricordatevi di non modificare manualmente i file di configurazione di BIND, poiché qualsiasi modifica manuale viene poi sovrascritta.

16.1. Introduzione a DNS e BIND

I sistemi che utilizzano reti IP devono conoscere l'indirizzo IP di un computer remoto per potersi connettere. Tuttavia, la maggior parte degli utenti preferisce usare il nome di un computer denominato nome host o *Fully qualified domain name (FQDN)*, a cui ci si collega.

L'uso dei nomi FQDN è vantaggioso per gli amministratori di sistema, perché consente una certa flessibilità nella modifica degli indirizzi IP di singoli computer senza influire sulle query basate sui nomi dei computer stessi. Gli amministratori inoltre possono stabilire quale computer gestisce una query basata sui nomi in modo visibile.

Il servizio che facilita queste operazioni è il DNS, il quale viene normalmente implementato utilizzando server centrali che risultano "autorevoli" per alcuni domini e si rivolgono ad altri server DNS per ottenere le informazioni di cui non dispongono.

Il servizio DNS in Linux funziona grazie all'uso di demoni dei *server dei nomi* che eseguono la conversione IP/nome. Un'applicazione client richiede informazioni al server dei nomi, di solito consentendosi ad esso tramite la porta 53 del server. Il server dei nomi cerca di risolvere l'FQDN in base alla sua libreria di conversione, che può contenere informazioni "autorevoli" per l'FQDN in questione oppure dati memorizzati in seguito a una precedente query. Se la libreria di conversione del server dei nomi non contiene già la risposta, esso si rivolgerà ad altri server di nomi, chiamati *server dei nomi root* per determinare quali sono i server di nomi autorevoli per l'FQDN in questione. Poi, con queste informazioni, chiede ai server dei nomi autorevoli il nome per determinare l'indirizzo IP. Se invece si esegue una ricerca inversa, viene utilizzata la stessa procedura, ma la richiesta è inoltrata con un indirizzo IP sconosciuto anziché un nome.

16.1.1. Zone

In Internet, l'FQDN di un host può essere suddiviso in sezioni diverse, organizzate in una gerarchia ad albero con un tronco, dei rami principali, dei rami secondari e così via. Considerate il seguente FQDN:

```
bill.sales.domain.com
```

Per capire come l'FQDN venga convertito nell'indirizzo IP che si riferisce a un determinato sistema, è necessario leggere il nome da destra a sinistra, con ogni livello della gerarchia separato da punti (.). In questo esempio, `com` indica il *dominio ad alto livello* per questo FQDN. `domain` indica un sottodominio di `com` e, a sua volta, `sales` è un sottodominio di `domain`. L'ultimo nome a sinistra in un FQDN è il nome dell'host che identifica una determinato calcolatore.

A eccezione del nome host, ogni sezione è definita *zona* e contraddistingue un particolare spazio dei nomi. Uno *spazio dei nomi* controlla la denominazione dei sottodomini alla propria sinistra. Mentre in questo esempio sono contenuti solo due sottodomini, un FQDN deve contenerne almeno uno, ma può comprenderne molti di più, a seconda di come sono organizzati gli spazi dei nomi.

Le zone sono definite nei server dei nomi autorevoli mediante l'uso di *file della zona*, che descrivono lo spazio dei nomi di quella zona e i server di posta da usare per un dominio o sottodominio particolare. I file della zona sono memorizzati in *server dei nomi primari* (chiamati anche *server master*) che sono autorevoli e consentono la modifica dei file e nei *server dei nomi secondari* (chiamati anche *server slave*), che ricevono i propri file della zona dai server dei nomi primari. Ogni server dei nomi può essere allo stesso tempo primario o secondario per zone diverse e può essere riconosciuto autorevole per più zone. Dipende tutto dalla configurazione del server dei nomi.

16.1.2. Server dei nomi: tipi

Esistono quattro tipi principali di configurazione per i server dei nomi:

- *master* — memorizza i record di zona originali e autorevoli per un determinato spazio dei nomi, rispondendo a richieste di altri server e cercando risposte relative a quello spazio dei nomi.
- *slave* — risponde a richieste di altri server dei nomi relative agli spazi dei nomi sui quali ha autorità. Comunque i server slave ottengono le informazioni sugli spazi dei nomi dai server master tramite un *trasferimento della zona*, dove lo slave invia al master una richiesta di `NOTIFY` per una zona particolare e il master fornisce l'informazione se lo slave è autorizzato a ricevere il trasferimento.
- *caching-only* — offre il nome ai servizi di risoluzione IP ma non è autorevole per tutte le zone. Di norma, le risposte a tutte le risoluzioni vengono memorizzate in un database archiviato in memoria per un determinato periodo di tempo, solitamente specificato dal record di zona recuperato, per garantire una risoluzione più veloce ad altri client DNS dopo la prima risoluzione.
- *forwarding* — inoltra richieste a un elenco specifico di server dei nomi da convertire. Se nessuno dei server specificati può eseguire la risoluzione, il processo s'interrompe e la risoluzione non viene completata.

Un server dei nomi può essere di uno o più tipi tra quelli descritti. Per esempio può essere un master per alcune zone e uno slave per altre e può offrire solo una risoluzione forwarding.

16.1.3. BIND come server dei nomi

Red Hat Linux contiene BIND, un server dei nomi open source molto diffuso e potente. BIND utilizza il demone `named` per fornire i propri servizi di risoluzione dei nomi.

La versione 9 di BIND comprende inoltre un'utility chiamata `rndc`, che consente all'amministratore di eseguire il demone `named`. Per maggiori informazioni su `rndc`, consultate la Sezione 16.3.

16.2. File di configurazione di BIND

Il file di configurazione di BIND è `/etc/named.conf`. Tutte i file della zona si trovano nella directory `/var/named`.

Il file `/etc/named.conf` non deve contenere errori per l'avvio di `named`. Mentre l'utilizzo di alcune opzioni errate all'interno delle istruzioni non compromette il funzionamento del server, qualsiasi errore contenuto nelle istruzioni stesse impedisce l'avvio del demone.



Avvertenza

Non modificate manualmente il file `/etc/named.conf` o qualsiasi file nella directory `/var/named` se state utilizzando l'applicazione **Bind Configuration Tool**. Tutte le modifiche manuali apportate a questi file verranno, infatti, sovrascritte al successivo utilizzo del **Bind Configuration Tool**.

16.2.1. `/etc/named.conf`

Il file `/etc/named.conf` è un insieme di istruzioni che utilizza opzioni nidificate tra parentesi graffe `{ }`. Un file `/etc/named.conf` ha una struttura simile a quella riportata di seguito:

```
<istruzione-1> [ "<nome-istruzione-1>" ] [ <classe-istruzione-1> ] {
    <opzione-1>;
    <opzione-2>;
    <opzione-N>;
};

<istruzione-2> [ "<nome-istruzione-2>" ] [ <classe-istruzione-2> ] {
    <opzione-1>;
    <opzione-2>;
    <opzione-N>;
};

<istruzione-N> [ "<nome-istruzione-N>" ] [ <classe-istruzione-N> ] {
    <opzione-1>;
    <opzione-2>;
    <opzione-N>;
};
```

Il `<nome-istruzione-N>` è necessario solo con le istruzioni `acl`, `include`, `server`, `view` e `zone`. La `<classe-istruzione-N>` può essere specificata solo con l'istruzione `zone`.

Di seguito è riportato un elenco di tag di commento validi che potete utilizzare in `/etc/named.conf`:

- `//` — Utilizzatelo all'inizio di una linea per commentarla.
- `#` — Utilizzatelo all'inizio di una linea per commentarla.
- `/* e */` — Includete del testo tra questi tag per creare un commento di un blocco.

Nel file `/etc/named.conf` è possibile utilizzare le istruzioni seguenti:

- `acl <acl-nome>4` — configura un elenco di controllo accessi degli indirizzi IP da abilitare o disabilitare per determinati servizi `named`. La maggior parte delle volte vengono utilizzati gli indirizzi IP individuali o le notazioni di rete IP (come `10.0.1.0/24`) per identificare gli indirizzi IP esatti.

Alcuni elenchi di controllo accessi sono già specificati, dunque non è necessario configurare un'istruzione `acl` per definirli:

- `any` — corrisponde a ogni indirizzo IP.
- `localhost` — corrisponde a qualsiasi indirizzo IP in uso nel sistema locale.
- `localnets` — corrisponde a qualsiasi indirizzo IP in qualsiasi rete a cui il sistema locale è connesso.
- `none` — non corrisponde a nessun indirizzo IP.

Se utilizzate con altre istruzioni di `/etc/named.conf` e relative opzioni, le istruzioni `acl` possono essere molto utili nell'assicurare un uso corretto del server dei nomi BIND come nell'esempio riportato di seguito:

```
acl black-hats {
    10.0.2.0/24;
    192.168.0.0/24;
};

acl red-hats {
    10.0.1.0/24;
};

options {
    blackhole { black-hats; };
    allow-query { red-hats; };
    allow-recursion { red-hats; };
}
```

Questo file contiene due elenchi di controllo accessi `named.conf` (`black-hats` e `red-hats`).

- `controls` — configura diversi requisiti di sicurezza richiesti per l'utilizzo del comando `rndc`, necessario per amministrare il servizio `named`.

Per vedere la struttura di un'istruzione `controls` e le opzioni che possono essere utilizzate solo con essa, consultate la Sezione 16.3.1.1

- `include "<nome-file>"` — comprende un file all'interno del file di configurazione in uso e fa sì che i dati di configurazione riservati (come `keys`) possano essere posizionati in un file diverso con permessi che ne impediscano la lettura da parte di utenti non privilegiati.
- `key "<nome-chiave>"` — definisce una determinata chiave per nome. Le chiavi sono utilizzate per autenticare varie azioni, come gli aggiornamenti sicuri o l'uso del comando `rndc`. Con `key` vengono utilizzate due opzioni:
 - `algorithm <nome-algoritmo>` — indica il tipo di algoritmo usato, come `dsa` o `hmac-md5`.
 - `secret "<valore-chiave>"` — chiave crittografata.

Per un esempio di istruzione `key`, consultate la la Sezione 16.3.1.2.

- `logging` — permette l'utilizzo di diversi tipi di registri, chiamati *canali*. Usando l'opzione `channel` all'interno dell'istruzione `logging`, è possibile creare un tipo di registro personalizzato, con un proprio nome file (`file`), un limite di dimensione (`size`), la versione (`version`) e il livello d'importanza (`severity`). Una volta definito un canale personalizzato, viene usata l'opzione `category` per classificare il canale ed eseguire la registrazione al riavvio di `named`.

`named` trasmette, per default, i messaggi standard al demone `syslog`, che li posiziona in `/var/log/messages`. Ciò avviene perché in BIND diversi canali standard sono creati con vari livelli di importanza: uno per esempio gestisce messaggi di registrazione (`default_syslog`) e un altro gestisce, nello specifico, messaggi di debug (`default_debug`). Una categoria predefinita, chiamata `default`, utilizza canali incorporati per effettuare le normali registrazioni senza alcuna configurazione speciale.

Personalizzare il processo di registrazione può richiedere un lavoro molto minuzioso che esula dallo scopo di questo capitolo. Per informazioni su come creare delle registrazioni personalizzate in BIND, fate riferimento al *BIND 9 Administrator Reference Manual*.

- `options` — assegna valori a molte opzioni, come l'uso di `forwarder`, la posizione della directory di lavoro `named`, i nomi dei vari file e molto altro.

Ecco alcune delle opzioni più diffuse:

- `allow-query` — specifica gli host autorizzati a interrogare questo server dei nomi, (per default, tutti gli host sono autorizzati). Un elenco di controllo accessi o un insieme degli indirizzi o reti IP può essere utilizzato, in questo caso, solo per autorizzare degli host particolari a interrogare il server dei nomi.
- `allow-recursion` — simile a `allow-query`, tranne per il fatto che viene utilizzato per richieste ricorsive. Per default, tutti gli host sono autorizzati ad effettuare richieste ricorsive ai server dei nomi.
- `directory` — modifica la directory di lavoro `named` in una directory diversa da quella predefinita (`/var/named`).
- `forward` — controlla il modo in cui avviene l'inoltro, se l'opzione `forwarders` contiene indirizzi IP validi che indicano dove inviare le richieste.

Se viene utilizzata l'opzione `first`, i server dei nomi specificati nell'opzione `forwarders` vengono interrogati per primi e se non dispongono delle informazioni richieste `named` tenta di eseguire la risoluzione da solo.

Se viene usata l'opzione `only`, `named` non tenta di eseguire la risoluzione se i `forwarder` non hanno buon esito.

- `forwarders` — specifica un elenco di server dei nomi a cui inviare delle richieste di risoluzione.
- `listen-on` — specifica l'interfaccia di rete che `named` utilizza per ricevere le interrogazioni. Per default, vengono utilizzate tutte le interfacce.

Quest'opzione è particolarmente utile se disponete di diverse interfacce di rete e vorreste limitare il numero di sistemi che possono richiedere il vostro server dei nomi. Per esempio se possedete una macchina che funge da gateway e da server dei nomi e vorreste bloccare tutte le richieste a eccezione di quelle originate dalla vostra rete privata, impostate l'opzione `listen-on` come descritto di seguito:

```
options {
    listen-on { 10.0.1.1; };
};
```

In tal modo vengono accettate solo le richieste che provengono dall'interfaccia che funge da rete privata (10.0.1.1).

- `notify` — controlla se `named` invia una notifica ai server slave quando viene aggiornata una zona. Per default, è impostato `yes`, ma è possibile impostare `no`, per evitare che gli slave ricevano notifiche o per specificare che vengano notificati solo i server presenti nell'elenco `also-notify`.
- `pid-file` — vi consente di specificare la posizione del file di processo ID creato da `named` al suo avvio.
- `statistics-file` — vi consente di specificare la posizione in cui è stato salvato il file delle statistiche. Per default, le statistiche di `named` vengono salvate in `/var/named/named.stats`.

Sono inoltre disponibili decine di altre opzioni, molte delle quali dipendono l'una dall'altra per poter funzionare correttamente. Per maggiori dettagli, consultate il *BIND 9 Administrator Reference Manual*.

- `server` — definisce delle opzioni particolari che influiscono sul modo in cui `named` deve agire nei confronti di server dei nomi remoti, soprattutto nel caso di notifiche e trasferimenti di zona.

L'opzione `transfer-format` stabilisce se inviare un solo record di risorsa con ogni messaggio (`one-answer`) o inviare record di risorsa multipli con ogni messaggio (`many-answers`). Sebbene `many-answers` sia il più efficiente, solo i server dei nomi BIND più recenti lo supportano.

- `trusted-keys` — contiene chiavi pubbliche utilizzate per DNSSEC. Per informazioni introduttive sulla sicurezza di BIND, consultate la Sezione 16.4.3.
- `view "<nome-visualizzazione>"` — crea modalità di visualizzazione speciali che rispondono con un tipo particolare di informazione e dipendono dall'host che contatta il server dei nomi. Ciò consente ad alcuni host di ricevere una risposta relativa a zone particolari mentre gli altri host ricevono informazioni completamente diverse. In alternativa alcune zone possono essere disponibili solo per determinati host fidati mentre gli host non affidabili possono comunque inoltrare richieste per altre zone.

Le visualizzazioni multiple possono essere utilizzate se i loro nomi sono unici. L'opzione `match-clients` specifica gli indirizzi IP validi per una determinata visualizzazione. Qualsiasi istruzione `option` può anche essere utilizzata all'interno di una visualizzazione, annullando le opzioni globali già configurate per `named`. La maggior parte delle istruzioni `view` contiene varie istruzioni zone valide per l'elenco `match-clients`. L'ordine in cui vengono elencate le istruzioni `view` è importante, poiché viene utilizzata la prima istruzione `view` che corrisponde all'indirizzo IP di un determinato client.

Per maggiori informazioni sull'istruzione `view`, consultate la Sezione 16.4.2.

- `zone "<nome-zona>"` — specifica determinate zone per le quali il server dei nomi è autorevole. L'istruzione `zone` è usata principalmente per specificare il file contenente la configurazione della zona e per trasmettere a `named` determinate opzioni su quella zona che annullano le altre istruzioni `option` generali utilizzate nel file `/etc/named.conf`.

Il nome della zona è importante, poiché si tratta del valore predefinito assegnato alla direttiva `$ORIGIN` utilizzata nel file della zona e accodata a nomi di dominio incompleti. Quindi, per esempio, se tale istruzione di zona definisce lo spazio dei nomi per `domain.com`, questo andrebbe utilizzato come `<nome-zona>` e posizionato alla fine del nome host utilizzato nel file della zona.

Le opzioni più comuni dell'istruzione `zone` comprendono:

- `allow-query` — specifica i client autorizzati a richiedere informazioni su questa zona. Per default, vengono autorizzate tutte le richieste.
- `allow-transfer` — specifica i server slave autorizzati a richiedere un trasferimento delle informazioni sulla zona. Per default, vengono autorizzate tutte le richieste di trasferimento.
- `allow-update` — specifica gli host autorizzati ad aggiornare dinamicamente le informazioni nelle proprie zone. Per default, vengono negate tutte le richieste di aggiornamento dinamico.

Prestate molta attenzione nell'autorizzare gli host ad aggiornare le informazioni relative alle proprie zone. Non abilitate questa opzione, a meno che l'host specificato non sia completamente fidato. Sarebbe opportuno che fosse un amministratore a occuparsi di aggiornare manualmente i record delle zone e a ricaricare, se possibile, il servizio `named`.

- `file` — specifica il nome del file nella directory operativa di `named` (per default `/var/named`) che contiene i dati di configurazione della zona.
- `masters` — utilizzato se la zona viene definita come di tipo slave. L'opzione `masters` indica al demone `named` di un server slave l'indirizzo IP a cui richiedere informazioni sulla zona autorevole.
- `notify` — opera in modo simile all'opzione `notify` utilizzata nell'istruzione `option`.
- `type` — definisce il tipo di zona. I tipi possibili sono:
 - `forward` — chiede al server dei nomi di reindirizzare ad altri server tutte le richieste di informazioni su questa zona.
 - `hint` — tipo speciale di zona usato per fare riferimento ai server dei nomi root, utilizzati per risolvere richieste quando una zona è sconosciuta. In genere non è necessario configurare una zona del tipo `hint` oltre a quella predefinita nel file `/etc/named.conf`.

- `master` — definisce il server dei nomi autorevole per questa zona. Occorre impostare una zona del tipo `master` se nel vostro sistema vi sono i file di configurazione della zona.
- `slave` — definisce il server come slave per questa zona, ordinando a `named` di richiedere i file di configurazione della zona dall'indirizzo IP del server master per quella zona.
- `zone-statistics` — ordina al demone `named` di conservare le statistiche relative alla zona, memorizzandole nella posizione predefinita (`/var/named/named.stats`) o in una posizione definita con l'opzione `statistics-file` nell'istruzione `server`, se esistente.

16.2.1.1. Istruzioni di zona: esempi

La maggior parte delle modifiche al file `/etc/named.conf` di un server dei nomi master o slave riguarda l'aggiunta, la modifica o la cancellazione di istruzioni `zone`. Sebbene queste istruzioni possano contenere molte opzioni, quasi tutti i server dei nomi ne usano solo alcune. Le istruzioni `zone` che seguono sono alcuni esempi di base da utilizzare in una relazione master/slave.

Quello riportato di seguito è un esempio di istruzione `zone` per il server dei nomi primario con dominio `domain.com`.

```
zone "domain.com" IN {
    type master;
    file "domain.com.zone";
    allow-update { none; };
};
```

All'interno dell'istruzione la zona è identificata come `domain.com`, il tipo è impostato a `master` e indica a `named` di leggere il file `/var/named/domain.com.zone`. Indica inoltre a `named` di non consentire l'aggiornamento da parte di nessun altro host.

Un'istruzione `zone` di un server slave per `domain.com` è leggermente diversa dall'esempio precedente. Per un server slave il tipo è impostato a `slave` e invece della riga `allow-update` è presente una direttiva che indica a `named` l'indirizzo IP del server master.

