

IPFIRE-wall

Guida all'installazione e alla prima esecuzione di un

Filtro di Pacchetti Alternativo

Giacomo Strangolino

Settembre 2006

<http://www.giacomos.it/ipfire/>

Introduzione¹.

Cos'è *IPFIRE-wall*.

IPFIRE-wall è un filtro di pacchetti di rete in grado anche di effettuare la traduzione degli indirizzi (*Network Address Translation*) e il cosiddetto *IP Masquerading*.

Esso analizza il traffico di rete che attraversa un computer collegato ad una *internet* e, in quanto *filtro di pacchetti*, consente o nega loro l'accesso al sistema e quindi ai servizi che essi rappresentano. Qualsiasi utente di un computer *Linux* può beneficiare del filtro di pacchetti e personalizzare *IPFIRE-wall* con le proprie regole, al fine di proteggersi nel modo migliore.

In quanto capace di *Network Address Translation (NAT)*, può venire utilizzato per manipolare gli indirizzi che caratterizzano ciascun pacchetto di rete per cambiarne la destinazione o la sorgente. Questa proprietà avanzata può essere gestita solamente dall'amministratore, che, conoscendo i servizi disponibili nella rete e la sua configurazione, può usare *IPFIRE-wall* per riorganizzare, reindirizzandoli, determinati flussi di dati al fine di ottenere maggiore sicurezza o efficienza nella propria *internet*.

L'*IP Masquerading*, termine introdotto dal celeberrimo software *firewall iptables*, non è altro che una forma di *NAT* in cui i pacchetti in uscita assumono come indirizzo sorgente quello del dispositivo di rete per mezzo del quale lasciano la macchina locale. Il *mascheramento* degli indirizzi consente una semplice configurazione di una *lan*² che utilizza un *gateway* per collegarsi ad *Internet*. Un *gateway* è un computer con due interfacce di rete, la prima collegata ai computer della rete interna, la seconda collegata ad *Internet*, ad esempio attraverso un modem *ADSL* o *56k*. Le macchine della rete interna "escono" verso *Internet* proprio attraverso il *gateway*, e ricevono le risposte da esso stesso, ignorando il percorso effettuato dai propri pacchetti al di fuori della *lan*. Anche la configurazione del *mascheramento degli indirizzi* è prerogativa dell'amministratore.

IPFIRE-wall è stato scritto interamente in linguaggio *C* e presenta un'interfaccia di gestione delle regole del *filtro di pacchetti* a linea di comando, ovvero non ancora integrata nell'interfaccia grafica del proprio sistema. È stato studiato per essere utilizzato con semplicità anche da chi non è "esperto" di reti di computer o della struttura e implementazione dell'architettura di *internetworking* di un sistema *Linux*.

Alcune caratteristiche peculiari distinguono *IPFIRE-wall* dagli altri software di amministrazione della rete tipici dei sistemi *Linux*:

- ogni utente, anche non amministratore, può configurare le proprie regole, senza tuttavia alterare le scelte operate da quest'ultimo;

1 Questo paragrafo presenta alcuni concetti riguardanti le *reti di calcolatori*. Essi potrebbero risultare poco chiari per chi si accingesse per la prima volta a studiare l'*internetworking* o ad utilizzare il software descritto in questo manuale. Si rimanda a dei testi specifici per approfondire o chiarire gli argomenti accennati, come ad esempio *Andrew S. Tanenbaum, Reti di Computer*.

2 *Local Area Network*, rete locale.

Cos'è IPFIRE-wall.

- è semplice bloccare dei siti web inserendo il loro indirizzo in un semplice file di testo;
- l'interfaccia utente non solo consente la configurazione delle regole e in generale la gestione del firewall in tutti i suoi aspetti, ma fornisce un ottimo sistema per imparare il funzionamento delle reti di computer e in particolare la loro implementazione nel sistema Linux. Ogni verdetto sui pacchetti in transito è rappresentato sulla console di IPFIRE-wall in colori diversi e ogni regola possiede un breve nome per consentire la sua identificazione all'interno di un insieme numeroso e variegato di regole;
- l'inserimento e la rimozione delle regole sono molto semplici perché sono guidati da una procedura assistita che chiede all'utente il valore da dare ad ogni parametro;
- l'algoritmo di filtro è sostanzialmente diverso da quello normale che applica la regola trovata nell'elenco e poi esce: esistono un insieme di regole di permesso esplicito e uno di regole di negazione esplicita.

Prima vengono esaminate le regole di negazione esplicita: i pacchetti corrispondenti vengono immediatamente scartati;

in seguito viene cercato un consenso all'interno delle regole di permesso: in caso di successo, il pacchetto corrispondente viene accettato;

se non c'è riscontro in alcun insieme di regole esplicite viene applicata una delle due possibili politiche predefinite: negazione o consenso³.

Questa scelta, oltre a rappresentare un'alternativa immediata e naturale, fornisce all'amministratore la certezza che le sue regole di negazione esplicita saranno sempre le prime norme ad essere applicate. Un utente normale, non potendo rimuovere le prescrizioni dell'utente privilegiato, non potrà in alcun modo garantire un permesso a un pacchetto esplicitamente negato;

- la stampa a video dei pacchetti filtrati può essere sospesa per diminuire il carico di lavoro imposto da IPFIRE-wall;
- altre caratteristiche di minore impatto che l'utente scoprirà utilizzando il software...

Seguiranno una guida all'installazione di IPFIRE-wall e una breve trattazione sul suo primo avvio, sia per l'amministratore che per l'utente normale. Si augurano all'interessato una buona lettura e un buon divertimento.

Cosa non è IPFIRE-wall.

IPFIRE-wall costituisce sostanzialmente un progetto didattico, atto a fornire ai programmatori alcuni esempi di sviluppo di codice in spazio *kernel* e in spazio utente⁴.

³ Si veda la nota *otto* per apprendere come cambiare o impostare la politica predefinita del firewall.

⁴ Si veda la documentazione *web* per ottenere un elenco degli argomenti affrontati nello sviluppo della parte *kernel* e *utente*.

Non è stato progettato per l'utilizzo su *server* di rete⁵, o in ambienti in cui sia richiesto un livello di affidabilità che l'immatùrità di questo software non può garantire, essendo molto “giovane” e poco testato.

È invece stato pensato per l'utilizzo *desktop* e per aiutare l'utente nella comprensione del funzionamento delle reti di computer, nonché per fornire un livello di protezione facilmente configurabile anche da parte di un utilizzatore non esperto. Le stampe a video dei pacchetti filtrati da *IPFIRE-wall* sono molto istruttive e rappresentano in modo semplice e immediato l'applicazione delle regole sui pacchetti in transito all'interno della propria macchina.

⁵ Si legga il paragrafo “*Attenzione*” appena di seguito riportato.

Attenzione.

Attenzione.

Il software *IPFIRE-wall* viene fornito *così com'è*, senza alcuna garanzia di funzionamento e senza particolari obiettivi a livello di *performance* o *implementativi*.

In particolare, l'autore sottolinea che il motore del filtro di pacchetti viene eseguito in *spazio kernel*. Questo implica che *un errore di programmazione anche banale può causare il blocco di tutta la macchina, la perdita di tutto il lavoro non salvato e persino la compromissione di dati salvati*. Pertanto, egli non assume la responsabilità di eventuali *danni* dovuti all'utilizzo di *IPFIRE-wall*.

Ciò che si può affermare è che il software è in esecuzione da un anno all'interno di un ambiente di rete molto attivo, in cui il lavoro di ogni giorno si basa sull'esecuzione di procedure e interfacce che risiedono su sistemi remoti, e i pacchetti scambiati e quindi in circolazione attraverso il *firewall* sono dell'ordine di qualche *milione* al giorno. *IPFIRE-wall* viene eseguito attualmente su tre macchine ad *Elettra* (<http://www.elettra.trieste.it>), tra le quali, ovviamente, quella dell'autore.

Qualche riserva va espressa a proposito del *NAT*, che non è stato possibile testare accuratamente e approfonditamente in quel contesto, ma unicamente all'interno di una piccola rete. Si confida pertanto nella disponibilità di qualche *tester* delle funzionalità di *NAT*.

Il software è libero, riutilizzabile, modificabile, a patto che, in ogni occasione in cui esso venga sfruttato, sia presente una nota che ne riporti l'autore e la data di sviluppo (nota di *copyright*).

1. Installazione del software.

1.1 Ottenere il pacchetto per l'installazione.

Il pacchetto per l'installazione di *IPFIRE-wall* si può scaricare dal sito Internet <http://www.giacomos.it/ipfire> seguendo il link per il download, ovvero direttamente dal server presso cui il progetto è ospitato: <http://www.sourceforge.net/projects/ipfire-wall>.

Una volta completato il download del pacchetto, esso deve essere scompattato in una cartella locale del proprio computer, cosicché sia possibile accedere agli strumenti progettati per l'installazione del software.