L'istruzione di zona del server slave per `domain.com` è simile a quanto riportato di seguito:

```
zone "domain.com" {
    type slave;
    file "domain.com.zone";
    masters { 192.168.0.1; };
};
```

Questa istruzione `zone` indica a `named` nel server slave di cercare le informazioni di configurazione per la zona chiamata `domain.com` nel server master all'indirizzo IP `192.168.0.1`. Le informazioni che il server slave riceve dal server master vengono salvate in `/var/named/domain.com.zone`.

16.2.2. File della zona

I *file della zona*, contenenti le informazioni relative a un determinato spazio dei nomi, sono memorizzati nella directory operativa di `named` (per default `/var/named`). Il nome di ogni file della zona dipende dai dati indicati nell'opzione `file` contenuta nell'istruzione `zone`. In genere si riferisce al dominio in questione e identifica il file poiché contiene dati sulla zona, come per esempio `example.com.zone`.

Ogni file della zona può contenere direttive e record di risorsa. Le *direttive* indicano al server dei nomi di eseguire una certa azione o di eseguire un'impostazione speciale per la zona. I *record di risorsa* definiscono i parametri della zona attribuendo un'identità a determinati sistemi nello spazio dei nomi della zona. Le direttive sono facoltative, mentre i record di risorsa sono necessari per fornire il servizio dei nomi a quella zona. Tutte le direttive e i record di risorsa dovrebbero occupare righe diverse.

Nei file della zona è possibile aggiungere dei commenti dopo il carattere punto e virgola (;).

16.2.2.1. Direttive dei file della zona

Le direttive sono contraddistinte dal carattere \$ che precede il nome della direttiva e che di solito si trova all'inizio del file della zona.

Le direttive più usate sono le seguenti:

- `$INCLUDE` — indica a `named` di includere un file della zona in un altro file nel punto in cui viene usata la direttiva. Ciò consente di memorizzare impostazioni di zona aggiuntive separatamente dal file della zona principale.
- `$ORIGIN` — imposta il nome del dominio da accodare a qualsiasi record non qualificato, come quelli che specificano solo l'host e nient'altro.

Per esempio, un file della zona potrebbe contenere una riga come la seguente:

```
$ORIGIN domain.com
```

A questo punto a qualsiasi nome utilizzato nei record di risorsa, che non finisca con un punto (.), verrà aggiunto questo nome del dominio. In altre parole, quando il record di zona viene letto dal server dei nomi, la prima riga illustrata qui di seguito viene interpretata come la seconda:

```
ftp                IN      CNAME    server1
ftp.domain.com.   IN      CNAME    server1.domain.com.
```



Nota bene

Non è necessario usare la direttiva `$ORIGIN` se alla zona si assegna un nome nel file `/etc/named.conf` identico al valore che assegnereste a `$ORIGIN`. Il nome della zona viene utilizzato, per default, come valore della direttiva `$ORIGIN`.

- `$TTL` — imposta il valore predefinito *Time to Live (TTL)* per la zona. Si tratta di un valore, in secondi, assegnato ai server dei nomi che indica il periodo di validità dei record di risorsa della zona. Un record di risorsa può avere un valore TTL proprio, che annulla quindi quello impostato da questa direttiva.

Impostando un valore più alto si indica ai server dei nomi di conservare in memoria queste informazioni di zona per un periodo di tempo maggiore. Ciò riduce il numero di richieste relative a questa zona, ma allunga anche il tempo necessario per modificare il record di risorse.

16.2.2.2. Record di risorsa del file della zona

I record di risorsa del file della zona contengono colonne di dati separate da spazi bianchi. A tutti i record di risorsa viene attribuito un tipo particolare che determina lo scopo del record. I seguenti tipi sono tra i più usati:

- `A` — record di indirizzo che specifica un indirizzo IP da assegnare al nome, come in questo esempio:

```
<host>      IN      A      <Indirizzo-IP>
```

Se non viene indicato il valore `<host>`, il record `A` fa riferimento a un indirizzo IP predefinito per l'inizio dello spazio dei nomi. Questo sistema è utilizzato per tutte le richieste non FQDN.

Prendete in considerazione i seguenti esempi di record A per il file della zona `domain.com`:

```

server1      IN      A      10.0.1.3
server1      IN      A      10.0.1.5

```

Le richieste per `domain.com` vengono indirizzate a 10.0.1.3, mentre quelle per `server1.domain.com` a 10.0.1.5.

- CNAME — record di nome tipico che mappa un nome all'altro, cioè un alias.

L'esempio successivo indica a `named` che tutte le richieste inviate a `<nome-alias>` sono dirette all'host `<nome-reale>`. I record CNAME sono utilizzati più comunemente per essere diretti a servizi che utilizzano uno schema di assegnazione dei nomi comune per l'host corretto.

```

<nome-alias>      IN      CNAME   <nome-reale>

```

Considerate l'esempio riportato di seguito, in cui il record A imposta un indirizzo IP per un determinato nome di host e il record CNAME evidenzia come nome alias il nome host più comunemente usato `www`.

```

server1      IN      A      10.0.1.5
www          IN      CNAME  server1

```

- MX — si tratta del record "Mail eXchange", che indica dove va inoltrata la posta inviata a un particolare spazio dei nomi controllato da questa zona.

```

IN      MX      <valore-preferenza> <nome-server-email>

```

In questo esempio il `<valore-preferenza>` vi consente di classificare numericamente i server e-mail su cui preferite ricevere la posta elettronica per questo spazio dei nomi, dando la precedenza ad alcuni sistemi e-mail rispetto ad altri. Il record di risorsa MX con il `<valore-preferenza>` più basso ha la precedenza sugli altri, ma potete comunque impostare vari server di posta elettronica con lo stesso valore e distribuire quindi il traffico di posta.

Il `<nome-server-email>` può essere un nome host o un FQDN, purché faccia riferimento al sistema corretto.

```

IN      MX      10      mail.domain.com.
IN      MX      20      mail2.domain.com.

```

In questo esempio il primo server di posta `mail.domain.com` ha la precedenza sul server `mail2.domain.com` al momento della ricezione di posta per il dominio `domain.com`.

- NS — record che annuncia i server dei nomi autorevoli per una determinata zona.

```

IN      NS      <nome-servernomi>

```

L'opzione `<nome-servernomi>` deve corrispondere a un FQDN.

Di seguito sono elencati due nomi di server come autorevoli per un dominio. Non importa se questi server dei nomi sono slave o master, poiché entrambi sono considerati autorevoli.

```

IN      NS      dns1.domain.com.
IN      NS      dns2.domain.com.

```

- PTR — record "PoinTeR" che serve per fare riferimento a un'altra "porzione" dello spazio dei nomi.

I record PTR vengono usati principalmente per invertire la risoluzione del nome, riconvertendo gli indirizzi IP in un particolare nome. Per maggiori esempi sull'opzione PTR in uso, consultate la Sezione 16.2.2.4.

- SOA — acronimo di "Start Of Authority", questo record indica al server dei nomi importanti informazioni autorevoli sullo spazio nomi.

Posizionato dopo le direttive, SOA è il primo record di risorsa in un file della zona.

L'esempio riportato di seguito mostra la struttura di base di un record SOA:

```

@      IN      SOA      <nome-server-primario>    <email-hostmaster> (
                                <numero-seriale>
                                <tempo-per-aggiornamento>
                                <tempo-per-riprovare>
                                <tempo-di-scadenza>
                                <TTL-minimo> )

```

Il simbolo @ serve a posizionare la direttiva \$ORIGIN (o il nome della zona, se la direttiva non è impostata) come spazio dei nomi definito dal record di risorsa SOA. In `<nome-server-primario>` va indicato il server dei nomi primario e autorevole per questo dominio, mentre in `<email-hostmaster>` va inserito l'indirizzo e-mail della persona da contattare per questo spazio dei nomi.

Il `<numero-seriale>` viene incrementato ogni volta che modificate il file della zona, affinché named riceva l'informazione di ricaricare questa zona. Il `<tempo-per-aggiornamento>` indica a ogni server slave quanto aspettare prima di chiedere al server dei nomi master se sono state effettuate delle modifiche per quella zona. Il valore `<numero-seriale>` è utilizzato dallo slave per determinare se sta usando dati di zona obsoleti e deve dunque aggiornarli.

Il `<tempo-per-riprovare>` indica al server dei nomi slave quanto attendere prima di formulare un'altra richiesta di aggiornamento, se il server dei nomi master non risponde. Se il master non ha risposto alla richiesta di aggiornamento entro il `<tempo-di-scadenza>`, il server slave cessa di essere un'autorità per quello spazio dei nomi.

Il `<TTL-minimo>` richiede che gli altri server dei nomi memorizzino le informazioni di zona per almeno questo lasso di tempo (in secondi).

In BIND il tempo viene riportato in secondi. Comunque potete utilizzare anche delle abbreviazioni per altre unità di tempo, come minuti (M), ore (H), giorni (D) e settimane (W). La tabella in Tabella 16-1 mostra un lasso di tempo in secondi e l'equivalente in un'altra unità.

Secondi	Altre unità di tempo
60	1M
1800	30M
3600	1H
10800	3H
21600	6H
43200	12H
86400	1D
259200	3D
604800	1W

Tabella 16-1. Secondi paragonati ad altre unità di tempo

L'esempio seguente mostra la struttura che un record di risorsa di base SOA potrebbe avere se configurato con valori reali.

```
@      IN      SOA      dns1.domain.com.    hostmaster.domain.com. (
                                2001062501 ; serial
                                21600      ; refresh after 6 hours
                                3600      ; retry after 1 hour
                                604800   ; expire after 1 week
                                86400    ) ; minimum TTL of 1 day
```

16.2.2.3. Esempi di file della zona

Le direttive e i record di risorsa, visti individualmente, possono essere difficili da comprendere. Comunque, tutto ha molto più senso se riunito in un unico file.

```
$ORIGIN domain.com
$TTL 86400
```



```

        3600      ; retry after 1 hour
        604800   ; expire after 1 week
        86400 )  ; minimum TTL of 1 day

IN      NS      dns1.domain.com.
IN      NS      dns2.domain.com.

20     IN      PTR    alice.domain.com.
21     IN      PTR    betty.domain.com.
22     IN      PTR    charlie.domain.com.
23     IN      PTR    doug.domain.com.
24     IN      PTR    ernest.domain.com.
25     IN      PTR    fanny.domain.com.

```

Questo file della zona viene utilizzato con l'istruzione `zone` nel file `/etc/named.conf` simile a quello riportato di seguito:

```

zone "1.0.10.in-addr.arpa" IN {
    type master;
    file "domain.com.rr.zone";
    allow-update { none; };
};

```

Esiste una differenza davvero minima tra questo esempio e un'istruzione standard `zone`, salvo per il nome della zona. Una zona per la risoluzione inversa dei nomi richiede che siano invertiti i primi tre blocchi dell'indirizzo IP e che dopo di questi venga aggiunto `".in-addr.arpa"`. Ciò permette che il blocco singolo di numeri IP utilizzato nel file della zona per la risoluzione inversa dei nomi venga collegato correttamente in questa zona.

16.3. Uso di `rndc`

BIND dispone di un'utility chiamata `rndc` che vi consente di amministrare il demone `named` in modo locale o remoto tramite istruzioni dalla linea di comando. Il programma `rndc` utilizza il file `/etc/rndc.conf` per le proprie opzioni di configurazione, che possono essere sovrascritte con opzioni dalla linea di comando.

Per impedire che utenti non autorizzati di altri sistemi controllino BIND sul vostro server, è possibile utilizzare un metodo a chiave segreta condivisa per garantire in modo esplicito dei privilegi a determinati host. Affinché `rndc` possa impartire dei comandi a qualsiasi demone `named`, perfino in una macchina locale, le chiavi usate in `/etc/named.conf` e `/etc/rndc.conf` devono coincidere.

16.3.1. Configurazione di `rndc`

Prima di utilizzare il comando `rndc`, verificate che le righe di configurazione esatte si trovino nei file necessari. Molto probabilmente i vostri file di configurazione non sono impostati correttamente per eseguire `rndc` e per questo verrà visualizzato il seguente messaggio:

```
rndc: connect: connection refused
```

16.3.1.1. `rndc` e `/etc/named.conf`

Per consentire a `rndc` di connettersi al servizio `named`, è necessario disporre dell'istruzione `controls` nel proprio file `/etc/named.conf` all'avvio di `named`. L'esempio dell'istruzione `controls` riportato di seguito vi permette di eseguire i comandi `rndc` a livello locale.

```
controls {
    inet 127.0.0.1 allow { localhost; } keys { <nome-chiave>; };
};
```

Questa istruzione indica a `named` di attendere sulla porta TCP 953 predefinita dell'indirizzo di loop-back e autorizza i comandi `rndc` provenienti dall'host locale, su corretta indicazione della chiave. Il `<nome-chiave>` si riferisce all'istruzione `key` contenuta anch'essa nel file `/etc/named.conf`. L'esempio riportato di seguito illustra l'istruzione `key`.

```
key "<nome-chiave>" {
    algorithm hmac-md5;
    secret "<valore-chiave>";
};
```

In questo caso, il `<valore-chiave>` è una chiave HMAC-MD5. Potete generare le vostre chiavi HMAC-MD5 con il seguente comando:

```
dnssec-keygen -a hmac-md5 -b <lunghezza-bit> -n HOST <nome-file-chiave>
```

È consigliabile una chiave con una lunghezza minima di 256 bit. La chiave effettiva da inserire nell'area `<valore-chiave>` si trova nel `<nome-file-chiave>`.

Il nome della chiave utilizzata in `/etc/named.conf` deve, ovviamente, essere diverso da `key`.

16.3.1.2. /etc/rndc.conf

Per configurare `rndc` in modo tale che utilizzi automaticamente le chiavi specificate nel file `/etc/named.conf`, è necessario aggiungere le righe riportate di seguito. Questa operazione viene effettuata con un'istruzione `options`:

```
options {
    default-server localhost;
    default-key "<nome-chiave>";
};
```

Questo comando imposta una chiave predefinita globale, ma il comando `rndc` può anche utilizzare chiavi diverse per server particolari, come nell'esempio riportato di seguito:

```
server localhost {
    key "<nome-chiave>";
};
```

Tuttavia, l'istruzione `server` è davvero utile solo se ci si connette a diversi server tramite `rndc`.

`key` è l'istruzione più importante contenuta nel file `/etc/rndc.conf`.

```
key "<nome-chiave>" {
    algorithm hmac-md5;
    secret "<valore-chiave>";
};
```

Il `<nome-chiave>` e il `<valore-chiave>` devono avere le stesse impostazioni indicate nel file `/etc/named.conf`.

Per controllare tutte le impostazioni, usate il comando `rndc reload`. Verrà visualizzato il seguente output:

```
rndc: reload command successful
```

Se invece il comando non ha esito positivo, scorrete attentamente i file `/etc/named.conf` e `/etc/rndc.conf` alla ricerca di eventuali errori.

16.3.2. Opzioni della linea di comando `rndc`

Un comando `rndc` ha la seguente forma:

```
rndc <opzioni> <comando> <opzioni-comando>
```

L'area `<opzioni>` non è obbligatoria e non occorre che usiate le `<opzioni-comando>` a meno che il comando non le richieda.

Quando si esegue `rndc` in un host locale configurato in modo corretto, è possibile utilizzare i seguenti comandi:

- `halt` — interrompe immediatamente il servizio `named`.
- `querylog` — attiva la registrazione delle richieste effettuate dai client a questo server dei nomi.
- `refresh` — aggiorna il database del server dei nomi.
- `reload` — indica al server dei nomi di ricaricare i file della zona, ma di non cancellare tutti i responsi memorizzati in precedenza. Ciò vi consente di effettuare delle modifiche ai file della zona e di applicarle sui vostri server master e slave senza perdere tutte le risoluzioni di nomi archiviate.

Se le modifiche riguardano solo una zona, potete indicare a `named` di ricaricare solo quella zona digitando il nome della zona dopo il comando `reload`.

- `stats` — trasferisce le attuali statistiche di `named` nel file `/var/named/named.stats`.
- `stop` — interrompe il server in modo tale da salvare tutti gli aggiornamenti dinamici e i dati IXFR prima di uscire.

Se desiderate annullare le impostazioni predefinite nel file `/etc/rndc.conf`, sono disponibili le seguenti opzioni:

- `-c <file-configurazione>` — indica a `rndc` di utilizzare un file di configurazione diverso da quello predefinito `/etc/rndc.conf`.
- `-p <numero-porta>` — indica un numero di porta differente per la connessione di `rndc` (anziché il predefinito 953).
- `-s <server>` — indica a `rndc` di inviare il comando a un server diverso da quello indicato nell'opzione `server-default` nel file `/etc/rndc.conf`.

Questa opzione funziona solo se l'altro servizio `named` è già stato configurato per accettare comandi dal vostro host e se disponete della chiave per quel servizio dei nomi.

- `-y <nome-chiave>` — vi consente di specificare una chiave diversa da quella indicata dall'opzione `chiave-predefinita` nel file `/etc/rndc.conf`.

Per ulteriori informazioni su queste opzioni, consultate la pagina `man` di `rndc`.

16.4. BIND: caratteristiche avanzate

La maggior parte delle versioni di BIND utilizza `named` per fornire servizi di risoluzione nomi o per fungere da autorità per un particolare dominio o sottodominio. Tuttavia la versione 9 di BIND comprende una serie di caratteristiche avanzate che, se opportunamente configurate e utilizzate, garantiscono un servizio DNS più sicuro ed efficiente.



Avvertenza

Alcune di queste caratteristiche avanzate, come DNSSEC, TSIG e IXFR, andrebbero usate solo in ambienti di rete con server dei nomi che le supportino. Se la vostra rete comprende server dei nomi non BIND o con versioni precedenti, verificate che una determinata caratteristica avanzata sia disponibile, prima di cercare di utilizzarla.

Ricordate che molti tipi di server dei nomi non supportano tutte queste caratteristiche.

Tutte le caratteristiche trattate in questo paragrafo vengono approfondite nel *BIND 9 Administrator Reference Manual*. Per sapere dove reperire questo manuale, consultate la Sezione 16.6.

16.4.1. Miglioramenti del protocollo DNS

BIND supporta i *trasferimenti di zona incrementali (IXFR)*, grazie ai quali i server slave effettuano il download solo delle parti aggiornate (o modificate) di una zona in un server master. Invece il processo di trasferimento standard AXFR richiede che l'intera zona venga trasferita a ogni server slave, perfino per la più piccola modifica. Per domini molto diffusi con file della zona lunghi e molti server slave, IXFR semplifica i processi di notifica e aggiornamento.

IXFR è disponibile solo se utilizzate un *aggiornamento dinamico* per modificare i record di zona master. Se invece modificate manualmente i file della zona, viene utilizzato AXFR. Per maggiori informazioni sull'aggiornamento dinamico, consultate il *BIND 9 Administrator Reference Manual*.

16.4.2. Visualizzazioni multiple

Mediante l'istruzione `view` del file `/etc/named.conf` BIND consente di configurare un server dei nomi in modo che risponda diversamente alle richieste di alcuni client rispetto ad altri.

Questa possibilità è utile soprattutto se desiderate che i client esterni alla vostra rete non possano eseguire un determinato servizio DNS o vedere un determinato tipo di informazioni, ma non volete invece escludere i client interni.

L'istruzione `view` utilizza l'opzione `match-clients` per far corrispondere indirizzi IP o reti intere e per offrire loro opzioni speciali e dati di zona.

16.4.3. Sicurezza

BIND supporta vari metodi di protezione per l'aggiornamento e il trasferimento di zone, in entrambi i server master e slave:

- *DNSSEC* — abbreviazione di *DNS SECURITY*, questa caratteristica consente di cifrare alcune zone con una *chiave zona*.

In tal modo le informazioni relative a zone specifiche possono essere verificate come provenienti da un server che le ha firmate con una determinata chiave privata, se il destinatario possiede la chiave pubblica del server dei nomi.

La versione 9 di BIND supporta inoltre il metodo SIG(0) chiave pubblica/privata per l'autenticazione del messaggio.

- *TSIG* — abbreviazione di *Transaction SIGNatures*, si tratta di una chiave segreta condivisa, presente sui server master e slave, che verifica l'autorizzazione a un trasferimento da master a slave.

Questa caratteristica rafforza il metodo IP standard basato sull'indirizzo per l'autorizzazione al trasferimento. Infatti un eventuale intruso per poter trasferire la zona non solo dovrebbe conoscere l'indirizzo IP ma anche la chiave segreta.

La versione 9 di BIND supporta inoltre *TKEY*, ovvero un altro metodo a chiave segreta condivisa per l'autorizzazione a trasferimenti di zona.

16.4.4. IP versione 6

La versione 9 di BIND può fornire il servizio dei nomi in ambienti IP versione 6 (IPv6) utilizzando i record di zona A6.

Se la vostra rete comprende host con IPv4 e IPv6, è necessario usare il demone `lwresd` (lightweight resolver daemon) nei vostri client. Questo demone è un server dei nomi caching-only davvero efficiente, che riconosce i nuovi record A6 e DNAME utilizzati con IPv6. Per maggiori informazioni, consultate la pagina man di `lwresd`.

16.5. Errori comuni da evitare

Spesso i principianti commettono degli errori quando modificano i file di configurazione di BIND o incontrano molte difficoltà nell'utilizzare `named`. Assicuratevi quindi di evitare i seguenti problemi:

- *Quando modificate un file della zona, assicuratevi di incrementare il numero seriale.*
Se non incrementate il numero seriale, il vostro server dei nomi master potrà disporre delle nuove informazioni, ma i server slave non riceveranno notifiche di cambiamenti e non aggiorneranno quindi i propri dati relativi a quella zona.
- *Ricordatevi di usare in modo corretto le parentesi graffe e i punti e virgola nel file `/etc/named.conf`*
Se omettete un punto e virgola oppure una parentesi, `named` non può essere avviato.
- *Non dimenticatevi di mettere i punti (.) nei file della zona dopo tutti gli FQDN e di ometterli nei nomi host.*
Il punto indica che il nome assegnato è completo. Se manca il punto, `named` completerà questo nome con quello della zona o con il valore di `$ORIGIN`.
- *Se il vostro firewall blocca le connessioni tra il demone `named` e altri server dei nomi, potrebbe essere necessario modificare il file di configurazione.*

Per default, la versione 9 di BIND utilizza infatti porte casuali superiori alla 1024 per chiedere ad altri server di risolvere i nomi. Alcuni firewall, tuttavia, presuppongono che i server dei nomi comunichino tra loro utilizzando la porta 53. Quindi per forzare questo comportamento, inserite la seguente riga nell'istruzione `options` di `/etc/named.conf`:

```
query-source address * port 53;
```

16.6. Risorse aggiuntive

Le seguenti fonti potranno fornirvi ulteriori informazioni relative all'uso di BIND.

16.6.1. Documentazione installata

- BIND dispone di documentazione installata molto esauriente su diversi argomenti, ciascuno dei quali è contenuto in una cartella propria:
 - `/usr/share/doc/bind-<numero-versione>` — contiene un file README con un elenco delle caratteristiche più recenti.

- `/usr/share/doc/bind-<numero-versione>/arm` — contiene il *BIND 9 Administrator Reference Manual* in formato HTML o SGML. Elenca nel dettaglio i requisiti di risorse di BIND, illustra come configurare diversi tipi di server, esegue il bilanciamento del carico e spiega altri argomenti avanzati. Questo manuale è il punto di partenza, soprattutto per i nuovi utenti di BIND.
 - `/usr/share/doc/bind-<numero-versione>/draft` — contiene documenti tecnici di vario genere su problemi correlati al servizio DNS e relativi metodi per risolverli.
 - `/usr/share/doc/bind-<numero-versione>/misc` — contiene documenti ideati per trattare tematiche complesse. Gli utenti della versione 8 dovrebbero consultare il documento *migration* per le modifiche da eseguire prima di migrare alla versione 9. Il file `options` elenca tutte le opzioni implementate in BIND 9 e utilizzate nel file `/etc/named.conf`.
 - `/usr/share/doc/bind-<numero-versione>/rfc` — in questa directory si trovano tutti i documenti RFC relativi a BIND.
- Sono molto utili anche le seguenti pagine man:
- `named` — tratta diversi argomenti utili per il controllo del demone di BIND, come l'uso di un file di configurazione alternativo e l'esecuzione del demone su un diverso numero di porta o come utente diverso.
 - `rndc` — illustra le varie opzioni disponibili con il comando `rndc` per controllare un server dei nomi BIND.

16.6.2. Siti Web utili

- <http://www.isc.org/products/BIND> — home page del progetto BIND. In questo sito potete trovare le informazioni relative alle versioni attuali e scaricare una versione PDF del *BIND 9 Administrator Reference Manual*.
- <http://www.redhat.com/mirrors/LDP/HOWTO/DNS-HOWTO.html> — illustra l'uso di BIND come server dei nomi "caching" oppure la configurazione dei vari file della zona che fungono da server dei nomi primari per un dominio.

16.6.3. Libri correlati

- *DNS and BIND* di Paul Albitz e Cricket Liu; edito da O'Reilly & Associates — un famoso libro di riferimento che illustra le opzioni di configurazione di BIND, comuni e non, e suggerisce strategie per un server DNS sicuro.
- *The Concise Guide to DNS and BIND* di Nicolai Langfeldt; edito da Que — approfondisce la connessione tra servizi di rete multipli e BIND, concentrandosi in modo particolare sugli argomenti tecnici "task-oriented".

Network File System (NFS)

NFS (Network File System) consente di montare delle partizioni su un sistema remoto e utilizzarle come se fossero filesystem locali. In questo modo i file possono essere organizzati all'interno di una posizione centrale, garantendo agli utenti autorizzati la possibilità di accedervi costantemente.

Al momento sono disponibili due versioni di NFS. La versione 2 (NFSv2), che esiste già da alcuni anni, è ampiamente supportata da diversi sistemi operativi. La versione 3 (NFSv3) offre alcune opzioni aggiuntive, come un file handle di lunghezza variabile e report di errore più completi. Red Hat Linux ha il supporto per entrambi le versioni (NFSv2 e NFSv3) e utilizza NFSv3 per default in caso di connessione con un server che supporta tale versione.

Questo capitolo si concentra sulla versione 2 di NFS, ma molti degli argomenti presentati si adattano anche alla versione 3. Inoltre, vengono presentati solo i concetti fondamentali relativi a NFS, con l'aggiunta di qualche informazione supplementare; pertanto, se desiderate delle istruzioni specifiche circa la configurazione e il funzionamento di NFS su macchine client o server, dovrete consultare il capitolo *Network File System (NFS)* nella *Official Red Hat Linux Customization Guide*.

17.1. Metodologia

Per consentire la condivisione dei file NFS, Linux utilizza il supporto a livello del kernel combinato con una serie di processi demoni in continua esecuzione; il supporto NFS deve essere abilitato nel kernel di Linux per poter funzionare. Per instradare le richieste tra client e server, NFS utilizza delle chiamate *RPC (Remote Procedure Call)*; questo significa che il servizio `portmap` deve essere abilitato e attivato con `runlevel` corretti perché la comunicazione possa avere luogo. Lavorando con `portmap`, vari altri processi permettono l'avvio di una connessione NFS e ne garantiscono la corretta esecuzione:

- `rpc.mountd` — il processo in esecuzione che riceve la richiesta di montaggio da un client NFS ed esegue un controllo per vedere se corrisponde con un filesystem correntemente esportato.
- `rpc.nfsd` — il processo che implementa la parte del servizio NFS relativa all'utente. Funziona con il kernel di Linux per soddisfare le richieste dinamiche dei client NFS, come per esempio l'aggiunta di thread del server.
- `rpc.lockd` — un demone non più necessario per i kernel moderni. Attualmente, il blocco dei file NFS avviene per mezzo del kernel. Viene incluso nel pacchetto `nfs-utils` ad appannaggio degli utenti in possesso di kernel di vecchia generazione che non supportano questa funzionalità per default.
- `rpc.statd` — implementa il protocollo RPC *NSM (Network Status Monitor)*. Fornisce una notifica di riavvio quando un server NFS viene riavviato in seguito a una chiusura non regolare.
- `rpc.rquotad` — un server RPC che fornisce informazioni relative alla quota utente per utenti remoti.

Non tutti questi programmi sono necessari per il servizio NFS. Gli unici servizi che bisogna abilitare sono `rpc.mountd`, `rpc.nfsd` e `portmap`. Gli altri demoni forniscono funzionalità aggiuntive e dovrebbero essere utilizzati soltanto qualora il server lo richiedesse.

La versione 2 di NFS utilizza il protocollo *UDP (User Datagram Protocol)* per instaurare una connessione di tipo *stateless* tra il client e il server. NFSv3 è in grado di utilizzare UDP o TCP su connessione IP. La connessione *stateless* con UDP riduce al minimo il traffico di rete, poiché il server NFS invia un cookie al client dopo che è stato autorizzato ad accedere al volume condiviso. Questo cookie, un valore casuale memorizzato dalla parte del server, viene trasmesso con le richieste RPC inviate dal client. Il server NFS può essere riavviato senza coinvolgere i client e il cookie rimane intatto.

NFS effettua l'autenticazione solo quando il client cerca di eseguire un montaggio su un filesystem remoto. Per limitare l'accesso, il server NFS attiva i wrappers TCP, che leggono i file `/etc/hosts.allow` e `/etc/hosts.deny` per stabilire se concedere o negare a un determinato client l'accesso al server NFS. Per maggiori informazioni su come configurare i controlli di accesso con wrapper TCP, consultate il Capitolo 8.

Una volta che il client ha passato il wrapper TCP, il server NFS consulta il suo file di configurazione, `/etc/exports`, per determinare se il client possiede i privilegi minimi necessari per montare i filesystem esportati. Dopo aver consentito l'accesso, tutte le operazioni di file e directory vengono inviate al server mediante chiamate di procedura remota RPC.



Avvertenza

I privilegi di montaggio vengono concessi a un client, non a un utente. Se assicurate a un client l'accesso a un filesystem esportato, qualsiasi utente che userà quella macchina avrà accesso ai dati.

Nel configurare il file `/etc/exports`, fate molta attenzione a concedere permessi di lettura/scrittura (`rw`) su un host remoto.

17.1.1. NFS e portmap

NFS si serve di chiamate di procedura remota (RPC) per poter funzionare; il comando `portmap` è necessario per mappare le richieste RPC ai servizi corretti. I processi RPC notificano al `portmap` quando si avviano, rivelando il numero di porta che stanno monitorando e quali numeri di programmi RPC che sono pronti a servire. Il sistema client poi contatta `portmap` sul server con un particolare numero di programma RPC. A questo punto, `portmap` reindirizza il client al numero di porta corretto perché possa comunicare con il servizio desiderato.

Dal momento che i servizi basati su RPC si affidano al `portmap` per effettuare tutte le connessioni con le richieste in ingresso dei client, è chiaro che `portmap` deve essere disponibile prima che i servizi di cui sopra siano avviati. Se, per qualche motivo, il servizio `portmap` si arresta in modo inaspettato, riavviate `portmap` insieme a tutti i servizi che erano in esecuzione quando l'avevate lanciato.

Il servizio `portmap` può essere usato con i file di accesso al portmapper (`/etc/hosts.allow` e `/etc/hosts.deny`) per controllare quali sistemi remoti possono utilizzare i servizi RPC sulla vostra macchina. Per maggiori informazioni consultate il Capitolo 8. Le regole di controllo dell'accesso al `portmap` riguardano tutti i servizi RPC, ma se lo desiderate, potete specificare a quale demone RPC di NFS applicare una determinata regola. Le pagine man relative a `rpc.mountd` e `rpc.statd` contengono informazioni sulla sintassi precisa di queste regole.

17.1.1.1. Stato del portmap

Poiché `portmap` serve per coordinare i servizi RPC con i numeri di porta utilizzati per comunicare con loro, può essere utile farsi un'idea dei servizi RPC che attualmente usano `portmap` nelle operazioni di risoluzione dei problemi. Il comando `rpcinfo` mostra ogni singolo servizio RPC con accanto il suo numero di porta, il numero di programma RPC, la versione e il tipo di protocollo IP (TCP o UDP).

Il comando `rpcinfo-p` vi può essere utile se volete accertarvi che i servizi RPC di NFS corretti siano abilitati per il `portmap`:

```

program vers proto  port
100000    2    tcp    111  portmapper
100000    2    udp    111  portmapper
100024    1    udp    1024 status
100024    1    tcp    1024 status
100011    1    udp    819  rquotad
100011    2    udp    819  rquotad

```

```

100005  1  udp  1027  mountd
100005  1  tcp  1106  mountd
100005  2  udp  1027  mountd
100005  2  tcp  1106  mountd
100005  3  udp  1027  mountd
100005  3  tcp  1106  mountd
100003  2  udp  2049  nfs
100003  3  udp  2049  nfs
100021  1  udp  1028  nlockmgr
100021  3  udp  1028  nlockmgr
100021  4  udp  1028  nlockmgr

```

L'opzione `-p` verifica il corretto funzionamento del portmapper sull'host specificato o, se nessun host specifico è presente nell'elenco, opera di default sull'host locale. Nella pagina man di `rpcinfo` troverete le altre opzioni disponibili.

Osservando l'output mostrato sopra, si nota che vi sono vari servizi di NFS in esecuzione. Se uno dei servizi non si avvia in modo corretto, `portmap` non sarà in grado di mappare alla porta corretta le richieste RPC dei client relative a quel servizio. In molti casi, è opportuno riavviare NFS come root (`service nfs restart`), per consentire a quel servizio di registrarsi correttamente in `portmap` ed entrare in funzione.

17.2. File di configurazione del server NFS

Configurare un sistema in modo che possa condividere file e directory tramite NFS è molto semplice. Tutti i filesystem esportati verso utenti remoti via NFS e, i relativi diritti di accesso per quei filesystem, si trovano nel file `/etc/exports`. Questo file viene letto dal comando `exportfs` per fornire a `rpc.mountd` e `rpc.nfsd` le informazioni necessarie per consentire il montaggio remoto di un filesystem da parte di un host autorizzato.

Il comando `exportfs` vi permette di selezionare le directory da esportare o non esportare senza dover riavviare i vari servizi di NFS. Quando `exportfs` riceve le opzioni corrette, i filesystem da esportare vengono salvati in `/var/lib/nfs/xtab`. Poiché `rpc.mountd` si rivolge al file `xtab` nel decidere i privilegi di accesso a un filesystem, le modifiche all'elenco dei filesystem esportati hanno effetto immediato.

Sono disponibili varie opzioni per l'utilizzo del comando `exportfs`:

- `-r` — Fa sì che tutte le directory elencate in `/etc/exports` vengano esportate mediante la creazione di un nuovo elenco `export` in `/etc/lib/nfs/xtab`. Questa opzione aggiorna in modo effettivo l'elenco delle esportazioni in base alle modifiche apportate a `/etc/exports`.
- `-a` — Seleziona quali directory esportare e quali no, in base alle altre opzioni scelte per `exportfs`.
- `opzioni -o` — Permette all'utente di specificare delle directory da esportare non presenti nell'elenco di `/etc/exports`. Queste condivisioni di filesystem aggiuntive devono essere salvate nello stesso modo in cui sono state specificate in `/etc/exports`. Questa opzione viene utilizzata per testare un filesystem esportato prima di aggiungerlo in modo permanente all'elenco dei filesystem da esportare.
- `-i` — Indica a `exportfs` di ignorare `/etc/exports`; in questo caso, solo le opzioni fornite dalla linea di comando vengono usate per definire i filesystem esportati.
- `-u` — Impedisce il montaggio di determinate directory da parte di utenti remoti. Il comando `exportfs -ua` sospende realmente la condivisione dei file di NFS, mentre mantiene attivi i vari demoni. Per riprendere la condivisione in NFS, digitate `exportfs -r`.

- `-v` — Operazioni complesse: i filesystem da esportare o da non esportare vengono visualizzati molto più nel dettaglio quando viene eseguito il comando `exportfs`.

Se non viene fornita alcuna opzione al comando `exportfs`, visualizza un elenco degli attuali filesystem esportati.

Potete leggere le modifiche apportate a `/etc/exports` anche ricaricando il servizio NFS tramite il comando `service nfs reload`. In questo modo i demoni di NFS rimangono attivi durante la riesportazione del file `/etc/exports`.

17.2.1. `/etc/exports`

Il file `/etc/exports` è il file standard per controllare verso quali host vengono esportati determinati filesystem e per specificare particolari opzioni adibite a controllare tutto quanto. Le linee vuote vengono ignorate, i commenti si possono fare usando il `#` e le linee lunghe possono essere spezzate con il backslash (`\`). Ogni filesystem esportato dovrebbe avere la propria linea. Gli elenchi degli host autorizzati posti dopo un filesystem esportato devono essere separati tra loro da uno spazio. Le opzioni relative a ciascun host devono essere poste tra parentesi direttamente (ovvero senza spazi di separazione) dopo l'identificatore dell'host.

Nella sua forma più semplice, `/etc/exports` ha bisogno di conoscere soltanto la directory da esportare e gli host autorizzati a utilizzarla:

```
/some/directory bob.domain.com
/another/exported/directory 192.168.0.3
```

Dopo aver riesportato `/etc/exports` con il comando `/sbin/service nfs reload`, l'host `bob.domain.com` è in grado di montare una certa directory (`/some/directory`), e `192.168.0.3` potrà montare un'altra delle directory esportate (`/another/exported/directory`). Poiché in questo esempio non è stata specificata alcuna opzione, avranno effetto alcune preferenze predefinite di NFS:

- `ro` — Solo lettura. Gli host che montano questo filesystem non saranno in grado di modificarlo. Per consentire agli host di apportare delle modifiche al filesystem, dovete specificare l'opzione `rw` (lettura/scrittura).
- `async` — Consente al server di salvare i dati sul disco quando lo ritiene opportuno. Questo non ha grande importanza se l'host sta accedendo ai dati in modalità di sola lettura; se però sta apportando delle modifiche a un filesystem con opzione lettura/scrittura, in caso di crash del server i dati potrebbero andare perduti. Specificando l'opzione `sync`, tutte le modifiche apportate ai file vengono salvate sul disco prima che sia completata la richiesta di scrittura da parte del client. L'attivazione di questa opzione può ridurre le prestazioni.
- `wdelay` — Consente al server NFS di posticipare un salvataggio su disco nel caso sospetti l'arrivo imminente di un'altra richiesta di scrittura. Questo può accrescere le prestazioni riducendo il numero di volte in cui comandi di scrittura diversi devono accedere al disco. Utilizzate `no_wdelay` per disattivare questa opzione, che funziona solo se state utilizzando l'opzione `sync`.
- `root_squash` — Fa in modo che tutti gli accessi dei client al filesystem esportato, effettuati da utenti root sulle macchine client, avvengano come UID "nobody". Questo in effetti declassa il potere dell'utente root remoto a quello del più semplice utente locale, impedendo agli utente root remoti di comportarsi come utenti root sul sistema locale. In alternativa, l'opzione `no_root_squash` disabilita questa funzione. Per applicare l'opzione di squash a tutti gli utenti remoti, compreso l'utente root, dovete usare l'opzione `all_squash`. Per specificare gli ID utente e di gruppo da utilizzare per gli utenti remoti da un particolare host, servitevi rispettivamente delle opzioni `anonuid` e `anongid`. In questo modo, potete creare uno speciale account utente per consentire agli utenti remoti di NFS di condividere e specificare UID e GID (`anonuid=<valore-uid>`, `anongid=<valore-gid>`), dove il `<valore-uid>` rappresenta il numero dell'ID utente e il `<valore-gid>` rappresenta il numero dell'ID di gruppo.

Per sovrascrivere questi parametri di default, dovete specificare delle opzioni che li sostituiscano. Per esempio, se non specificate `rw`, l'esportazione viene condivisa soltanto in modalità di sola lettura. È necessario sovrascrivere in modo esplicito ogni singolo parametro di default per ciascuno dei filesystem esportati. Inoltre, sono disponibili altre opzioni in caso non sia presente alcun valore di default. Per esempio la capacità di disabilitare i controlli sui sottoalberi, il permesso di accedere da porte non sicure e di bloccare file in modo non sicuro (procedura necessaria per certe implementazioni dei client NFS meno recenti). Per maggiori dettagli su queste opzioni meno usate e diffuse, consultate la pagina man relativa alle esportazioni `exports`.