1.2 Installazione di IPFIRE-wall.

1.2.1 Estrazione dei file compressi.

IPFIRE-wall è facilmente installabile sul vostro computer grazie ad una procedura automatica che dovrebbe portare a termine la costruzione del programma eseguibile, dei moduli software necessari al sistema per far funzionare il *firewall*, nonché la copia dei file nei punti corretti perché immediatamente dopo l'esecuzione dell'*installer* il programma sia già utilizzabile in modo immediato e con attivate le sue funzionalità predefinite ce consentono una protezione di base della vostra postazione di lavoro.

Per procedere all'installazione dunque, ci si sposti nella cartella in cui si è deciso di salvare il file scaricato dalla rete *Internet* e lo si estragga localmente cliccando con il tasto destro sull'icona di *IPFIRE-wall-0.98.tar.gz*⁶ oppure *IPFIRE-wall-0.98.tar.bz2*, a seconda del tipo di file compresso che avete trovato disponibile presso il sito "*sourceforge.net*". Scegliete quindi l'opzione "*Estrai*" e quindi "*Estrai qui*".

6 In questo testo si riporta il nome del file nella versione corrente, la 0.98. Ovviamente il nome può cambiare con le versioni successive, così come il tipo di file compresso, che potrà essere *.tar.bz2* o *.tar.gz*.

1.2.1 Estrazione dei file compressi.

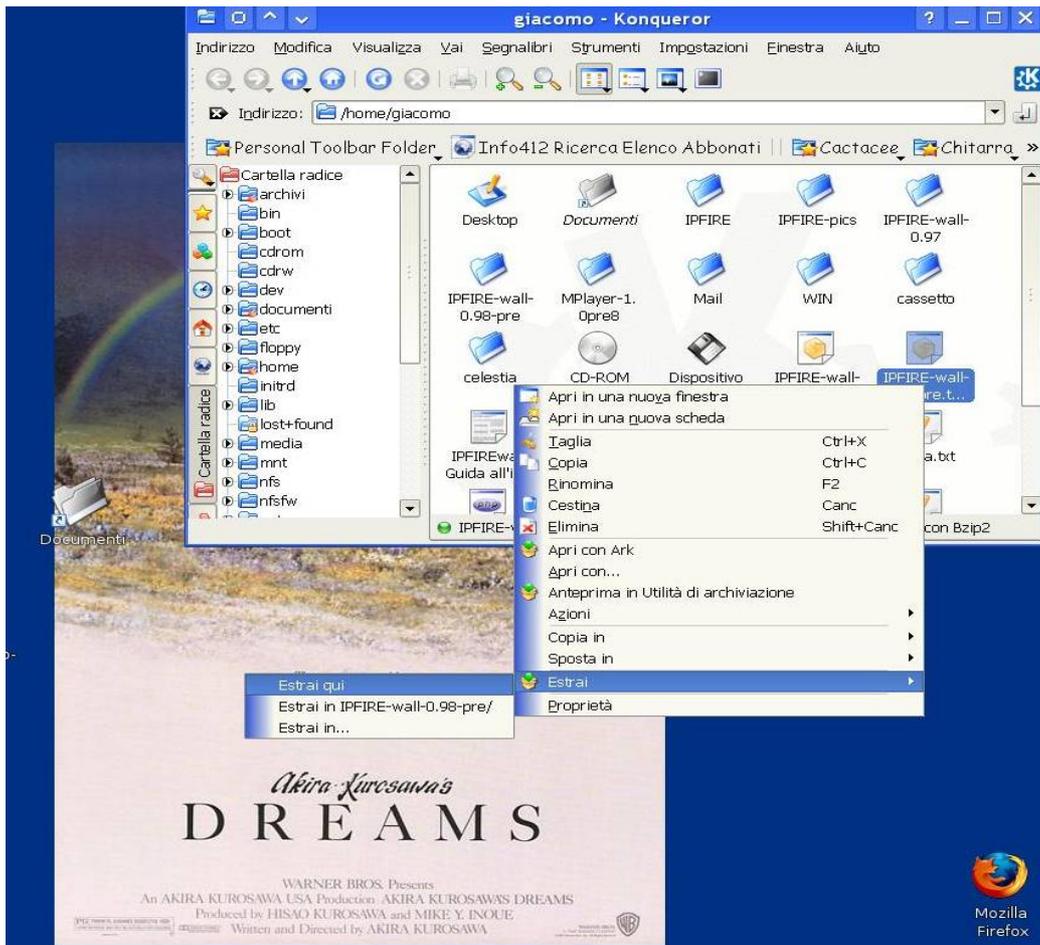


figura 1: estrazione del software dalla cartella compressa scaricata dal web nell'ambiente kde.