Quando si specificano i nomi host, potete utilizzare i metodi descritti di seguito:

- *host singolo* — viene specificato un particolare host con un nome di dominio completamente qualificato, un hostname o un indirizzo IP.
- *caratteri jolly* — un asterisco (*) o un punto di domanda (?) vengono utilizzati per prendere in considerazione un raggruppamento di nomi di dominio completamente qualificati o di indirizzi IP oppure quelli che corrispondono a una determinata stringa di lettere.

Tuttavia, fatte attenzione con l'utilizzo di caratteri jolly nei nomi di dominio completamente qualificati, poiché tendono a essere più precisi di quanto si pensi. Per esempio, l'uso del carattere `*.domain.com` consentirà al dominio `sales.domain.com` di accedere al filesystem esportato, mentre non lo permetterà a `bob.sales.domain.com`. Per abilitare entrambe queste possibilità, e includere anche `sam.corp.domain.com`, dovete usare `*.domain.com *.*.domain.com`.

- *reti IP* — consente la corrispondenza per gli host sulla base dei loro indirizzi IP all'interno di una rete più ampia. Per esempio, `192.168.0.0/28` permette ai primi 16 indirizzi IP, dal `192.168.0.0` al `192.168.0.15`, di accedere al filesystem esportato, mentre non lo consente all'indirizzo `192.168.0.16` e seguenti.
- *netgroup* — permette l'utilizzo di un nome di gruppo di rete NIS, scritto `@<nome-gruppo>`. In questo modo, il server NIS diventa effettivamente responsabile del controllo degli accessi a questo filesystem esportato; si possono aggiungere e rimuovere utenti da un gruppo NIS senza intaccare `/etc/exports`.

Avvertenza

Il modo in cui viene formattato il file `/etc/exports` è molto importante, soprattutto per quanto riguarda l'uso degli spazi. Ricordatevi sempre di separare con uno spazio i filesystem esportati dagli host e gli host tra di loro. Non ci devono però essere altri spazi nel file, a parte quelli eventualmente usati nelle linee di commento.

Per esempio, le due linee seguenti non hanno lo stesso significato:

```
/home bob.domain.com(rw)
/home bob.domain.com (rw)
```

La prima linea consente ai soli utenti di `bob.domain.com` di accedere alla directory `/home` in modalità lettura/scrittura. La seconda linea permette agli utenti di `bob.domain.com` di montare la directory in sola lettura (configurazione di default), mentre tutto il resto del mondo può montarla in modalità lettura/scrittura. Dunque prestate attenzione a come inserite gli spazi all'interno di `/etc/exports`.

17.3. File di configurazione del client NFS

Tutte le condivisioni NFS rese disponibili da un server possono essere montate mediante vari metodi, tra cui naturalmente quello manuale, tramite il comando `mount`, che consente di acquisire il filesystem esportato su un particolare mount point. Questo, però obbliga l'utente `root` a digitare il

comando `mount` a ogni riavvio del sistema. L'utente `root` deve inoltre ricordarsi di somontare il filesystem alla chiusura della macchina. Due metodi di configurazione dei mount NFS sono la modifica di `/etc/fstab` o l'utilizzo del servizio `autofs`.

17.3.1. /etc/fstab

Inserendo una linea formattata correttamente nel file `/etc/fstab` si ottiene lo stesso risultato prodotto dal montaggio manuale del filesystem esportato. Il file `/etc/fstab` viene letto dallo script `/etc/rc.d/init.d/netfs` all'avvio del sistema. I mount di filesystem corretti, tra cui NFS, vengono sistemati al loro posto.

Un esempio di linea `/etc/fstab` per montare un export di NFS ha il seguente aspetto:

```
<server>:</path/of/dir> </local/mnt/point> nfs <options> 0 0
```

La parte `<server-host>` si riferisce all'hostname, indirizzo IP o al nome di dominio completamente qualificato del server che sta esportando il filesystem. La parte `</path/to/shared/directory>` indica al server quale esportazione montare. La parte `</local/mount/point>` specifica in quale punto del filesystem locale montare la directory esportata. Questo mount point deve esistere già prima che il file `/etc/fstab` venga letto, altrimenti l'operazione di montaggio non riesce. L'opzione `nfs` specifica il tipo di filesystem che viene montato.

L'area `<options>` (area delle opzioni) determina il modo in cui va montato il filesystem. Per esempio, se nell'area delle opzioni è specificato il parametro `rw,suid` per un particolare montaggio, il filesystem esportato sarà montato in modalità lettura/scrittura e verranno utilizzate le ID utente e di gruppo impostate dal server. Si noti che qui non vanno utilizzate le parentesi. Per conoscere altre opzioni relative al montaggio, consultate la Sezione 17.3.3.

17.3.2. autofs

Uno degli effetti "collaterali" nell'utilizzo del file `/etc/fstab` è rappresentato dal fatto che, a prescindere da quanto utilizzate quel filesystem montato, il vostro sistema deve comunque dedicare parte delle sue risorse per mantenerlo in funzione. Se sul sistema sono montati solo un paio di filesystem, il problema non è così grave; se però il sistema deve mantenere contemporaneamente in funzione decine di filesystem montati, le prestazioni generali possono risentirne. In alternativa al file `/etc/fstab` potete usare `automount`, un'utility basata su kernel che monta e smonta automaticamente i filesystem NFS, riducendo il consumo di risorse.

Lo script `autofs`, contenuto in `/etc/rc.d/init.d`, viene utilizzato per controllare `automount` attraverso il file di configurazione contenuto in `/etc/auto.master`. Mentre `automount` può essere specificato sulla linea di comando, è più opportuno specificare mount point, hostname, directory esportata e opzioni all'interno di un gruppo di file, invece di digitare il tutto manualmente. Eseguendo `autofs` come un servizio che si avvia e si arresta su runlevel specificati, si possono implementare automaticamente i parametri di configurazione contenuti nei vari file. Per poter usare `autofs`, dovete avere installato sul vostro sistema il relativo pacchetto RPM.

I file di configurazione di `autofs` sono organizzati secondo rapporti genitore-figlio. Un file di configurazione principale (`/etc/auto.master`) riferisce ai mount point sul vostro sistema che sono collegati a un particolare *tipo di mappa*, che assume la forma di altri file di configurazione, programmi, mappe NIS e altri metodi di montaggio meno comuni. Il file `auto.master` contiene delle linee che si rivolgono a ciascuno di questi mount point, organizzate come segue:

```
<mount-point> <map-type>
```

La parte `<mount-point>` indica il punto del vostro filesystem locale in cui montare il dispositivo o il filesystem esportato. La parte relativa al tipo di mappa (`<map-type>`) si riferisce al modo in cui

verrà montato il mount point. Il metodo più comune per il montaggio automatico di export di NFS è quello di usare un file come tipo di mappa per un particolare mount point. Il file mappa, di solito chiamato `auto.<mount-point>`, dove per `<mount-point>` si intende il mount point designato in `auto.master`, contiene linee simili a questa:

```
<directory> <mount-options> <host>:<exported-filesystem>
```

La parte `<directory>` si riferisce alla directory contenuta nel mount point in cui il filesystem esportato dovrebbe essere montato. Proprio come un comando `mount` standard, l'host che esporta il filesystem e lo stesso filesystem esportato sono richiesti nella sezione `<host>:<exported-filesystem>`. Per specificare opzioni particolari da utilizzare nel montaggio di un filesystem esportato, inseritele nella sezione `<mount-options>`, separate da virgole. Per i montaggi di NFS che utilizzano `autoofs`, dovete assolutamente inserire `-fstype=nfs` nella sezione `<mount-options>`.

I file di configurazione di `autoofs`, che possono essere utilizzati per svariati montaggi su molti tipi di dispositivi e filesystem, sono particolarmente utili nella creazione di montaggi NFS. Per esempio, alcune aziende salvano la directory `/home` di un utente su un server centrale tramite una condivisione di NFS. Poi, configurano il file `auto.master` su ciascuna delle workstation in modo che si rivolgano a un file `auto.home` contenente le specifiche su come montare la directory `/home` via NFS. In questo modo, l'utente ha accesso ai dati personali e ai file di configurazione contenuti nella directory `/home` collegandosi da un punto qualsiasi della rete interna. In un caso come questo, il file `auto.master` ha il seguente aspetto:

```
/home /etc/auto.home
```

In questo modo, il mount point di `/home` sul sistema locale è impostato per essere configurato dal file `/etc/auto.home`, che ha un aspetto simile al seguente:

```
* -fstype=nfs,soft,intr,rsize=8192,wsiz=8192,nosuid server.domain.com:/home/&
```

Secondo questa linea, qualsiasi directory cui un utente cerchi di accedere sotto la directory `/home` locale (per via dell'asterisco) dovrebbe risultare in un montaggio NFS sul sistema `server.domain.com` all'interno del suo filesystem `/home` esportato. Le opzioni di montaggio specificano che ogni montaggio NFS nella directory `/home` deve servirsi di una particolare serie di impostazioni. Per maggiori informazioni sulle opzioni di montaggio, tra cui quelle incluse in questo esempio, consultate la Sezione 17.3.3.

17.3.3. Comuni opzioni di montaggio NFS

Oltre al montaggio di un filesystem via NFS su un host remoto, si possono specificare numerose altre opzioni al momento del montaggio, che possono facilitare tale operazione. Queste opzioni possono essere utilizzate con comandi manuali `mount`, impostazioni di `/etc/fstab`, `autoofs` e altri metodi di montaggio.

Le opzioni che seguono sono le più diffuse per i montaggi via NFS:

- `hard` o `soft` — specificano se il programma che utilizza un file via connessione NFS deve arrestarsi e attendere (`hard`) che il server torni in linea nel caso in cui l'host che sta servendo il filesystem esportato non sia disponibile o se deve invece inviare un report di errore (`soft`).

Se specificate `hard`, non sarete in grado di terminare il processo in attesa che la comunicazione NFS riprenda, a meno che non specificiate anche l'opzione `intr`.

Se indicate l'opzione `soft`, potete impostare un'opzione aggiuntiva del tipo `timeo=<valore>` per specificare (nella sezione `<value>`) il numero di secondi che devono trascorrere prima che venga effettuato il report di errore.

- `intr` — fa sì che le richieste NFS vengano interrotte qualora si verifichi una caduta del server o nel caso non fosse possibile raggiungerlo.
- `nolock` — a volte è richiesto nelle connessioni con server NFS di vecchia generazione. Per richiedere il blocco, usate l'opzione `lock`.
- `noexec` — non permette l'esecuzione di binari sul filesystem montato. È utile se il vostro sistema Red Hat Linux sta montando via NFS un filesystem non di tipo Linux contenente binari che non possono essere eseguiti sulla vostra macchina.
- `nosuid` — rende ineffettivi bit SUID (Set User Identifier) o SGID (Set Group Identifier).
- `rsize=8192` e `wsize=8192` — possono aumentare la velocità di comunicazione NFS per operazioni di lettura (`rsize`) e scrittura (`wsize`) impostando una dimensione maggiore per i blocchi di dati (in termini di byte), da trasferire in una volta. Fate attenzione quando modificate questi valori; alcuni kernel di Linux e schede di rete di vecchia generazione non funzionano molto bene con blocchi di dimensioni maggiori.
- `nfsvers=2` o `nfsvers=3` — specificano quale versione del protocollo NFS utilizzare.

Nella pagina man di `mount` sono elencate molte altre opzioni, tra cui quelle relative al montaggio di filesystem non NFS.

17.4. Sicurezza e NFS

NFS è molto utile per condividere in modo trasparente, interi filesystem con un gran numero di host conosciuti. Molti utenti che accedono a dei file via NFS possono non essere a conoscenza del fatto che il filesystem che stanno usando non è il filesystem locale per il loro sistema. Comunque, al di là della facilità d'uso, esistono numerosi potenziali problemi di sicurezza.

Dovete tenere in considerazione i seguenti punti quando esportate filesystem NFS su un server o li montate su un client, così potrete ridurre al minimo i rischi legati alla sicurezza di NFS e sarete in grado di proteggerne meglio i vostri dati.

17.4.1. Accesso Host

NFS controlla chi può montare un filesystem esportato in base all'host che effettua la richiesta di montaggio, non in base all'utente che utilizza il filesystem. Occorre fornire agli host dei diritti specifici perché possano montare il filesystem esportato. Il controllo degli accessi non è possibile per gli utenti, a differenza dei permessi di file e directory. In altre parole, quando esportate un filesystem via NFS verso un host remoto, non solo vi state "fidando" dell'host cui permettete di montare il filesystem, ma state anche permettendo a tutti gli utenti che hanno il permesso di accedere a quell'host di utilizzarlo. È comunque possibile tenere sotto controllo questo rischio, per esempio esclusivamente montaggi in modalità di sola lettura e limitando il potere degli utenti e dei gruppi tramite l'opzione `squash`. Tenete presente, tuttavia, che queste soluzioni possono avere delle ripercussioni sull'utilizzo previsto per il montaggio.

Inoltre, se un aggressore assume il controllo del server DNS utilizzato dal sistema che esporta il filesystem NFS, il sistema associato a un particolare hostname o a un nome di dominio completamente qualificato può essere indirizzato verso una macchina non autorizzata. A questo punto, tale macchina non autorizzata è il sistema che ha il permesso di montare la condivisione NFS, dal momento che non vengono scambiate informazioni di sorta relative a nome utente e password mirate ad aumentare la sicurezza del montaggio. Gli stessi rischi valgono anche per i server NIS compromessi, se i gruppi di rete NIS vengono utilizzati per consentire a determinati host di montare una condivisione NFS. Invece, usando degli indirizzi IP in `/etc/exports`, questo attacchi di questo genere sono molto più difficili da effettuare.

I caratteri jolly vanno usati con cautela quando si sta concedendo l'accesso a una condivisione NFS. Il loro raggio di azione potrebbe raggiungere anche sistemi di cui non conoscete l'esistenza e cui non dovrebbe essere dato il permesso di montare il filesystem.

17.4.2. Permessi dei file

Una volta che il filesystem NFS è stato montato in modalità lettura/scrittura da un host remoto, la protezione per ciascuno dei file condivisi coinvolge i relativi permessi e ID utente e di gruppo. Se due utenti che condividono lo stesso valore di ID utente montano lo stesso filesystem NFS, l'uno sarà in grado di modificare i file dell'altro e viceversa. Inoltre, chiunque si colleghi come root sul sistema client può utilizzare il comando `su -` per assicurarsi i permessi per accedere a determinati file attraverso la condivisione NFS.

Il modo di operare di default, quando si esporta un filesystem via NFS è il *root squashing* ("schiacciamento di root"), che imposta l'ID utente di chiunque acceda alla condivisione NFS come utente di root sulla propria macchina locale su un valore di UID anonimo, o "nobody". È altamente sconsigliabile disabilitare questa opzione, a meno che la presenza di vari utenti con accessi di root al vostro server non rappresenti una preoccupazione.

Se accordate agli utenti permessi di sola lettura per i file cui accedono mediante la condivisione NFS, dovrete usare anche l'opzione `all_squash`, che attribuisce un UID anonimo a chiunque acceda al vostro filesystem esportato.

17.5. Risorse aggiuntive

L'amministrazione di un server NFS può rappresentare una sorta di sfida. Per esportare filesystem NFS o montarli come client, sono disponibili numerose opzioni, alcune delle quali non sono state menzionate in questo capitolo. Per maggiori dettagli, dunque, consultate queste ulteriori fonti di informazione.

17.5.1. Documentazione installata

- `/usr/share/doc/nfs-utils-<numero-versione>` — tratta dell'implementazione di NFS in Linux e offre una panoramica delle varie configurazioni NFS, descrivendone gli effetti sulle prestazioni del trasferimento di file.
- Le seguenti pagine man sono molto utili:
 - `mount` — offre una descrizione esauriente delle varie opzioni di montaggio relative alla configurazione di server e client NFS.
 - `fstab` — fornisce dettagli in merito al formato del file `/etc/fstab`, utilizzato per montare filesystem all'avvio del sistema.
 - `nfs` — fornisce dettagli in merito alle opzioni di montaggio ed esportazione specifiche per NFS.
 - `exports` — mostra le opzioni comunemente utilizzate nel file `/etc/exports` per l'esportazione di filesystem NFS.

17.5.2. Libri correlati

- *Managing NFS and NIS* di Hal Stern, Mike Eisler e Ricardo Labiaga; O'Reilly & Associates — un'eccellente guida di riferimento circa le numerose opzioni di montaggio ed esportazione disponibili per NFS.
- *NFS Illustrated* di Brent Callaghan; Addison-Wesley Publishing Company — mette a confronto NFS con altri filesystem di rete e spiega nel dettaglio come avviene la comunicazione NFS.

LDAP (Lightweight Directory Access Protocol)

Il protocollo *LDAP (Lightweight Directory Access Protocol)* è costituito da una serie di protocolli utilizzati per accedere ai servizi sullo standard *X.500* per la condivisione di directory, ma è meno complesso e non richiede grandi risorse. Per tale ragione, LDAP viene spesso chiamato *X.500 Light*.

Come *X.500*, LDAP organizza le informazioni in ordine gerarchico servendosi di directory che sono in grado di immagazzinare numerose informazioni e possono anche essere utilizzate a mo' di NIS (Network Information Service), permettendo a chiunque di accedere al loro account da qualsiasi macchina presente sulla rete LDAP abilitata.

In molti casi, tuttavia, LDAP viene utilizzato semplicemente come rubrica virtuale che permette agli utenti di accedere con facilità alle informazioni relative ai contatti di altri utenti. LDAP, però è più che una semplice guida telefonica, poiché è in grado di propagare le proprie directory ad altri server LDAP in tutto il mondo, fornendo accesso globale all'informazione. Attualmente, LDAP è comunque utilizzata per lo più da organizzazioni individuali, quali università dipartimenti governativi e aziende private.

LDAP è un sistema client-server. Il server può avvalersi di numerosi database per salvare una directory, ciascuno dei quali ottimizzato per operazioni di lettura rapide e sostanziose. Quando un'applicazione di un client LDAP si connette a un server LDAP può interrogare una directory oppure caricare informazioni al suo interno. Nel caso dell'interrogazione, il server risponde oppure, se non può fare a livello locale, può delegare il flusso dell'interrogazione a un LDAP di livello superiore che sia in grado di rispondere. Se l'applicazione client cerca di caricare informazioni all'interno di una directory LDAP, il server verifica che l'utente abbia il permesso di attuare la modifica, poi aggiunge o aggiorna le informazione.

Questo capitolo si riferisce alla configurazione e all'utilizzo di OpenLDAP 2.0, un'implementazione Open Source dei protocolli LDAPv2 e LDAPv3.

18.1. Perché utilizzare LDAP?

Il vantaggio principale nell'uso di LDAP è la possibilità di consolidare certi tipi di informazione all'interno della vostra azienda. Per esempio, invece di gestire elenchi di utenti per ogni gruppo all'interno di un'azienda, potete utilizzare LDAP come directory centrale accessibile da chiunque sulla rete. Dal momento, poi, che LDAP supporta i protocolli SSL (Secure Sockets Layer) e TLS (Transport Layer Security), è possibile proteggere i dati importanti da occhi indiscreti.

LDAP supporta anche numerosi database backend in cui è possibile immagazzinare directory. Questo accorda agli amministratori la flessibilità necessaria per attivare il database più indicato per il tipo di informazione che il server deve diffondere. Inoltre, poiché LDAP ha un'API (Application Programming Interface (API) ben definita, il numero di applicazioni abilitate all'uso di LDAP è elevato e sta ancora crescendo in termini di quantità e di qualità.

Lo svantaggio consiste nel fatto che LDAP può essere abbastanza complesso da configurare.

18.1.1. Potenziamento delle caratteristiche di OpenLDAP 2.0

OpenLDAP 2.0 comprende numerose caratteristiche importanti.