La figura sopra rappresenta quanto descritto, e questo vale per gli utenti *kde*. Per coloro che utilizzassero l'ambiente *Gnome*, la procedura è del tutto analoga, ed è rappresentata nella *figura 2*.

1.2.1 Estrazione dei file compressi.



figura 2: estrazione del software dalla cartella compressa scaricata dal web nell'ambiente gnome.

Ovviamente gli utenti più esperti possono estrarre il software da linea di comando collocandosi all'interno della *directory* in cui si trova il file compresso e digitando dalla *shell* i comandi:

```
tar -xzf IPFIRE-wall-0.98.tar.gz
```

oppure

```
tar -xjf IPFIRE-wall-0.98.tar.bz2
```

a seconda del tipo di file scaricato.

1.2.2 Installazione automatica⁷.

L'installazione di *IPFIRE-wall* costituisce una procedura molto semplice; essa va eseguita da *ogni* utente che desideri utilizzare il programma.

Una condizione fondamentale per l'uso di *IPFIRE* da parte degli utenti di una workstation Linux è rappresentata dalla necessità che l'utente *root* (amministratore) esegua un'installazione del software

⁷ Si veda la sezione più in basso "Installazione manuale" (par. 1.2.3) se si lavora su sistemi *BDS-like* come *Slackware Linux*.

affinché gli utenti *non privilegiati* possano avviare la loro istanza del *firewall*.

Pertanto la trattazione proseguirà descrivendo l'installazione da utente amministratore e quella complementare da utente normale. In ogni caso, sarà necessario che l'utente apra un *terminale* (o *shell*) all'interno della cartella creata al paragrafo precedente dalla procedura di *estrazione dei file compressi*, spostandosi nella *directory* “*ipfi*” di “*IPFIRE-wall-0.98*”, e quindi dal menù “*strumenti*” della finestra di navigazione dei file, scelga “*Apri Terminale*”, come in *figura 3*.

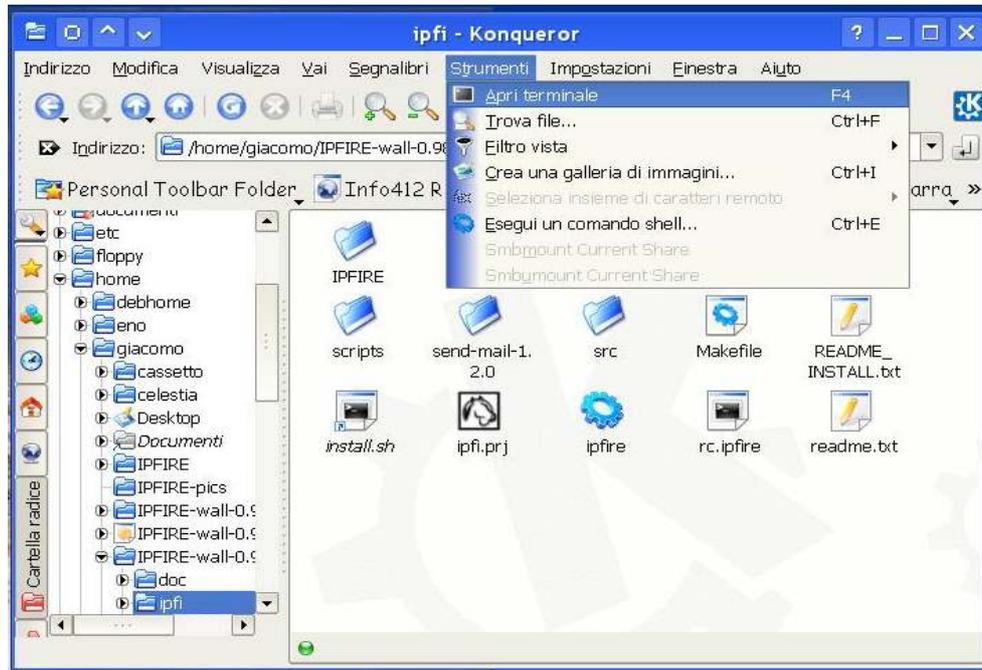


figura 3: apertura di un terminale all'interno della cartella corrente nell'ambiente kde.

Gli utenti *Gnome* invece, dovranno aprire un terminale dal menù “*Applicazioni*” del pannello (l'analogo di “*Start*” in *Windows*), scegliere “*Accessori*”, e quindi “*Terminale*”.

Di norma il terminale aperto in modo siffatto è collocato nella cartella *home* dell'utente corrente, e quindi è necessario che quest'ultimo si sposti nella *directory* in cui si trovano i file di *IPFIRE*. Facendo riferimento alla *figura 2*, in cui *IPFIRE* veniva estratto in */home/giacomo*, si dovrà digitare:

```
cd IPFIRE-wall-0.98/ipfi
```

seguito dal tasto *invio*, come sempre per ogni comando *shell*.

A questo punto, la *shell* aperta dovrebbe mostrare una riga simile, a seconda della posizione in cui avete estratto i file, ma in ogni caso dovrebbe indicare la *directory* “*ipfi*” come quella in cui vi trovate (*figura 4*).

NOTA: per l'uso di *IPFIRE-wall*, selezionate sempre lo *sfondo nero con i caratteri bianchi* sul vostro terminale preferito, in quanto le stampe dei messaggi sono a colori e l'effetto migliore si

1.2.2 Installazione automatica7.

ottiene solo in tali condizioni. Per cambiare i colori della console di *Kde*, ovvero “*konsole*”, si vada nel menù “*Impostazioni*”, si scelga “*Schema*” e quindi “*Bianco su Nero*”.

Nel terminale *Gnome*, “*Gnome-terminal*”, si deve cliccare invece sulla voce “*Modifica*” del menù, quindi scegliere “*Profili*”. Si aprirà una finestra in cui si cliccherà su “*Nuovo*”. Si dia un nome al nuovo profilo, ad esempio “*Firewall*”. A questo punto comparirà una finestra di modifica della configurazione con diverse schede. Potete personalizzare il terminale *gnome* nel modo preferito, e in particolare, nella scheda “*Colori*”, potete scegliere lo sfondo nero e il carattere bianco. Una volta soddisfatti, cliccate su “*Chiudi*” e le modifiche verranno salvate automaticamente. Chiudete anche la finestra “*Profili*” e ora dal menù “*Terminale*”, scegliendo “*Cambia Profilo*”, potrete caricare le impostazioni appena salvate.

Una procedura più veloce che vi consentirà di salvare il profilo come predefinito, e quindi senza doverlo caricare ogni volta da “*Cambia Profilo*” dal menù “*Terminale*”, sarà costituita dallo scegliere direttamente l'opzione “*Profilo Attuale*” dal menù “*Modifica*”, e quindi cambiare i colori che verranno di norma applicati all'avvio di *ogni* terminale *gnome*.

Anche per la “*konsole*” di *Kde* è certamente possibile salvare uno schema di colori predefinito, optando, dal menù “*Impostazioni*”, per la voce “*Configura konsole...*”, e quindi selezionando lo schema *Bianco su Nero*.



figura 4: il terminale, se aperto correttamente, si deve trovare nella directory “ipfi” all’interno di “IPFIRE-wall-0.98”.

1.2.2.1 Installazione da root.

Una volta collocatosi nella *directory* “ipfi” all’interno della cartella di *IPFIRE-wall-0.98*, l’utente che desidera installare il filtro di pacchetti nel sistema dovrà assumere i privilegi di *amministratore*.

Se ciò non avviene, non sarà possibile né installare, né eseguire *IPFIRE-wall* in alcun modo. Il software deve infatti integrarsi nel *kernel*⁸ *Linux* installando i moduli di basso livello necessari al filtraggio dei pacchetti di rete in base alle regole imposte dagli utenti.

Per assumere l’identità dell’utente amministratore, si digiti il comando

SU -

seguito da *invio*⁹.

Si dovrà inserire la password di *root* affinché si possa diventare l’*amministratore*, e godere dei privilegi necessari per l’installazione.

Essendo stata eseguita questa operazione con successo, è necessario che l’utente avvii l’installazione vera e propria, nel modo descritto al punto successivo e del tutto analogo per l’utente *root* e per quelli non privilegiati.

1.2.2.2 Installazione da utente non privilegiato.

Il procedimento descritto in questo paragrafo deve essere seguito da *tutti gli utenti che desiderano utilizzare IPFIRE-wall*, in primis dall’utente *root*.

Trovandosi sempre nella *directory* “ipfi” all’interno della cartella di *IPFIRE-wall-0.98*, l’utente

⁸ Il *Kernel* di un sistema operativo rappresenta il suo cuore ed è lo strato software che *interagisce* con i dispositivi del vostro computer, ad esempio la scheda video, la scheda di rete, il processore, la memoria, e così via.

⁹ D’ora in poi non si specificherà più “*seguito da invio*”, perché ogni comando *shell* viene impartito solo in seguito alla conferma rappresentata dal tasto *invio*.