- *Supporto LDAPv3* — OpenLDAP 2.0 supporta i protocolli SASL (Simple Authentication and Security Layer), TLS e SSL ed è stato sottoposto ad altre migliori. Molte delle modifiche attuate al protocollo rispetto alla versione 2 sono state concepite per rendere LDAP più sicuro.

- *Supporto IPv6* — OpenLDAP supporta il protocollo Internet Protocol 6 di nuova generazione.
- *LDAP tramite IPC* — OpenLDAP è in grado di comunicare all'interno di una sistema utilizzando la comunicazione interprocess (IPC), che rafforza la sicurezza ovviando alla necessità di comunicare su una rete.
- *API C aggiornata* — migliora il modo in cui i programmatori possono connettersi e utilizzare l'applicazione.
- *Supporto LDIFv1* — interamente compatibile con le versione 1 di LDIF (LDAP Data Interchange Format).
- *Server LDAP stand-alone avanzato* — comprende un sistema di controllo dell'accesso aggiornato, un raggruppamento dei thread, strumenti migliori e altro ancora.

18.2. Demoni e utility OpenLDAP

Il pacchetto di librerie e tool OpenLDAP si estende ai pacchetti seguenti:

- `openldap` — contiene le librerie necessarie per eseguire le applicazioni server e client `openldap`.
- `openldap-clients` — contiene tool a linea di comando per visualizzare e modificare directory su un server LDAP.
- `openldap-server` — contiene i server e altre utility necessari a configurare ed gestire un server LDAP.

Nel pacchetto `openldap-servers` sono contenuti due server: il demone *Standalone LDAP* (`slapd`) e il demone *Standalone LDAP Update Replication* (`/usr/sbin/slurpd`).

Il demone `slapd` è il server LDAP effettivo mentre il demone `slurpd` viene utilizzato per sincronizzare le modifiche da un server LDAP ad altri server LDAP sulla rete. Il demone `slurpd` è necessario soltanto quando si ha a che fare con un server LDAP multiplo.

Per eseguire task di amministrazione, il pacchetto `openldap-server` installa le utility seguenti all'interno di `/usr/sbin`:

- `slapadd` — aggiunge a una directory LDAP voci da un file LDIF. Per esempio, `/usr/sbin/slapadd -l ldif-input` legge il file LDIF (`ldif-input`), contenente le nuove voci.
- `slapcat` — estrae voci da una directory LDAP nel formato predefinito, Berkeley DB, e li salva in un file LDIF. Per esempio, il comando `/usr/sbin/slapcat -l ldif-output` produrrà un file LDIF chiamato `ldif-output`, contenente le voci della directory LDAP.
- `slapindex` — riordina la directory `slapd` in base al contenuto attualizzato.
- `slappasswd` — genera un valore password utente criptato da utilizzare con `ldapmodify` o il valore `rootpw` nel file di configurazione di `slapd` (`/etc/openldap/slapd.conf`). Per creare la password eseguite il comando `/usr/sbin/slappasswd`.



Avvertenza

Assicuratevi di fermare `slapd` con il comando `/usr/sbin/service slapd stop` prima di utilizzare `slapadd`, `slapcat` o `slapindex`, per non compromettere l'integrità della directory LDAP.

Per maggiori informazioni su come utilizzare queste utility, consultate le relative pagine `man`.

Il pacchetto `openldap-clients` installa i tool necessarie ad aggiungere, modificare e cancellare voci in una directory LDAP all'interno di `/usr/bin/`. Tra questi tool sono compresi i seguenti:

- `ldapmodify` — modifica le entry in una directory LDAP, mediante input standard o via file.
- `ldapadd` — aggiunge le entry alla vostra directory, accettando input standard o mediante file; `ldapadd` rappresenta un collegamento a `ldapmodify -a`.
- `ldapsearch` — cerca le entry in una directory LDAP utilizzando il prompt della shell.
- `ldapdelete` — cancella le entry di una directory LDAP accettando input tramite un file o un prompt della shell.

A eccezione di `ldapsearch`, ognuna di queste utility è molto più semplice da utilizzare, poiché è sufficiente digitare il file con le modifiche da effettuare in una directory LDAP piuttosto che digitare i comandi uno dopo l'altro. Ognuna delle relative pagine man illustra il formato di questi file.

Red Hat Linux comprende un'applicazione grafica chiamata **GQ** che consente di creare e modificare le voci di una directory. Tale applicazione non fa parte del pacchetto di tool di OpenLDAP, ma è contenuta nell'rpm `gq`. Per maggiori informazioni sulle applicazioni del client LDAP, consultate la Sezione 18.2.3.

18.2.1. NSS, PAM e LDAP

Oltre ai pacchetti OpenLDAP, Red Hat Linux fornisce un pacchetto chiamato `nss_ldap` che migliora la capacità di LDAP di integrarsi con Linux e altri ambienti UNIX.

Il pacchetto `nss_ldap` fornisce i moduli seguenti:

- `/lib/libnss_ldap-<glibc-version>.so`
- `/lib/security/pam_ldap.so`

Il modulo `libnss_ldap-<glibc-version>.so` consente alle applicazioni di fare ricerche sugli utenti, i gruppi, gli host e in merito ad altre informazioni utilizzando una directory LDAP mediante l'interfaccia NSS (*Nameservice Switch*) di `glibc`. NSS può essere usata al posto di o in aggiunta al name service NIS (*Network Information Service*) o ai file lineari per l'autenticazione.

Il modulo `pam_ldap` consente alle applicazioni che supportano PAM di autenticare gli utenti utilizzando le informazioni contenute in una directory LDAP. Tali applicazioni comprendono il collegamento da console, server di posta POP e IMAP e inoltre Samba. Attivando un server LDAP sulla vostra rete, tutte queste possibilità di login possono eseguire l'autenticazione tramite una combinazione ID e password utente, semplificando molto le operazioni di amministrazione.

18.2.2. Apache HTTP Server, PHP4 e LDAP

Red Hat Linux comprende anche pacchetti contenenti moduli LDAP per Apache HTTP Server e lo script di linguaggio di parte server PHP.

Il pacchetto `php-ldap` conferisce a LDAP il supporto per lo script di linguaggio PHP4 HTML-integrato mediante il modulo `/usr/lib/php4/ldap.so`, che consente agli script PHP4 di accedere alle informazioni contenute in una directory LDAP.



Importante

Red Hat Linux 8.0 non dispongono più del pacchetto `auth_ldap`, che forniva il supporto LDAP per le versioni 1.3 e precedenti di Apache HTTP Server. Per maggiori informazioni sullo stato di questo modulo, consultate il sito Web di Apache Software Foundation all'indirizzo <http://www.apache.org/>.

18.2.3. Applicazioni client LDAP

Esistono client grafici LDAP in grado di supportare la creazione e la modifica di directory, ma non sono compresi nella confezione di Red Hat Linux 8.0. Uno di essi è **LDAP Browser/Editor** — un tool basato su Java disponibile all'indirizzo <http://www.iit.edu/~gawojar/ldap>.

La maggior degli altri client LDAP accedono alle directory in sola lettura e si possono usare per consultare le informazioni globali, ma non per alterarle. Esempi di tali applicazioni sono i browser Web che si basano su Mozilla, **Balsa** per Sendmail, **Pine**, **Evolution**, **Gnome Meeting**.

18.3. Terminologia LDAP

Una *entry* è un'unità di una directory LDAP. Ciascuna entry è identificabile grazie al suo DN (Distinguished Name) univoco.

Ciascuna entry ha degli *attributi*, ovvero informazioni a esse direttamente associate. Per esempio, se una delle entry fosse un'organizzazione, gli attributi associati potrebbero essere il numero di fax, l'indirizzo ecc. Anche le persone possono essere voci di una directory LDAP e gli attributi più comuni sono il numero di telefono e l'indirizzo email.

Alcuni attributi sono obbligatori, altri sono opzionali. Una definizione *objectclass* stabilisce quali attributi sono obbligatori e quali no per ciascuna voce. Le definizioni *objectclass* si trovano all'interno di vari file schema, situati nella directory `/etc/openldap/schema/`.

L'LDIF (*LDAP Data Interchange Format*) è un formato testo ASCII per le voci LDAP. I file che importano o esportano dati verso da e verso i server LDAP devono essere in formato LDIF. Una voce LDIF ha un aspetto simile a questo:

```
[<id>]
dn: <distinguished name>
<attrtype>: <attrvalue>
<attrtype>: <attrvalue>
<attrtype>: <attrvalue>
```

Una voce può contenere tutte le coppie `<attrtype>: <attrvalue>` necessarie. Una linea vuota indica la fine della voce.



Attenzione

Tutte le coppie `<attrtype>` e `<attrvalue>` *devono* essere definite in un file schema corrispondente.

Tutti i valori compresi all'interno di un `<` e di un `>` sono delle variabili e possono essere impostati ogni volta che viene creata una nuova voce LDAP, tranne che nel caso di `<id>`. La `<id>` è un numero determinato dall'applicazione che avete interpellato per modificare la voce.

**Nota Bene**

Raramente vi occorrerà modificare manualmente una voce LDIF. È consigliabile, piuttosto, utilizzare un'applicazione client LDAP, come quelle elencate nella la Sezione 18.2.

18.4. File di configurazione di OpenLDAP

I file di configurazione di OpenLDAP vengono installati all'interno della directory `/etc/openldap/`. Di seguito è riportato un breve elenco delle directory e dei file più importanti:

- Directory `/etc/openldap/schema/` — questa sottodirectory contiene il file schema utilizzato dal demone `slapd`. Per maggiori informazioni su questa directory, consultate la la Sezione 18.4.2.
- `/etc/openldap/ldap.conf` — il file di configurazione per tutti le applicazioni *client* utilizzate dalle librerie OpenLDAP. Tra queste (ma ce ne sono altre), Sendmail, **Pine**, **Balsa**, **Evolution** e **Gnome Meeting**.
- `/etc/openldap/slapd.conf` — Questo è il file di configurazione per il demon `slapd`. Per maggiori informazioni, consultate la la Sezione 18.4.1.

**Nota Bene**

Il pacchetto `nss_ldap` crea un file chiamato `/etc/ldap.conf`, che viene utilizzato dai moduli PAM e NSS forniti nel pacchetto `nss_ldap`. Per maggiori informazioni, consultate la la Sezione 18.7.

18.4.1. `slapd.conf`

Per poter utilizzare il server LDAP `slapd`, dovete modificarne il file di configurazione (`/etc/openldap/slapd.conf`). Bisogna agire su questo file in modo da renderlo specifico per il vostro dominio e il vostro server.

La linea `suffix` indica il dominio per il quale il server LDAP fonirà le informazioni e deve essere cambiata da:

```
suffix          "dc=your-domain,dc=com"
```

in modo che rifletta il vostro nome di dominio. Per esempio:

```
suffix          "dc=example,dc=com"
```

La voce `rootdn` è il *DN (Distinguished Name)* per un utente che non ha restrizioni da parte del controllo di accesso o dai parametri per i limiti di amministrazione impostati per lo svolgimento di operazioni sulla directory LDAP. L'utente `rootdn` può essere considerato come l'utente `root` per la directory LDAP. La linea `rootdn` deve essere cambiata come riportato di seguito:

```
rootdn          "cn=root,dc=example,dc=com"
```

Cambiate la linea `rootpw` in qualcosa di simile all'esempio riportato qui di seguito:

```
rootpw          {SSHA}vv2y+i6V6esazrIv70xSSnNAJE18bb2u
```

Nell'esempio `rootpw`, la password di root è: criptata, molto più sicura rispetto a una password di testo, che non andrebbe mai utilizzata nel file `slapd.conf`. Per creare questa stringa criptata, digitate il comando seguente:

```
slappasswd
```

Vi verrà richiesto di digitare e confermare una password. Il programma visualizza la password criptata sul terminale.



Avvertimento

Se non viene attivata la cifratura TLS, le password LDAP, compresa la direttiva `rootpw` specificata in `/etc/openldap/slapd.conf`, vengono inviate via rete come testo.

Per maggior sicurezza, la direttiva `rootpw` dovrebbe essere utilizzata solo nel caso in cui la configurazione e il popolamento iniziali della directory LDAP avvengono attraverso una rete. Una volta terminata tale operazione, è meglio commentare la direttiva `rootpw` anteponendo ad essa un cancelletto(#).



Suggerimento

Se state utilizzando il tool a linea di comando `slapadd` a livello locale per popolare la directory LDAP, non è necessario usare la direttiva `rootpw`.

18.4.2. La directory `/etc/openldap/schema/`

La directory `/etc/openldap/schema/` contiene definizioni LDAP, in precedenza situate nei file `slapd.at.conf` e `slapd.oc.conf`. Tutte le *definizioni delle sintassi di attributi* e le *definizioni objectclass* si trovano ora in file schema diversi. I vari file schema contenuti in `/etc/openldap/slapd.conf` vengono definiti mediante linee `include`, come mostrato in questo esempio:

```
include /etc/openldap/schema/core.schema
include /etc/openldap/schema/cosine.schema
include /etc/openldap/schema/inetorgperson.schema
include /etc/openldap/schema/nis.schema
include /etc/openldap/schema/rfc822-MailMember.schema
include /etc/openldap/schema/autofs.schema
include /etc/openldap/schema/kerberosobject.schema
```



Attenzione

Non modificate gli elementi schema definiti nei file schema installati da OpenLDAP.

Potete estendere il file schema utilizzato da OpenLDAP per far sì che supporti tipi di attributi e object class servendosi dei file schema predefiniti a mo' di guida. Per farlo, create un file `local.schema`

nella directory `/etc/openldap/schema`. Create un riferimento a questo nuovo schema Reference all'interno di `slapd.conf` aggiungendo la linea seguente sotto le vostre linee `schema include` predefinite:

```
"include/etc/openldap/schema/local.schema
```

Successivamente, definite i vostri nuovi tipi di attributi e gli object class all'interno del file `local.schema`. Molte organizzazioni prendono tipi di attributi e object class esistenti dai file schema installati per default e li modificano per poterli utilizzare nel file `local.schema`. Questo può essere utile per imparare la sintassi di schema riuscendo contemporaneamente a soddisfare le esigenze immediate della propria organizzazione.

L'estensione di file schema per poter soddisfare determinati requisiti specifici non rientra negli argomenti trattati da questo capitolo. Per maggiori informazioni, visitate il sito <http://www.openldap.org/doc/admin/schema.html>.

18.5. Panoramica sulla configurazione di OpenLDAP

Questa sezione fornisce una rapida panoramica su come installare e configurare una directory OpenLDAP. Per approfondimenti, visitate gli indirizzi seguenti:

- <http://www.openldap.org/doc/admin/quickstart.html> — La *Quick-Start Guide* sui siti OpenLDAP.
- <http://www.redhat.com/mirrors/LDP/HOWTO/LDAP-HOWTO.html> — Il *LDAP Linux HOWTO* tratto dal Linux Documentation Project, presente anche sul sito di Red Hat.

Di seguito sono riportate le operazioni basilari necessarie a creare un server LDAP:

1. Installate gli RPM `openldap`, `openldap-servers` e `openldap-clients`.
2. Modificate il file `/etc/openldap/slapd.conf` in modo che rimandi al vostro dominio e al vostro server LDAP. Per maggiori informazioni su come modificare questo file, consultate la Sezione 18.4.1.
3. Avviate `slapd` con il comando:


```
/sbin/service/ldap start
```

 Dopo aver configurato LDAP correttamente, potete utilizzare `chkconfig`, `ntsysv` o **redhat-config-services** per configurare LDAP in modo che venga lanciato all'avvio. Per maggiori informazioni su come configurare i servizi, consultate il capitolo intitolato *Controllo dell'accesso ai servizi* nella *Official Red Hat Linux Customization Guide*.
4. Aggiungete delle voci alla vostra directory LDAP mediante `ldapadd`.
5. Utilizzate `ldapsearch` per controllare se `slapd` sta accedendo in modo corretto alle informazioni.
6. A questo punto, la vostra directory LDAP dovrebbe funzionare correttamente e potete configurare qualsiasi applicazione che supporti LDAP in modo che possa servirsi della directory.

18.6. Aggiornamento alla versione 2.0 di OpenLDAP

In OpenLDAP versione 2.0 il formato di archiviazione su disco utilizzato dal server LDAP `slapd` è cambiato. Se state aggiornando LDAP da Red Hat Linux 7.0 o versioni precedenti, sarà necessario estrarre le directory LDAP esistenti in un file LDIF mediante il seguente comando:

```
ldbmcat -n > <ldif_file>
```

In questo comando sostituite `<ldif_file>` con il nome del file di output. Digitate quindi il seguente comando per importare il file in OpenLDAP 2.0:

```
slapadd -l <ldif_file>
```

18.7. Configurare il sistema per l'autenticazione mediante OpenLDAP

Questa sezione vi offre una rapida panoramica su come configurare un sistema Red Hat Linux in modo che possa eseguire le autenticazioni mediante OpenLDAP. Se non siete utenti esperti di OpenLDAP, probabilmente vi occorrerà una documentazione più approfondita rispetto a quella fornita in questa sezione. Per maggiori informazioni, consultate i riferimenti forniti nella la Sezione 18.8.

18.7.1. Installazione dei pacchetti LDAP necessari

Dovete anzitutto assicurarvi di aver installato i pacchetti adeguati sia sul server LDAP sia sulle macchine client. Il server LDAP necessita del pacchetto `openldap-server`.

Sulle macchine client LDAP devono essere installati i pacchetti seguenti: `openldap`, `openldap-clients` e `nss_ldap`.

18.7.2. Modifica dei file di configurazione

18.7.2.1. Modifica di `slapd.conf` sul server

Modificate il file `/etc/openldap/slapd.conf` sul server LDAP in modo che soddisfi correttamente le specifiche della vostra organizzazione. Per istruzioni su come modificare `slapd.conf`, consultate la Sezione 18.4.1.

18.7.2.2. Modifica di `/etc/ldap.conf` e `/etc/openldap/ldap.conf` sui client

In tutti i computer client `/etc/ldap.conf` e `/etc/openldap/ldap.conf` devono contenere il server appropriato e cercare informazioni di base per l'azienda.

Il modo più semplice di effettuare questa operazione è quello di eseguire `authconfig` e selezionare **Use LDAP** nella schermata **User Information Configuration**.

Potete anche modificare questi file manualmente.

18.7.2.3. Modifica di `/etc/nsswitch.conf` sui client

In tutti i computer client il file `/etc/nsswitch.conf` deve essere modificato per utilizzare LDAP.

Il modo più semplice di effettuare questa operazione è quello di eseguire `authconfig` e selezionare **Use LDAP** nella schermata **User Information Configuration**.

Se modificate il file `/etc/nsswitch.conf` manualmente, aggiungete `ldap` nei campi appropriati.

Per esempio:

```
passwd: files ldap
shadow: files ldap
group: files ldap
```

18.7.2.4. PAM e LDAP

Per fare in modo che le applicazioni che supportano PAM utilizzino LDAP per le autenticazioni, eseguite `authconfig` e selezionate **Usa LDAP Authentication** nella schermata **Authentication Configuration**. Per maggiori informazioni su come configurare PAM, consultate il Capitolo 7 e le pagine man di PAM.

18.7.3. Migrazione di informazioni di autenticazione preesistenti in formato LDAP

La directory `/usr/share/openldap/migration/` contiene una serie di shell e script Perl per la migrazione di informazioni di autenticazione in formato LDAP.



Nota Bene

Per utilizzare questi script dovete aver installato Perl sul vostro sistema.

Anzitutto, dovete modificare il file `migrate_common.ph` in modo che rifletta il vostro dominio. Occorre cambiare il valore predefinito del dominio DNS di default in qualcosa di simile a:

```
$DEFAULT_MAIL_DOMAIN = "vostra_azienda";
```

Bisogna inoltre cambiare la base predefinita all'incirca come riportato qui sotto:

```
$DEFAULT_BASE = "dc=vostra_azienda,dc=com";
```

La migrazione di un database utente in un formato LDAP rientra in un gruppo di script di migrazione installati con il pacchetto `nss_ldap`. Con l'aiuto della Tabella 18-1, decidete quale script eseguire per poter migrare il vostro database utente.

Name service esistente	LDAP è in esecuzione?	Script da utilizzare
file lineari /etc	sì	<code>migrate_all_online.sh</code>
file lineari /etc	no	<code>migrate_all_offline.sh</code>
NetInfo	sì	<code>migrate_all_netinfo_online.sh</code>
NetInfo	no	<code>migrate_all_netinfo_offline.sh</code>
NIS (YP)	sì	<code>migrate_all_nis_online.sh</code>
NIS (YP)	no	<code>migrate_all_nis_offline.sh</code>

Tabella 18-1. Script di migrazione LDAP

Eseguite lo script più adeguato in base al vostro name service esistente.

I file `README` `migration-tools.txt` contenuti nella directory `/usr/share/openldap/migration` forniscono maggiori dettagli su come migrare le informazioni.

18.8. Risorse aggiuntive

Sono disponibili molte informazioni utili relative a LDAP. Usate queste risorse e, in particolare, visitate il sito Web OpenLDAP. Leggete LDAP HOWTO prima di iniziare a usare LDAP sul vostro sistema.

18.8.1. Documentazione installata

- La pagina man di `ldap` è un ottimo punto di partenza per conoscere questo protocollo. Esistono varie pagine man per i diversi demoni e utility.
- `/usr/share/docs/openldap-<numeroversione>` — contiene un documento generale README e informazioni varie.

18.8.2. Siti Web utili

- <http://www.openldap.org> — pagina principale del progetto OpenLDAP. Il sito contiene una ricca quantità di informazioni relative alla configurazione di OpenLDAP.
- <http://www.redhat.com/mirrors/LDP/HOWTO/LDAP-HOWTO.html> — un documento LDAP Linux HOWTO meno recente, ma comunque importante.
- <http://www.padl.com> — sviluppatori di `nss_ldap` e `pam_ldap` e altri tool LDAP.
- <http://www.kingsmountain.com/ldapRoadmap.shtml> — la Road Map LDAP di Jeff Hodges contiene link a diverse e utili FAQ e include le news sul protocollo LDAP.
- <http://www.webtechniques.com/archives/2000/05/wilcox> — un sito utile per gestire i gruppi in LDAP.
- <http://www.ldapman.org/articles> — contiene articoli che offrono una buona introduzione a LDAP, tra cui metodi per creare un'alberatura delle directory e personalizzare la loro struttura delle directory.

18.8.3. Libri correlati

- *Implementing LDAP* di Mark Wilcox, pubblicato da Wrox Press, Inc.
- *Understanding and Deploying LDAP Directory Services* di Tim Howes et al., pubblicato da Macmillan Technical Publishing

Appendici

Parametri generali e moduli

Questa appendice illustra *alcuni* dei possibili parametri disponibili per alcuni driver hardware comuni¹. In molti casi, questi parametri aggiuntivi non sono necessari, poiché il kernel potrebbe già essere in grado di usare il dispositivo senza tali parametri. I parametri contenuti in quest'appendice devono essere usati solo se Red Hat Linux ha problemi con un particolare dispositivo oppure se dovete sovrascrivere i parametri di default del sistema per il dispositivo.

Durante l'installazione di Red Hat Linux, vengono posti dei limiti su alcuni filesystem e su alcuni driver di dispositivi supportati dal kernel. Tuttavia, dopo l'installazione è disponibile un supporto per tutti i filesystem sotto Linux. Al momento dell'installazione, il kernel modularizzato ha un supporto per i dispositivi (E)IDE (compresi i CD-ROM ATAPI), gli adattatori SCSI e le schede di rete.



Nota Bene

Dato che Red Hat Linux supporta l'installazione su diverse piattaforme hardware, alcuni driver (controller SCSI, schede di rete e alcuni CD-ROM) non sono compilati all'interno del kernel di Linux utilizzato durante la fase d'installazione, ma sono disponibili come moduli e vengono caricati all'occorrenza. Se necessario, avete la possibilità di specificare le opzioni per questi moduli in fase di caricamento.

Per specificare i parametri del modulo durante il caricamento di un driver, digitate **linux expert** al prompt `boot:` e inserite il dischetto dei driver quando richiesto dal programma di installazione. Una volta letto il dischetto dei driver, il programma vi chiede di selezionare il tipo di dispositivo che state configurando. In questa schermata potete specificare un parametro del modulo. Il programma di installazione visualizza in seguito una schermata dove potete digitare i parametri corrispondenti al tipo di dispositivo che state configurando.

Una volta completata l'installazione potete ricompilare il kernel per includervi il supporto per la vostra configurazione hardware specifica. È importante notare che in molti casi non è necessario ricompilare il kernel. Per maggiori informazioni sulla ricompilazione del kernel, consultate la *Official Red Hat Linux Customization Guide*.

A.1. Come specificare i parametri dei moduli

Se state fornendo i parametri sul caricamento di un modulo, potete specificarli usando due diversi metodi:

- Specificando tutti i parametri di un insieme in un'unica frase. Per esempio il parametro `cdu31=0x340,0` può essere usato con una CDU Sony 31 o 33 sulla porta 340 senza IRQ.
- Specificando i parametri individualmente. Questo metodo viene usato quando alcuni parametri del primo insieme non sono necessari. Per esempio, `cdu31_port=0x340 cdu31a_irq=0` può essere usato come parametro del CD-ROM usato sopra. Nelle tabelle CD-ROM, SCSI ed Ethernet di quest'appendice, *O* indica che il primo metodo termina e che inizia il secondo metodo.

1. Un *driver* è un tipo di software che permette al sistema di usare un particolare dispositivo hardware. Senza il driver, il kernel potrebbe non sapere come accedere correttamente al dispositivo.

**Nota Bene**

Quando caricate un modulo con parametri particolari, utilizzate un unico metodo anziché entrambi.

**Attenzione**

Se il parametro contiene delle virgole, assicuratevi che *non* ci siano spazi dopo la virgola.

A.2. Parametri per i CD-ROM

**Nota Bene**

Non tutte le unità CD-ROM elencate sono supportate. Consultate l'elenco delle compatibilità hardware disponibile nel sito Web di Red Hat all'indirizzo <http://hardware.redhat.com> per verificare che la vostra unità sia supportata.

Sebbene alcuni parametri vengano specificati dopo avere caricato il dischetto dei driver e indicato il dispositivo, uno dei parametri più comunemente usati (`hdX=cdr0m`) può essere digitato al prompt di avvio (`boot:`). Quest'eccezione alla regola è permessa poiché serve per il supporto per i CD-ROM IDE/ATAPI, che fa già parte del kernel.

Nelle tabelle seguenti, molti moduli sono elencati senza parametri perché sono in grado di effettuare automaticamente un test oppure vi chiedono di modificare manualmente i parametri nel codice sorgente del modulo e poi di effettuare la ricompilazione.

Hardware	Modulo	Parametri
Unità CD-ROM ATAPI/IDE		<code>hdX=cdr0m</code>
Aztech CD268-01A, Orchid CD-3110, Okano/Wearnes CDD110, Conrad TXC, CyCDROM CR520, CyCDROM CR540 (non-IDE)	<code>aztcd.o</code>	<code>aztcd=porta_io</code>
CD-ROM Sony CDU 31A	<code>cdu31a.o</code>	<code>cdu31a=porta_io,IRQ OR porta=_cdu31a=base_addr cdu31a_irq=irq</code>
Lettore CDROM Philips/LMS 206 con scheda adattatore host cm260	<code>cm206.o</code>	<code>cm206=io_port,IRQ</code>
CD-ROM Goldstar R420	<code>gscd.o</code>	<code>gscd=porta_io</code>

Hardware	Modulo	Parametri
Interfaccia CD-ROM di scheda audio ISP16, MAD16 o Mozart (OPTi 82C928 e OPTi 82C929) con lettori Sanyo/Panasonic, Sony o Mitsumi	isp16.o	isp16=porta_io,IRQ,dma, tipo_drive 0 isp16_cdrom_base=porta_io isp16_cdrom_irq=IRQ isp16_cdrom_dma=dma isp16_cdrom_type=tipo_drive
CD-ROM Mitsumi standard	mcd.o	mcd=porta_io,IRQ
CD-ROM Mitsumi, sperimentale	mcdx.o	mcdx=porta_io_1,IRQ_1, porta_io_n,IRQ_n
Lettori CD-ROM di memorizzazione ottica "Dolphin" 8000 AT, Lasermate CR328A	optcd.o	
CD-ROM IDE porta parallela	pcd.o	
Scheda audio compatibile Pro 16	sbpcd.o	sbpcd=porta_io
CDR-H94A Sanyo	sjcd.o	sjcd=porta_io 0 sjcd_base=porta_io
Sony CDU-535 & 531 (alcuni lettori Procomm)	sonycd535.o	sonycd535=porta_io

Tabella A-1. Parametri hardware

Di seguito sono riportati alcuni esempi di moduli utilizzati:

Configurazione	Esempio
CD-ROM ATAPI, impostato tramite i jumper come master sul secondo canale IDE	hdc=cdrom
CD-ROM Mitsumi non IDE sulla porta 340, IRQ 11	mcd=0x340,11
Tre lettori CD-ROM Mitsumi non IDE che utilizzano il driver sperimentale, le porte IO 300, 304 e 320 con gli IRQ 5, 10 e 11	mcdx=0x300,5,0x304,10,0x320,11
CDU Sony 31 o 33 sulla porta 340, senza IRQ	cdu31=0x340,0 0 cdu31_port=0x340 cdu31a_irq=0
CD-ROM Aztech sulla porta 220	aztcd=0x220
CD-ROM di tipo Panasonic su un'interfaccia SoundBlaster collegata alla porta 230	sbpcd=0x230,1
Phillips/LMS cm206 e cm260 su IO 340 e IRQ 11	cm206=0x340,11
Goldstar R420 su IO 300	gscd=0x300
Lettoce Mitsumi su scheda MAD16 su IO Addr 330 e IRQ 1, test DMA	isp16=0x330,11,0,Mitsumi
Sony CDU 531 su indirizzo di IO 320	sonycd535=0x320

Tabella A-2. Esempi di configurazione per i parametri hardware

**Nota Bene**

Le schede Sound Blaster più recenti hanno un'interfaccia IDE. Per queste schede non è necessario utilizzare i parametri `sbpcd`, utilizzate solo i parametri `hdX`.

A.3. Parametri SCSI

Hardware	Modulo	Parametri
Adaptec 28xx, R9xx, 39xx	aic7xxx.o	
Controller dei dischi 3ware	3w-xxxx.o	
NCR53c810/820/720, NCR53c700/710/700-66	53c7,8xx.o	
Driver AM53/79C974 (PC-SCSI)	AM53C974.o	
Quasi tutte le schede Buslogic (ora Mylex) con numero di parte "BT"	BusLogic.o	
Controller RAID Mylex DAC960	DAC960.o	
SCSI basato su MCR53c406a	NCR53c406a.o	
Initio INI-A100U2W	a100u2w.o	a100u2w=io,IRQ,scsi_id
Adaptec AACRAID	aacraid.o	
Schede SCSI Advansys	advansys.o	
Adaptec AHA-152x	aha152x.o	aha152x=io,IRQ,scsi_id
Adaptec AHA 154x amd 631x-based	aha1542.o	
Adaptec AHA 1740	aha1740.o	

Hardware	Modulo	Parametri
Adaptec AHA-274x, AHA-284x, AHA-29xx, AHA-394x, AHA-398x, AHA-274x, AHA-274xT, AHA-2842, AHA-2910B, AHA-2920C, AHA-2930/U/U2, AHA-2940/W/U/UW/AU/, U2W/U2/U2B/, U2BOEM, AHA-2944D/WD/UD/UWD, AHA-2950U2/W/B, AHA-3940/U/W/UW/, AUW/U2W/U2B, AHA-3950U2D, AHA-3985/U/W/UW, AIC-777x, AIC-785x, AIC-786x, AIC-787x, AIC-788x , AIC-789x, AIC-3860	aic7xxx.o	
Controller SCSI PCI ACARD ATP870U	atp870u.o	
Controller Compaq Smart Array 5300	cciss.o	
Controller RAID Compaq Smart2	cpqarray.o	
Controller FibreChannel Compaq	cpqfc.o	
Domex DMX3191D	dmx3191d.o	
Data Technology Corp DTC3180/3280	dtc.o	
Adattatori host SCSI DTP (EATA/DMA) PM2011B/9X ISA, PM2021A/9X ISA, PM2012A, PM2012B, PM2022A/9X EISA, PM2122A/9X, PM2322A/9X, SmartRAID PM3021, PM3222, PM3224	eata.o	
Adattatori host SCSI DTP PM2011, PM2021, PM2041, PM3021, PM2012B, PM2022, PM2122, PM2322, PM2042, PM3122, PM3222, PM3332, PM2024, PM2124, PM2044, PM2144, PM3224, PM3334	eata_dma.o	
Array di rete Sun Enterprise (FC-AL)	fc.al.o	
Future Domain TMC-16xx SCSI	fdomain.o	
NCR5380 (driver generico)	g_NCR5380.o	

Hardware	Modulo	Parametri
Controller RAID ICP	gdth.o	
Driver Block I2O	i2o_block.o	
Adattatore SCSI porta parallela IOMEGA MatchMaker	imm.o	
Scheda SCSI ISA Always IN2000	in2000.o	in2000= <i>setup_string</i> :valore O in2000 <i>setup_string</i> =valore
Adattatori host SCSI Initio INI-9X00U/UW	initio.o	
ServeRAID IBM	ips.o	
AMI MegaRAID 418, 428, 438, 466, 762	megaraid.o	
Controller SCSI NCR con chipset 810/810A/815/825/825A/860/875/876/895	ncr53c8xx.o	ncr53c8xx= <i>opzione1</i> :valore1, <i>opzione2</i> :valore2 O ncr53c8xx=" <i>opzione1</i> :valore1 <i>opzione2</i> :valore2..."
Pro Audio Spectrum/Studio 16	pas16.o	
PCI-2000 IntelliCache	pci2000.o	
RAID EIDE PCI-2220I	pci2220i.o	
Adattatore host SCSI porta parallela IOMEGA PPA3	ppa.o	
Perceptive Solutions PSI-240I EIDE	psi240i.o	
Qlogic 1280	qla1280.o	
Qlogic 2x00	qla2x00.o	
QLogic Fast SCSI FASXXX ISA/VLB/PCMCIA	qlogicfas.o	
QLogic ISP2100 SCSI-FCP	qlogicfc.o	
Schede SCSI QLogic ISP1020 Intelligent IQ-PCI, IQ-PCI-10, IQ-PCI-D	qlogicisp.o	
Qlogic ISP1020 SCSI SBUS	qlogicpti.o	
Future Domain TMC-885, TMC-950 Seagate ST-01/02, Future Domain TMC-8xx	seagate.o	controller_type=2 base_address=base_addr irq=IRQ
Schede con chipset sym53c416	sym53c416.o	sym53c416=PORTBASE,[IRQ] OR sym53c416 io=PORTBASE irq=IRQ
Adattatore host SCSI Trantor T128/T128F/T228	t128.o	
Tekram DC-390(T) PCI	tmscsim.o	

Hardware	Modulo	Parametri
UltraStor 14F/34F (non 24F)	ul14-34f.o	
UltraStor 14F, 24F e 34F	ultrastor.o	
WD7000 Series	wd7000.o	

Tabella A-3. Parametri SCSI

Di seguito sono riportati alcuni esempi di moduli utilizzati:

Configurazione	Esempio
Adaptec AHA1522 sulla porta 330, IRQ 11, SCSI ID 7	aha152x=0x330,11,7
Adaptec AHA1542 sulla porta 330	bases=0x330
Future Domain TMC-800 su CA000, IRQ 10	controller_type=2 base_address=0xca000 irq=10

Tabella A-4. Esempi di configurazione dei parametri SCSI

A.4. Parametri Ethernet

Hardware	Modulo	Parametri
3Com 3c501	3c501.o	3c501=porta_io,IRQ
3Com 3c503 e 3c503/16	3c503.o	3c503=porta_io,IRQ O 3c503 io=porta_io_1,porta_io_n irq=IRQ_1,IRQ_n
3Com EtherLink Plus (3c505)	3c505=porta_io,IRQ O 3c505 io=porta_io_1,porta_io_n irq=IRQ_1,IRQ_2	
3Com EtherLink 16	3c507.o	3c507=porta_io,IRQ O 3c507 io=porta_io irq=IRQ
3Com EtherLink III	3c509.o	3c509=porta_io,IRQ
3Com ISA EtherLink XL "Corkscrew"	3c515.o	
3Com EtherLink PCI III/XL Vortex (3c590, 3c592, 3c595, 3c597) Boomerang (3c900, 3c905, 3c595)	3c59x.o	full_duplex= 0 è off 1 è on
RTL8139, SMC EZ Card Fast Ethernet	8139too.o	
Schede RealTek che usano chipset RTL8129 o RTL8139 Fast Ethernet	8139too.o	

Hardware	Modulo	Parametri
Apricot 82596	82596.o	
Ansel Communications Model 3200	ac3200.o	ac3200= <i>porta_io</i> , <i>IRQ</i> <i>O</i> ac3200 io= <i>porta_io_1</i> , <i>porta_io_n</i> irq= <i>IRQ_1</i> , <i>IRQ_n</i>
Alteon AceNIC Gigabit	acenic.o	
Aironet Arlan 655	arlan.o	
Allied Telesis AT1700	at1700.o	at1700= <i>porta_io</i> , <i>IRQ</i> <i>O</i> at1700 io= <i>porta_io</i> irq= <i>IRQ</i>
Broadcom BCM5700 10/100/1000 ethernet adapter	bcm5700.o	
Crystal Semiconductor CS89[02]0	cs89x0.o	
Schede EtherWORKS DE425 TP/COAX EISA, DE434 TP PCI, DE435/450 TP/COAX/AUI PCI DE500 10/100 PCI Kingston, LinkSys, SMC8432, SMC9332, Znyx31[45] e Znyx346 10/100 con chipset DC21040 (no SRAM), DC21041[A], DC21140[A], DC21142, DC21143	de4x5= <i>porta_io</i> <i>O</i> de4x5 io= <i>porta_io</i> de4x5 args='ethX[<i>fdx</i>] au- tosense= <i>MEDIA_STRING</i> '	
Adapter Pocket Ethernet D-Link DE-600	de600.o	
Adapter Pocket Ethernet D-Link DE-620	de620.o	
DIGITAL DEPCA & EtherWORKS DEPCA, DE100, DE101, DE200 Turbo, DE201 Turbo DE202 Turbo TP/BNC, DE210, DE422 EISA	depca.o	depca= <i>porta_io</i> , <i>IRQ</i> <i>O</i> depca io= <i>porta_io</i> irq= <i>IRQ</i>
Digi Intl. RightSwitch SE-X EISA e PCI	dgrs.o	
Davicom DM9102(A)/DM9132/DM9801 Fast Ethernet	dmfe.o	
Intel Ether Express/100 driver	e100.o	e100_speed_duplex= <i>X</i> Se <i>X</i> = 0 = autodetect speed and duplex 1 = 10Mbps, half duplex 2 = 10Mbps, full duplex 3 = 100Mbps, half duplex 4 = 100Mbps, full duplex

Hardware	Modulo	Parametri
Intel EtherExpress/1000 Gigabit	e1000.o	
Cabletron E2100	e2100.o	e2100=porta_io,IRQ O e2100 io=porta_io irq=IRQ mem=mem
EtherExpress Pro10 Intel	eeepro.o	eeepro=porta_io,IRQ O eeepro io=porta_io irq=IRQ
Driver Intel i82557/i82558 PCI EtherExpressPro	eeepro100.o	
Intel EtherExpress 16 (i82586)	eexpress.o	eexpress=porta_io,IRQ O eexpress io=porta_io irq=IRQ options= 0x10 10base T half duplex 0x20 10base T full duplex 0x100 100base T half duplex 0x200 100baseT full duplex
SMC EtherPower II 9432 PCI (serie EPIC 83c170/175)	epic100.o	
Racal-Interlan ES3210 EISA	es3210.o	
ICL EtherTeam 16i/32 EISA	eth16i.o	eth16i=porta_io,IRQ O eth16i ioaddr=porta_io IRQ=IRQ
EtherWORKS 3 (DE203, DE204 and DE205)	ewrk3.o	ewrk=porta_io,IRQ O ewrk io=porta_io irq=IRQ
A Packet Engines GNIC-II Gigabit	hamachi.o	
HP PCLAN/plus	hp-plus.o	hp-plus=porta_io,IRQ O hp-plus io=porta_io irq=IRQ
HP LAN Ethernet	hp.o	hp=porta_io,IRQ O hp io=porta_io irq=IRQ
Adattatori di rete 100VG-AnyLan HP J2585B, J2585A, J2970, J2973, J2573 Compex ReadyLink ENET100-VG4, FreedomLine 100/VG	hp100.o	hp100=porta_io,name O hp100 hp100_port=porta_io hp100_name=name
IBM Token Ring 16/4, Shared-Memory IBM Token Ring 16/4	ibmtr.o	ibmtr=porta_io O io=porta_io
AT1500, HP J2405A, la maggior parte dei cloni NE2100	lance.o	
Mylex LNE390 EISA	lne390.o	