1.2.2.2 Installazione da utente non privilegiato.

digiti alla *shell*:

./install.sh

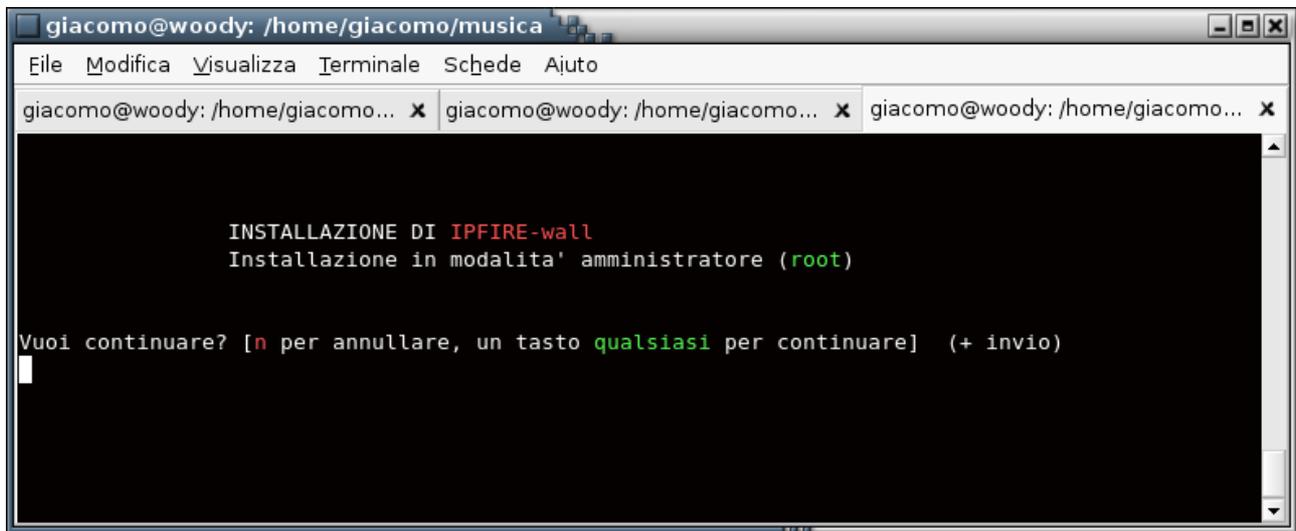


figura 5: la schermata di avvio del programma di setup. Il tasto *n* seguito da invio annulla la procedura, un qualsiasi altro tasto invece, seguito da invio, esegue l'installazione di *IPFIRE-wall*.

Lo *script* di installazione esegue dei comandi in modo automatico per la compilazione¹⁰ dell'interfaccia di *IPFIRE-wall*, atta a consentire all'utente di inserire le proprie regole e monitorare la rete, e dei moduli del *kernel*, necessari al sistema per applicare le norme degli utenti.

L'installatore cerca di individuare la lingua in uso dall'utente leggendo la variabile d'ambiente¹¹ *LANG=*, che di norma è l'italiano se l'utilizzatore del computer l'ha scelta come lingua per l'interfaccia grafica.

Per cambiare la lingua rilevata automaticamente dallo *script* di installazione, si veda la nota numero undici a piè di pagina.

10 La compilazione consente la generazione del programma eseguibile a partire dal codice sorgente scritto dall'autore del programma *IPFIRE-wall*. Questo passo conduce alla creazione del programma *adatto* ad essere eseguito sopra le caratteristiche del vostro sistema e del vostro *kernel* linux.

11 Una variabile d'ambiente è una coppia *VARIABILE=VALORE* che viene impostata spesso in modo automatico dalla *shell* linux all'avvio della *shell* stessa. In particolare, se la lingua in uso è l'italiano, allora la variabile *LANG* dovrebbe essere settata a *it_IT*.

Per verificarlo si digiti:

echo \$LANG

alla console.

Se il risultato non è quello aspettato, oppure se si desidera l'inglese come lingua per l'installazione di *IPFIRE-wall*, è sufficiente cambiare la variabile d'ambiente con il comando *export*:

export LANG=it_IT

oppure

export LANG=en_EN

per l'inglese.

Si leggano *attentamente* le informazioni stampate a video durante l'installazione. Verrà chiesta conferma in occasione delle azioni principali dell'installatore automatico. Di norma è sufficiente rispondere con il tasto “s” (sì) seguito da *invio* per procedere alle azioni predefinite.

I passi che osserverete saranno costituiti da:

a) *copia dei file delle impostazioni nella propria home;*

b) *compilazione dell'interfaccia di IPFIRE-wall;*

c) *nel caso di utente root, compilazione e installazione dei moduli del kernel e copia dei file di inizializzazione nelle cartelle di sistema.*

Alla conclusione, si rilegga completamente l'insieme dei messaggi emessi nella fase di *setup* di *IPFIRE-wall*.

In caso di errore, si legga con attenzione la sezione 1.3: *Prerequisiti per l'installazione di IPFIRE-wall*.

È infatti necessario un ambiente di sviluppo installato sulla propria macchina.

1.2.3 Installazione manuale.

L'utente esperto o particolarmente curioso può scegliere di compilare *IPFIRE-wall* e copiare manualmente i file di configurazione nelle posizioni corrette.

1.2.3.1 Compilazione e installazione dei moduli del kernel.

Per compilare i sorgenti del *kernel* di *IPFIRE*, è necessario spostarsi nella cartella “*kernel*” all'interno della *directory* principale. Sarà quindi sufficiente dare il comando

make

per compilare i moduli e

make install

per installarli nel sistema in uso.

La compilazione manuale dei sorgenti del *kernel* consente anche di leggere eventuali problemi o avvertimenti in fase di costruzione dei moduli stessi. Si legga la sezione *Prerequisiti per l'installazione di IPFIRE-wall* in caso di errori o al fine di approfondire l'argomento e preparare in modo ottimale il contesto d'uso di *IPFIRE-wall*.

Una volta compilati e installati i moduli del *kernel*, è possibile caricare gli stessi digitando dalla linea di comando:

modprobe ipfi

Si faccia attenzione che il caricamento dei moduli con il comando sopra riportato può interrompere

1.2.3.1 Compilazione e installazione dei moduli del kernel.

la comunicazione di rete, anche quella di *loopback*¹². Questa situazione permane fintanto che il modulo non viene rimosso oppure fino a quando non viene avviata un'istanza dell'interfaccia utente di *IPFIRE* che inserisca delle regole di permesso per attivare la comunicazione¹³.

È pertanto consigliabile avviare e arrestare il *firewall* con le utilità appositamente studiate allo scopo e di seguito descritte.

1.2.3.2 Compilazione dell'interfaccia utente e installazione dei file.

Per compilare l'interfaccia utente di *IPFIRE-wall* è necessario spostarsi nella cartella “*ipfi*” e digitare

make

Se la compilazione avviene senza problemi, nella *directory* “*ipfi*” stessa verrà creato l'eseguibile chiamato *ipfire*, che andrà copiato, da *root*¹⁴, in una cartella di sistema globalmente visibile, come ad esempio */usr/local/bin* oppure */usr/bin*, o, ancora, */sbin*.

Il programma di installazione automatico copia l'eseguibile in */sbin* e successivamente crea un collegamento simbolico in */usr/local/bin*, che rende visibile agli utenti normali il programma nascondendo al contempo il concetto che trattasi di un file di sistema.

Se un utente desiderasse utilizzare il programma *SMTPclient* distribuito assieme a *IPFIREwall* e scritto da *Ralf S. Engelschall* e *Davide Libenzi*, sappia che anche esso va compilato e copiato opportunamente in una cartella di sistema, come ad esempio */usr/local/bin*.

12 Spesso molti programmi utilizzano un'interfaccia detta di *loopback* per instaurare una comunicazione di rete anche all'interno dello stesso computer. Ad esempio il sistema grafico *X Window System*, in uso nei sistemi *Linux*, si basa su un'infrastruttura di rete di tipo *client-server* in esecuzione sul computer locale.

13 In realtà è possibile specificare dei parametri al momento del caricamento del modulo nel *kernel* in modo tale che la *politica* predefinita di *IPFIRE* sia quella di concedere il permesso a comunicazioni prive di una regola esplicita.

Per ottenere questo risultato il comando

modprobe ipfi

sopra riportato dovrebbe diventare

modprobe ipfi policy="accept".

In alternativa, dopo aver impartito

modprobe ipfi,

si può dire a *IPFIRE-wall* di applicare il permesso a comunicazioni prive di una regola esplicita digitando

echo "accept" > /proc/IPFIRE/policy,

sempre da *root*.

14 Quando si dice “*da root*”, si intende che un utente senza privilegi dovrà digitare il comando

su

e inserire la *password* di *root* al fine di non avere limiti nell'esecuzione di un'azione.

Si tenga sempre presente che un errore commesso quando si hanno privilegi di *superutente* può non essere rimediabile e il sistema può risultarne danneggiato.