Hardware	Modulo	Parametri
NatSemi DP83815 Fast Ethernet	natsemi.o	
NE1000 / NE2000 (non-pci)	ne.o	ne=porta_io,IRQ O ne io=porta_io irq=IRQ
Schede PCI NE2000 RealTEK RTL-8029, Winbond 89C940, Compex RL2000, PCI NE2000 clones, NetVin, NV5000SC, Via 82C926, SureCom NE34	ne2k-pci.o	
Novell NE3210 EISA	ne3210.o	
MiCom-Interlan NI5010	ni5010.o	
Scheda NI5210 (i82586 Ethernet chip)	ni52.o	ni52=porta_io,IRQ O ni52 io=porta_io irq=IRQ
NI6510 Ethernet	ni65.o	
IBM Olympic-based PCI token ring	olympic.o	
AMD PCnet32 e AMD PCnetPCI	pcnet32.o	
SIS 900/701G PCI Fast Ethernet	sis900.o	
SysKonnnect SK-98XX Gigabit	sk98lin.o	
SMC Ultra e SMC EtherEZ ISA ethercard (8K, 83c790)	smc-ultra.o	smc-ultra=porta_io,IRQ O smc-ultra io=porta_io irq=IRQ
Scheda Ethernet SMC Ultra32 EISA (32K)	smc-ultra32.o	
Sun BigMac Ethernet	sunbmac.o	
Sundance ST201 Alta	sundance.o	
Sun Happy Meal Ethernet	sunhme.o	
Sun Quad Ethernet	sunqe.o	
ThunderLAN	tlan.o	

Hardware	Modulo	Parametri
Schede Digital 21x4x Tulip PCI Ethernet SMC EtherPower 10 PCI(8432T/8432BT) SMC EtherPower 10/100 PCI(9332DST) DEC EtherWorks 100/10 PCI(DE500-XA) DEC EtherWorks 10 PCI(DE450) DEC QSILVER's, Znyx 312 etherarray Allied Telesis LA100PCI-T Danpex EN-9400, Cogent EM110	tulip.o	io=porta_io
Schede Ethernet Fast VIA Rhine PCI con PCI VIA VT86c100A Rhine-II o 3043 Rhine-I D-Link DFE-930-TX 10/100	via-rhine.o	
AT&T GIS (nee NCR) WaveLan ISA Card	wavelan.o	wavelan=[IRQ,0],porta_io,NWID
Schede Ethernet compatibili WD8003 e WD8013	wd.o	wd=porta_io,IRQ,mem,mem_end O wd io=porta_io irq=IRQ mem=mem mem_end=end
Compex RL100ATX-PCI	winbond.o	
Packet Engines Yellowfin	yellowfin.o	

Tabella A-5. Parametri del modulo Ethernet

Di seguito sono riportati alcuni esempi di moduli utilizzati:

Configurazione	Esempio
Scheda ISA NE2000 sull'indirizzo di IO 300 e IRQ 11	ne=0x300,11 ether=0x300,11,eth0
Scheda Wavelan su IO 390, autotest per IRQ e utilizzo di NWID fino a 0x4321	wavelan=0,0x390,0x4321 ether=0,0x390,0x4321,eth0

Tabella A-6. Esempi di configurazione dei parametri Ethernet

A.4.1. Utilizzo di schede Ethernet multiple

Potete utilizzare più schede Ethernet su una macchina. Se ciascuna scheda utilizza un driver diverso (per esempio, un 3c509 e un DE425), dovrete semplicemente aggiungere degli *alias* (e possibilmente delle *opzioni*) per ciascuna scheda a `/etc/conf.modules`. Per maggiori informazioni, consultate la *Official Red Hat Linux Customization Guide*.

Se due schede Ethernet utilizzano lo stesso driver (per esempio, due 3c509 o una 3c595 e una 3c905), nel caso di schede ISA avrete bisogno di specificare gli indirizzi di entrambe le schede nella linea delle opzioni del driver e nel caso di schede PCI dovrete aggiungere una linea di *alias* per ciascuna scheda PCI.

Per maggiori informazioni sull'uso di più schede Ethernet, consultate il *Linux Ethernet-HOWTO* all'indirizzo <http://www.redhat.com/mirrors/LDP/HOWTO/Ethernet-HOWTO.html>.

Indice

Symbols

- .fetchmailrc, 227
 - opzioni globali, 228
 - opzioni server, 228
 - opzioni utente, 229
- .procmailrc, 231
- /etc/exports, 262
- /etc/fstab, 264
- /etc/named.conf
 - (Vd. BIND)
- /etc/pam.conf, 115
 - (Vd. Anche PAM)
- /etc/pam.d, 115
 - (Vd. Anche PAM)
- /etc/sysconfig directory, 24
- /lib/security/, 115
 - (Vd. Anche PAM)
- /var/lib/rpm directory, 24
- /var/spool/up2date directory, 24
- protocollo SSH
 - file di configurazione, 136

A

- aboot, 64, 81
- AccessFileName
 - direttiva di configurazione di Apache, 204
- Action
 - direttiva di configurazione di Apache, 210
- AddDescription
 - direttiva di configurazione di Apache, 209
- AddEncoding
 - direttiva di configurazione di Apache, 209
- AddHandler
 - direttiva di configurazione di Apache, 210
- AddIcon
 - direttiva di configurazione di Apache, 209
- AddIconByEncoding
 - direttiva di configurazione di Apache, 208
- AddIconByType
 - direttiva di configurazione di Apache, 208
- AddLanguage
 - direttiva di configurazione di Apache, 210
- AddType
 - direttiva di configurazione di Apache, 210
- Alias
 - direttiva di configurazione di Apache, 207
- Allow
 - direttiva di configurazione di Apache, 203
- AllowOverride
 - direttiva di configurazione di Apache, 203
- ambienti desktop, 107

(Vd. Anche XFree86)

Apache

- Apache HTTP Server, 185
- Apache HTTP Server
 - avvio, 196
 - chiusura, 196
 - configurazione, 197
 - file di log, 197
 - introduzione, 185
 - lavorare in modalità non sicura, 215
 - report sullo stato del server, 211
 - riavvio, 196
 - ricaricamento, 196
 - risoluzione dei problemi, 197
 - risorse aggiuntive, 217
 - libri correlati, 217
 - siti Web utili, 217
 - versione 1.3
 - migrazione alla 2.0, 187
 - versione 2.0
 - caratteristiche di, 185
 - migrazione dalla 1.3, 187
 - modifiche ai pacchetti, 186
 - modifiche al filesystem, 186
- attacco Denial of Service
 - (Vd. attacco DoS)
- attacco DoS, 52
 - (Vd. Anche directory /proc/sys/net/)
 - definizione di, 52
- authconfig
 - e LDAP, 276, 277
- autofs, 264

B

- Basic Input/Output System
 - (Vd. BIOS)
- Berkeley Internet Name Domain
 - (Vd. BIND)
- BIND
 - caratteristiche, 254
 - IPv6, 256
 - miglioramenti DNS, 255
 - sicurezza, 255
 - visualizzazioni multiple, 255
 - configurazione di
 - direttive dei file della zona, 248
 - esempi di file della zona, 250
 - record di risorsa del file della zona , 248
 - risoluzione nomi inversa, 251
 - demone named, 242
 - errori comuni, 256
 - esempi di istruzione zone, 247
 - file di configurazione, 243
 - /etc/named.conf, 243

- directory /var/named/, 243
- file della zona, 247
- introduzione, 241, 241
- programma rndc, 252
 - /etc/rndc.conf, 253
 - configurazione delle chiavi, 253
 - configurazione di named per l'uso di, 252
 - opzioni della linea di comando, 254
- risorse aggiuntive, 256
 - documentazione installata, 256
 - libri correlati, 257
 - siti Web utili, 257
- server dei nomi
 - definizione di, 241
- server dei nomi root
 - definizione di, 241
- tipi di server dei nomi
 - caching-only, 242
 - forwarding, 242
 - master, 242
 - slave, 242
- zone
 - definizione di, 241

BIOS

- definizione di, 59
 - (Vd. Anche processo di avvio)

boot loader, 81, 81, 88, 81

- (Vd. Anche ELILO)

- definizione di, 81
- tipi di, 81

BrowserMatch

- direttiva di configurazione di Apache, 211

C

- CacheNegotiatedDocs
 - direttiva di configurazione di Apache, 204
- CD-ROM modules
 - (Vd. kernel modules)
- chiusura, 79
- chkconfig, 66
 - (Vd. Anche servizi)
- comando init, 61
 - (Vd. Anche processo di avvio)
 - accesso ai runlevel da parte del, 65
 - directory utilizzate da, 64
 - ruolo nel processo di avvio, 61
 - (Vd. Anche processo di avvio)
- SysV init
 - definizione di, 64
- comando ldapadd, 270
 - (Vd. Anche LDAP)
- comando ldapdelete, 270
 - (Vd. Anche LDAP)
- comando ldapmodify, 270

- (Vd. Anche LDAP)
- comando ldapsearch, 270
 - (Vd. Anche LDAP)
- comando slapadd, 270
 - (Vd. Anche LDAP)
- comando slapcat, 270
 - (Vd. Anche LDAP)
- comando slapd, 270
 - (Vd. Anche LDAP)
- comando slapindex, 270
 - (Vd. Anche LDAP)
- comando slappasswd, 270
 - (Vd. Anche LDAP)
- comando slurpd, 270
 - (Vd. Anche LDAP)
- configurazione
 - Apache, 197
 - host virtuali, 215
 - SSL, 213
- controllo dell'accesso
 - (Vd. wrapper TCP)
- /etc/hosts.allow, 123
- convenzioni
 - documento, xii
- copiare e incollare testi
 - usando X, xv
- CustomLog
 - direttiva di configurazione di Apache, 206

D

- DefaultIcon
 - direttiva di configurazione di Apache, 209
- DefaultType
 - direttiva di configurazione di Apache, 205
- demone named
 - (Vd. BIND)
- Denial of Service, 128
 - (Vd. Anche xinetd)
- Deny
 - direttiva di configurazione di Apache, 203
- directory
 - /dev, 20
 - /etc, 20
 - /lib, 20
 - /mnt, 20
 - /opt, 20
 - /proc, 21
 - /sbin, 21
 - /usr, 21
 - /usr/local, 22, 23
 - /var, 22
 - direttiva di configurazione di Apache, 202
- directory /dev, 20
- directory /etc, 20

- directory /etc/sysconfig
 - (Vd. directory sysconfig)
- directory /lib, 20
- directory /mnt, 20
- directory /opt, 20
- directory /proc, 21
- directory /sbin, 21
- directory /usr, 21
- directory /usr/local, 22, 23
- directory /var, 22
- Directory initrd, 24
- directory proc
 - (Vd. filesystem proc)
- directory public_html, 204
- directory sysconfig
 - /etc/sysconfig/amd, 68
 - /etc/sysconfig/apmd, 68
 - /etc/sysconfig/arpwatch, 68
 - /etc/sysconfig/authconfig, 68
 - /etc/sysconfig/clock, 69
 - /etc/sysconfig/desktop, 69
 - /etc/sysconfig/dhcpd, 70
 - /etc/sysconfig/firstboot, 70
 - /etc/sysconfig/gpm, 70
 - /etc/sysconfig/harddisks, 70
 - /etc/sysconfig/hwconf, 71
 - /etc/sysconfig/identd, 71
 - /etc/sysconfig/init, 71
 - /etc/sysconfig/ipchains, 72
 - /etc/sysconfig/iptables, 72, 182
 - /etc/sysconfig/irda, 72
 - /etc/sysconfig/keyboard, 73
 - /etc/sysconfig/kudzu, 73
 - /etc/sysconfig/mouse, 74
 - /etc/sysconfig/named, 74
 - /etc/sysconfig/netdump, 75
 - /etc/sysconfig/network, 75
 - /etc/sysconfig/ntpd, 75
 - /etc/sysconfig/pcmcia, 75
 - /etc/sysconfig/radvd, 76
 - /etc/sysconfig/rawdevices, 76
 - /etc/sysconfig/redhat-config-users, 76
 - /etc/sysconfig/redhat-logviewer, 76
 - /etc/sysconfig/samba, 77
 - /etc/sysconfig/sendmail, 77
 - /etc/sysconfig/soundcard, 77
 - /etc/sysconfig/squid, 77
 - /etc/sysconfig/tux, 77
 - /etc/sysconfig/ups, 78
 - /etc/sysconfig/vncservers, 78
 - /etc/sysconfig/xinetd, 79
- directory /etc/sysconfig/apm-scripts, 79
- directory /etc/sysconfig/cbq, 79
- directory /etc/sysconfig/network-scripts, 165, 79
 - (Vd. Anche rete)
- directory /etc/sysconfig/networking, 79
- directory /etc/sysconfig/rhn, 79
- directory sysconfig/
 - file disponibili nella, 67
- DirectoryIndex
 - direttiva di configurazione di Apache, 204
- direttiva di configurazione, Apache
 - ReadmeName, 209
- direttive della cache per Apache, 212
- direttive di configurazione, Apache, 198
 - AccessFileName, 204
 - Action, 210
 - AddDescription, 209
 - AddEncoding, 209
 - AddHandler, 210
 - AddIcon, 209
 - AddIconByEncoding, 208
 - AddIconByType, 208
 - AddLanguage, 210
 - AddType, 210
 - Alias, 207
 - Allow, 203
 - AllowOverride, 203
 - BrowserMatch, 211
 - CacheNegotiatedDocs, 204
 - CustomLog, 206
 - DefaultIcon, 209
 - DefaultType, 205
 - Deny, 203
 - Directory, 202
 - DirectoryIndex, 204
 - DocumentRoot, 202
 - ErrorDocument, 211
 - ErrorLog, 206
 - ExtendedStatus, 201
 - Group, 201
 - HeaderName, 209
 - HostnameLookups, 205
 - IfDefine, 200
 - IfModule, 205
 - Include, 200
 - IndexIgnore, 209
 - IndexOptions, 208
 - KeepAlive, 199
 - KeepAliveTimeout, 199
 - LanguagePriority, 210
 - Listen, 200
 - LoadModule, 200
 - Location, 211
 - LockFile, 198
 - LogFormat, 206
 - LogLevel, 206
 - MaxClients, 199
 - MaxKeepAliveRequests, 199
 - MaxRequestsPerChild, 200
 - MaxSpareServers, 199
 - MetaDir, 211

MetaSuffix, 211
 MinSpareServers, 199
 NameVirtualHost, 213
 Options, 203
 Order, 203
 per la funzionalità della cache, 212
 per la funzionalità SSL, 213
 PidFile, 198
 ProxyRequests, 212
 ProxyVia, 212
 Redirect, 208
 ScoreBoardFile, 198
 ScriptAlias, 207
 ServerAdmin, 201
 ServerName, 202
 ServerRoot, 198
 ServerSignature, 207
 SetEnvIf, 213
 StartServers, 199
 Timeout, 198
 TypesConfig, 205
 UseCanonicalName, 205
 User, 201
 UserDir, 204
 VirtualHost, 213
 direttive SSL, 213
 display manager
 (Vd. XFree86)
 dispositivi a blocchi
 /proc/devices, 28
 definizione di, 28
 dispositivi a carattere, 28
 (Vd. Anche /proc/devices)
 definizione di, 28
 dispositivi, locali
 proprietà dei, 121
 (Vd. Anche PAM)
 dispositivo frame buffer, 30
 (Vd. Anche /proc/fb)
 DNS, 241
 (Vd. Anche BIND)
 documentazione
 guru, xii
 ricerca, x
 utenti esperti, xii
 utenti inesperti, x
 libri, xii
 newsgroup, xi
 siti Web, xi
 DocumentRoot
 direttiva di configurazione di Apache, 202
 modifica, 215
 modifica della condivisione, 216
 DoS
 (Vd. Denial of Service)
 drag-and-drop, xv

driver
 (Vd. moduli del kernel)
 DSO
 caricamento, 215

E

e-mail, 219
 Fetchmail, 227
 Procmail, 230
 protocolli, 219
 IMAP, 219
 POP, 220
 SMTP, 220
 risorse aggiuntive, 237
 documentazione installata, 237
 libri correlati, 238
 siti Web utili, 238
 Sendmail, 222
 sicurezza, 236
 client, 236
 server, 237
 tipi
 Mail Delivery Agent, 222
 Mail Transfer Agent, 221
 Mail User Agent, 221
 tipi di programmi, 221
 ELILO, 64, 81
 epoca, 39
 (Vd. Anche /proc/stat)
 definizione di, 39
 ErrorDocument
 direttiva di configurazione di Apache, 211
 ErrorLog
 direttiva di configurazione di Apache, 206
 Ethernet
 (Vd. rete)
 ExtendedStatus
 direttiva di configurazione di Apache, 201

F

- Fetchmail, 227
 - opzioni di comando, 229
 - informative, 230
 - speciali, 230
 - opzioni di configurazione, 227
 - opzioni globali, 228
 - opzioni server, 228
 - opzioni utente, 229
 - risorse aggiuntive, 237
- FHS, 19, 20
 - (Vd. Anche filesystem)
- file di log
 - formato comune dei file di log, 206
- file virtuali
 - (Vd. filesystem proc)
- file, filesystem proc
 - modifica, 26
 - visualizzazione, 25
- file, filesystem proc
 - modifica, 56
 - visualizzazione, 56
- filesystem
 - FHS standard, 20
 - gerarchia, 19
 - organizzazione, 20
 - struttura, 19
 - virtuale
 - (Vd. filesystem proc)
- filesystem /proc
 - file nel, di livello superiore, 26
- filesystem proc
 - /proc/apm/, 27
 - /proc/cmdline, 27
 - /proc/cpuinfo, 27
 - /proc/devices
 - dispositivi a blocchi, 28
 - dispositivi a carattere, 28
 - /proc/dma, 29
 - /proc/execdomains, 29
 - /proc/fb, 30
 - /proc/filesystem, 30
 - /proc/interrupts, 30
 - /proc/iomem, 31
 - /proc/ioports, 32
 - /proc/isapnp, 32
 - /proc/kcore, 33
 - /proc/kmsg, 33
 - /proc/ksyms, 33
 - /proc/loadavg, 34
 - /proc/locks, 34
 - /proc/mdstat, 34
 - /proc/meminfo, 35
 - /proc/misc, 36
 - /proc/modules, 36
 - /proc/mounts, 36
 - /proc/mtrr, 37
 - /proc/partitions, 37
 - /proc/pci
 - visualizzazione mediante lspci, 37
 - /proc/slabinfo, 38
 - /proc/stat, 39
 - /proc/swaps, 39
 - /proc/uptime, 39
 - /proc/version, 40
 - directory /proc/bus/, 42
 - directory /proc/driver/, 43
 - directory /proc/fs/, 43
 - directory /proc/ide
 - directory dei dispositivi, 44
 - directory /proc/ide/, 43
 - directory /proc/irq/, 45
 - directory /proc/net/, 45
 - directory /proc/scsi/, 46
 - directory /proc/self/, 42
 - directory /proc/sys/, 48, 56
 - (Vd. Anche sysctl)
 - /proc/sys/kernel/sysrq
 - (Vd. tasto SysRq)
 - directory /proc/sys/dev/, 49
 - directory /proc/sys/fs/, 50
 - directory /proc/sys/kernel/, 50
 - directory /proc/sys/net/, 52
 - directory /proc/sys/vm/, 54
 - directory /proc/sysvipc/, 55
 - directory /proc/tty/, 55
 - directory di processo, 40
 - introduzione, 25
 - modifica dei file nel, 26, 48, 56
 - risorse aggiuntive, 57
 - documentazione installata, 57
 - siti Web utili, 57
 - sottodirectory nel, 40
 - visualizzazione dei file nel, 25
- filesystem virtuale
 - (Vd. filesystem proc)
- filtraggio di pacchetti
 - (Vd. iptables)
- formati degli eseguibili, 29
 - (Vd. Anche /proc/execdomains)
- definizione di, 29
- formato comune dei file di log, 206
- FrontPage, 196

G

- gerarchia, filesystem, 19
- GNOME, 107
 - (Vd. Anche XFree86)
- Group
 - direttiva di configurazione di Apache, 201
- GRUB, 60
 - (Vd. Anche boot loader)
 - caratteristiche, 82
 - comandi, 86
 - definizione di, 81
 - file di configurazione
 - /boot/grub/grub.conf, 88
 - struttura, 88
 - file di configurazione del menu, 87
 - comandi, 87
 - installazione, 82
 - interfacce, 85
 - a menu, 85
 - editor voci di menu, 85
 - linea di comando, 85
 - sequenza di utilizzo, 85
 - modifica dei runlevel con, 85, 91
 - processo di avvio, 81
 - risorse aggiuntive, 91
 - documentazione installata, 92
 - siti Web utili, 92
 - ruolo nel processo di avvio, 60
 - terminologia, 83
 - dispositivi, 83
 - file, 84
 - filesystem root, 84
 - grub.conf, 88
 - (Vd. Anche GRUB)
- gruppi, 93
 - definizione di, 93
 - GID, 93
 - standard, 95
 - strumenti per la gestione di
 - groupadd, 93, 97
 - redhat-config-users, 93, 97
 - utente privati, 97
 - logica relativa, 98
 - utenti privati, 93
 - gruppi utente privati
 - (Vd. gruppi)
 - logica relativa, 98

H

- HeaderName
 - direttiva di configurazione di Apache, 209
 - host virtuali
 - basati sul nome, 215
 - configurazione, 215
 - include lato server, 203, 210
 - Listen command, 216
 - Options, 203
 - HostnameLookups
 - direttiva di configurazione di Apache, 205
 - hosts.allow
 - (Vd. wrapper TCP)
 - hosts.deny
 - (Vd. wrapper TCP)
 - httpd.conf
 - (Vd. direttive di configurazione, Apache)
-
- I**
- IfDefine
 - direttiva di configurazione di Apache, 200
 - ifdown, 170
 - IfModule
 - direttiva di configurazione di Apache, 205
 - ifup, 170
 - Include
 - direttiva di configurazione di Apache, 200
 - include lato server, 203, 210
 - host virtuali, 203
 - IndexIgnore
 - direttiva di configurazione di Apache, 209
 - IndexOptions
 - direttiva di configurazione di Apache, 208
 - interruzione, 79
 - introduzione, ix
 - ipchains
 - (Vd. iptables)
 - iptables
 - catene
 - target, 173
 - concetti base sul filtraggio dei pacchetti, 173
 - elenco delle regole, 173
 - opzioni, 175
 - comandi, 176
 - elenco, 182
 - parametri, 177
 - struttura, 176
 - tabelle, 175
 - target, 181
 - opzioni estese, 178
 - moduli, 180
 - panoramica su, 173
 - paragonato a ipchains, 174
 - protocolli

- ICMP, 179
- TCP, 178
- UDP, 179
- risorse aggiuntive, 183
 - documentazione installata, 183
 - siti Web utili, 183
- salvataggio delle regole, 182
- tabelle, 173

K

- KDE, 107
 - (Vd. Anche XFree86)
- KeepAlive
 - direttiva di configurazione di Apache, 199
- KeepAliveTimeout
 - direttiva di configurazione di Apache, 199
- Kerberos
 - configurazione del server, 144
 - configurazione di client, 146
 - definizione di, 141
 - e PAM, 144
 - funzionamento, 143
 - Key Distribution Center (KDC), 143
 - Risorse aggiuntive, 147
 - Documentazione installata, 147
 - siti Web utili, 147
 - svantaggi di, 141
 - terminologia, 142
 - Ticket Granting Service(TGS), 143
 - Ticket Granting Ticket (TGT), 143
 - vantaggi di, 141
- kernel
 - ruolo nel processo di avvio, 61
- kernel modules
 - CD-ROM modules
 - examples, 283

L

- LanguagePriority
 - direttiva di configurazione di Apache, 210
- LDAP
 - applicazioni, 272
 - ldapadd, 270
 - ldapdelete, 270
 - ldapmodify, 270
 - ldapsearch, 270
 - OpenLDAP suite, 270
 - slapadd, 270
 - slapcat, 270
 - slapd, 270
 - slapindex, 270
 - slappasswd, 270
 - slurpd, 270

- utility, 270
- autenticazione mediante
 - authconfig, 276
 - configurazione dei client, 276
 - modifica di /etc/ldap.conf, 276
 - modifica di /etc/nsswitch.conf, 276
 - modifica di /etc/openldap/ldap.conf, 276
 - modifica di slapd.conf, 276
 - pacchetti, 276
 - PAM, 277
- caratteristiche di OpenLDAP, 269
- configurazione, 275
 - migrazione delle directory 1.x, 275
- definizione di, 269
- demoni, 270
- file di configurazione
 - /etc/ldap.conf, 273
 - /etc/openldap/ldap.conf, 273
 - /etc/openldap/schema/ directory, 273
 - /etc/openldap/slapd.conf, 273, 273
 - directory /etc/openldap/schema/, 274
- LDAPv2, 269
- LDAPv3, 269
- risorse aggiuntive, 278
 - documentazione installata, 278
 - libri correlati, 278
 - siti Web utili, 278
- terminologia, 272
- usato con Apache HTTP Server, 271
- usato con NSS, 271
- usato con PAM, 271
- usato con PHP4, 271
- vantaggi di, 269
- Lightweight Directory Access Protocol (Vd. LDAP)
- LILO, 60
 - (Vd. Anche boot loader)
 - definizione di, 88
 - file di configurazione
 - /etc/lilo.conf, 90
 - modifica dei runlevel con, 91
 - processo di avvio, 88
 - risorse aggiuntive, 91
 - documentazione installata, 92
 - siti Web utili, 92
 - ruolo nel processo di avvio, 60
- lilo.conf, 90
 - (Vd. Anche LILO)
- Listen
 - direttiva di configurazione di Apache, 200
- LoadModule
 - direttiva di configurazione di Apache, 200
- Location
 - direttiva di configurazione, 211
- LockFile
 - direttiva di configurazione di Apache, 198

LogFormat
 direttiva di configurazione di Apache, 206
 LogLevel
 direttiva di configurazione di Apache, 206
 lspci, 37

M

Mail Delivery Agent, 222
 Mail Transfer Agent, 221
 Mail User Agent, 221
 Master Boot Record
 (Vd. MBR)
 (Vd. MBR)
 MaxClients
 direttiva di configurazione di Apache, 199
 MaxKeepAliveRequests
 direttiva di configurazione di Apache, 199
 MaxRequestsPerChild
 direttiva di configurazione di Apache, 200
 MaxSpareServers
 direttiva di configurazione di Apache, 199
 MBR
 definizione di, 59, 59
 (Vd. Anche processo di avvio)
 MDA
 (Vd. Mail Delivery Agent)
 Metacity, 106
 (Vd. Anche XFree86)
 MetaDir
 direttiva di configurazione di Apache, 211
 MetaSuffix
 direttiva di configurazione di Apache, 211
 MinSpareServers
 direttiva di configurazione di Apache, 199
 moduli
 (Vd. moduli del kernel)
 (Vd. moduli del kernel)
 Apache
 caricamento, 215
 personali, 215
 predefiniti, 214
 moduli del kernel
 introduzione, 281
 moduli del CD-ROM
 parametri, 282
 moduli Ethernet
 esempi, 291
 parametri, 287
 supporto schede multiple, 291
 moduli SCSI
 esempi, 287
 parametri, 284
 parametri dei moduli
 specificare, 281

tipi di, 281
 moduli di Apache HTTP Server, 214
 moduli di autenticazione
 (Vd. PAM)
 moduli NIC
 (Vd. moduli del kernel)
 moduli SCSI
 (Vd. moduli del kernel)
 moduli Ethernet
 (Vd. moduli del kernel)
 mouse
 utilizzo, xv
 MTA
 (Vd. Mail Transfer Agent)
 MUA
 (Vd. Mail User Agent)

N

named.conf
 (Vd. BIND)
 NameVirtualHost
 direttiva di configurazione di Apache, 213
 netfilter
 (Vd. iptables)
 Network File System
 (Vd. NFS)
 NFS
 client
 /etc/fstab, 264
 autofs, 264
 configurazione, 263
 opzioni di montaggio, 265
 introduzione, 259
 metodologia, 259
 portmap, 260
 risorse aggiuntive, 267
 documentazione installata, 267
 libri correlati, 268
 server
 file di configurazione, 261
 sicurezza, 266
 accesso host, 266
 permessi dei file, 267
 ntsysv, 66
 (Vd. Anche servizi)

O

- oggetti, dinamicamente condivisi
 - (Vd. DSO)
- OpenLDAP
 - (Vd. LDAP)
- OpenSSH, 133
 - (Vd. Anche SSH)
 - file di configurazione per, 136
- Options
 - direttiva di configurazione di Apache, 203
- Order
 - direttiva di configurazione di Apache, 203

P**PAM**

- definizione di, 115
- file di configurazione, 115
- file di configurazione di esempio, 118
- Kerberos e, 144
- moduli, 116, 116, 118
 - argomenti, 118
 - componenti, 116
 - creazione, 117
 - percorsi dei, 118
 - tipi, 116
- nomi di servizio, 115
- opzioni di controllo, 117
- pam_console
 - definizione di, 121
- password shadow, 119
- risorse aggiuntive, 121
 - documentazione installata, 122
 - siti Web utili, 122
- vantaggi di, 115
- pam_console
 - (Vd. PAM)
- parametri dei moduli
 - (Vd. moduli del kernel)
- password, 119
 - (Vd. Anche PAM)
- password shadow, 119
- PidFile
 - direttiva di configurazione di Apache, 198
- pool di slab
 - (Vd. /proc/slabinfo)
- portmap, 260
- rpcinfo, 260
- posizione dei file specifici di Red Hat Linux
 - /etc/sysconfig/, 24
 - /var/lib/rpm/, 24
 - /var/spool/up2date/, 24
- prefdm
 - (Vd. XFree86)
- processo di avvio, 59, 59

- caricamento a catena, 81
- caricamento diretto, 81
- fasi del, 59, 59
 - BIOS, 59
 - boot loader, 60
 - comando /sbin/init, 61
 - kernel, 61
 - shell EFI, 59
- per x86, 59
- Procmail, 230
 - configurazione, 231
 - regole, 232
 - azioni speciali, 234
 - condizioni speciali, 234
 - distribuzione, 233
 - esempi, 235
 - file di lock locali, 234
 - flag, 233
 - non distribuzione, 233
 - risorse aggiuntive, 237
- programmi
 - esecuzione all'avvio, 64
- protocollo SSH, 133
 - autenticazione, 135
 - caratteristiche di, 133
 - livelli di, 134
 - canali, 136
 - livello di trasporto, 135
 - pericoli per la sicurezza, 133
 - port forwarding, 138
 - protocolli non sicuri e, 139
 - richiesta per le connessioni remote, 139
 - sequenza di connessione, 134
 - X11 forwarding, 137
- ProxyRequests
 - direttiva di configurazione di Apache, 212
- ProxyVia
 - direttiva di configurazione di Apache, 212

R

rc.local
 modifica, 64
 READMEName
 direttiva di configurazione di Apache, 209
 Redirect
 direttiva di configurazione di Apache, 208
 rete
 comandi
 /sbin/ifdown, 170
 /sbin/ifup, 170
 /sbin/service network, 170
 configurazione, 166
 funzioni, 171
 interfacce, 166
 alias, 169
 clone, 169
 dialup, 167
 Ethernet, 166
 risorse aggiuntive, 171
 script, 165
 risoluzione dei problemi
 log degli errori, 206
 rpcinfo, 260
 runlevel, 65
 modifica all'avvio, 91
 modifica con GRUB, 85

S

sawfish, 106
 (Vd. Anche XFree86)
 ScoreBoardFile
 direttiva di configurazione Apache, 198
 script CGI
 al di fuori della direttiva ScriptAlias, 210
 esecuzione dei programmi esterni a cgi-bin, 202
 ScriptAlias
 direttiva di configurazione di Apache, 207
 Sendmail, 222
 alias, 225
 con UUCP, 224
 installazione predefinita, 223
 LDAP e, 226
 limiti, 223
 masquerading, 225
 modifiche comuni alla configurazione, 224
 risorse aggiuntive, 237
 scopo, 223
 spam, 225
 storia, 222
 server dei nomi
 (Vd. BIND)
 server dei nomi caching-only
 (Vd. BIND)

server dei nomi forwarding
 (Vd. BIND)
 server dei nomi master
 (Vd. BIND)
 server dei nomi root
 (Vd. BIND)
 server dei nomi slave
 (Vd. BIND)
 server proxy, 212, 212
 server Web non sicuro
 disabilitazione, 216
 ServerAdmin
 direttiva di configurazione di Apache, 201
 ServerName
 direttiva di configurazione di Apache, 202
 ServerRoot
 direttiva di configurazione di Apache, 198
 ServerSignature
 direttiva di configurazione di Apache, 207
 Services Configuration Tool, 66
 (Vd. Anche servizi)
 servizi
 configurazione con chkconfig, 66
 configurazione con ntsysv, 66
 configurazione con Services Configuration Tool,
 66
 SetEnvIf
 direttiva di configurazione di Apache, 213
 shadow
 (Vd. password)
 utility, 98
 shell EFI
 definizione di, 59
 (Vd. Anche processo di avvio)
 shell Extensible Firmware Interface
 (Vd. shell EFI)
 sicurezza
 configurazione, 213
 eseguire Apache senza, 215
 sistema
 chiusura, 79
 sistema X Window
 (Vd. XFree86)
 StartServers
 direttiva di configurazione di Apache, 199
 startx, 107
 (Vd. Anche XFree86)
 struttura
 comune, 19
 stunnel, 237
 suggerimenti
 come contattarci per inviarci informazioni, xvi
 sysconfig
 sottodirectory nella, 79
 sysctl
 configurazione con /etc/sysctl.conf/, 56

controllo con `/proc/sys/`, 56
 SysRq
 (Vd. tasto SysRq)
 SysV init
 (Vd. comando init)

T

tasto SysRq
 attivazione, 48
 definizione di, 48
 Timeout
 direttiva di configurazione di Apache, 198
 Tripwire
 applicazioni, 160
 tripwire, 160
 tripwire-check, 155
 twadmin, 158, 160, 160
 twinstall.sh, 160
 twprint, 155, 156, 160
 controllo dell'integrità
 comando tripwire --check, 155
 database
 aggiornamento, 157
 definizione di, 161
 inizializzazione di, 154
 diagramma di flusso di, 149
 file di configurazione, 160
 aggiornamento, 160
 file database, 160, 161
 file delle chiavi, 160
 file report, 160, 161
 firma dei, 160
 modifica, 152
 tw.cfg, 160, 161
 tw.pol, 160, 161
 twcfg.txt, 160
 twpol.txt, 160
 file di policy
 aggiornamento, 158
 modifica, 153
 funzioni di posta elettronica, 159
 prova, 159
 installazione di
 comando tripwire --init, 154
 configurare la personalizzazione, 152
 impostazione delle password, 154
 inizializzazione del database di Tripwire, 154
 installazione dell'RPM, 151
 script twinstall.sh, 154
 introduzione, 149
 report
 definizione di, 161
 generare, 155
 visualizzazione, 155

risorse aggiuntive, 161
 documentazione installata, 162
 siti Web utili, 162
 TypesConfig
 direttiva di configurazione di Apache, 205

U

UseCanonicalName
 direttiva di configurazione di Apache, 205
 User
 direttiva di configurazione di Apache, 201
 UserDir
 direttiva di configurazione di Apache, 204
 utenti, 93
 /etc/passwd, 93
 definizione di, 93
 directory HTML personali, 204
 standard, 93
 strumenti per la gestione di
 redhat-config-users, 93
 useradd, 93
 tipi di, 93
 UID, 93
 utility
 shadow, 98
 utility Apache APXS, 215
 utility di initscript, 66
 (Vd. Anche servizi)

V

VirtualHost
 direttiva di configurazione di Apache, 213

W

webmaster
 indirizzo e-mail per, 201
 window manager, 106
 (Vd. Anche XFree86)
 Windowmaker
 (Vd. wmaker)
 wmaker, 106
 (Vd. Anche XFree86)
 wrapper TCP
 controllo dell'accesso, 123
 /etc/hosts.deny, 123
 caratteri jolly, 124
 comandi della shell, 126
 espansioni, 126
 modelli, 124
 operatori, 125
 regole di formattazione, 124

- definizione di, 123
- modi di utilizzo, 123
- risorse aggiuntive, 131
 - documentazione installata, 131
 - siti Web utili, 131
- vantaggi, 123
- xinetd, 127

X

X

- (Vd. XFree86)

X.500

- (Vd. LDAP)

X.500 Lite

- (Vd. LDAP)

XFree86

- ambienti desktop, 107

client X

- ambienti desktop, 107

- GNOME, 107

- KDE, 107

- sawfish, 106

- window manager, 106

- wmaker, 106

- xinit, 106

display manager di X

- definizione di, 108

- gdm, 108

- kdm, 108

- script prefdm, 108

- xdm, 108

file di configurazione, 102

- directory /etc/X11/, 102

- opzioni incluse nei, 102

- font, 109

- xfst, 109

font server X

- aggiungere font, 110

- configurazione, 110

- xfst, 102

introduzione, 101

- risorse aggiuntive, 111

- documentazione installata, 111

- libri correlati, 112

- siti Web utili, 111

runlevel, 107

- 3, 107

- 5, 108

server X, 101

- caratteristiche di, 102

- XFree86, 102

X client, 101

- mwn (Metacity), 106

- twm, 106

- window manager, 106

XFree87

utility

- X Configuration Tool, 101

xfst

- (Vd. XFree86)

xinetd, 127

- (Vd. Anche wrapper TCP)

- collegamento, 130

- controllo dell'accesso con, 129

- file di configurazione, 127

- /etc/xinetd.conf, 127

- directory /etc/xinetd.d/, 128

- per prevenire attacchi DoS, 128

- ridirezionamento delle porte, 130

xinit

- (Vd. XFree86)



Colophon

I manuali ufficiali di Red Hat Linux sono scritti in formato SGML v4.1. I formati HTML e PDF sono creati mediante il foglio di stile DSSSL e gli script jade wrapper personalizzati.

Elementi quali note, suggerimenti e simili sono stati creati da Marianne Pecci <goddess@ipass.net>. Possono essere ridistribuiti con i permessi scritti di Marianne Pecci e Red Hat, Inc..

Il Product Documentation Team di Red Hat Linux comprende:

Sandra A. Moore — Primary Writer/Maintainer della *Official Red Hat Linux x86 Installation Guide*; Contributing Writer per la *Official Red Hat Linux Getting Started Guide*

Tammy Fox — Primary Writer/Maintainer della *Official Red Hat Linux Customization Guide*; Contributing Writer per la *Official Red Hat Linux Getting Started Guide*; Writer/Maintainer di fogli di stile e script personalizzati del manuale

Edward C. Bailey — Contributing Writer per la *Official Red Hat Linux x86 Installation Guide*

Johnray Fuller — Primary Writer/Maintainer della *Official Red Hat Linux Reference Guide*; Co-writer/Co-maintainer per la *Official Red Hat Linux Security Guide*

John Ha — Primary Writer/Maintainer della *Official Red Hat Linux Getting Started Guide*; Co-writer/Co-maintainer per la *Official Red Hat Linux Security Guide*