Per compilare il *mailer SMTPclient*, sempre dalla directory “*ipfi*”, si impartisca il comando:

`gcc -o mailer/SMTPclient send-mail-1.2.0/*.c`

A tal punto, nella cartella “*mailer*”, si troverà l'eseguibile “*SMTPclient*”, pronto per l'uso.

Il comando di copia

`cp mailer/SMTPclient /usr/local/bin`

da root renderà quindi disponibile il comando per tutti gli utenti.

I file necessari al funzionamento di *IPFIRE-wall* sono presenti infine nella cartella *IPFIRE* all'interno della cartella principale *IPFIRE-wall-0.98*.

In particolare, *ogni utente che intenda usare IPFIRE-wall dovrà avere nella propria home una cartella chiamata “IPFIRE”, contenente i file:*

- *allowed*: contiene le regole di permesso esplicito;
- *blacklist*: contiene le regole di negazione esplicito;
- *blacksites*: contiene l'elenco dei *siti web* bloccati;
- *translation*: contiene le regole di *NAT*¹⁵, necessario solo all'utente *root*;
- *firehelp*: contiene l'help in linea;
- *options*: contiene le opzioni generali di configurazione di *IPFIRE-wall*.

Inoltre, serviranno le cartelle:

- *languages*, con i file della traduzione nella lingua desiderata, per l'italiano serve il file “*it*”;
- *mailer*, contenente il file di configurazione del mailer ed eventualmente l'eseguibile da lanciare nel caso in cui l'amministratore abbia deciso di non fornire un'installazione visibile a tutto il sistema.

Senza i file sopra descritti *IPFIRE-wall* non funzionerà correttamente o si rifiuterà di avviarsi.

L'esecuzione dell'installazione automatica copia anche i file di esempio nella lingua corretta che si trovano nella medesima cartella *IPFIRE* nella *directory* principale.

Nella cartella “*ipfi*” si trova anche uno *script* di avvio automatico di *IPFIRE-wall*. Esso è stato studiato per essere inserito nelle *directory* di avvio del sistema (*init*), che sono diverse a seconda della distribuzione in uso. Ad esempio, in un sistema *debian GNU/Linux*, *rc.ipfire* andrebbe copiato nella cartella */etc/init.d*, e andrebbero creati dei link simbolici nelle cartelle */etc/rc2.d*, */etc/rc0.d*, */etc/rc3.d* e */etc/rc6.d*.

In un sistema *Slackware* invece, è sufficiente copiare *rc.ipfire* in */etc/rc.d* e aggiungere una riga opportuna nel file */etc/rc.local*:

`/etc/rc.d/rc.ipfire start`

¹⁵ *Network Address Translation*, traduzione degli indirizzi di rete.

1.2.3.2 Compilazione dell'interfaccia utente e installazione dei file.

Inoltre, per lo spegnimento, si devono aggiungere al file `/etc/rc.d/rc.K` le righe:

```
if [ -x /etc/rc.d/rc.ipfire ]; then  
  ./etc/rc.d/rc.ipfire stop  
fi
```

Queste azioni automatizzano l'avvio di *IPFIRE-wall* all'avvio e allo spegnimento del computer, garantendo il caricamento delle regole dell'utente *root* quando il pc viene acceso e assicurando la rimozione del modulo dal kernel allo spegnimento. Le ultime nozioni fornite sono rivolte ad un pubblico esperto e quindi non vengono ulteriormente approfondite. Si tenga presente a proposito che l'*installer* automatico provvede a creare i collegamenti sopra descritti in un sistema *debian-like*, ovvero in tutti i sistemi in cui l'inizializzazione avviene per mezzo di *script* inseriti nelle cartelle `/etc/rc0.d ... /etc/rc6.d`.

Nei sistemi di tipo *Slackware* invece (*BSD-like*), sarà necessario scrivere a mano le righe sopra riportate nei file `/etc/rc.d/rc.local` e `/etc/rc.d/rc.K`, ma ciò non costituirà certo una difficoltà per gli utenti di una siffatta distribuzione.

1.2.3.3 Compilazione dell'analizzatore dei log, *analyzer*.

Sempre all'interno della cartella “*ipfi*”, ci si sposti nella sottodirectory “*analyzer*” se si desidera testare l'utilità di analisi dei log¹⁶ di *IPFIRE-wall*. La compilazione avviene come di solito digitando il comando

```
make
```

Basterà eseguire il programma *analyze* dalla *directory* corrente per l'utilizzo dell'analizzatore. L'installatore automatico non compila né installa l'analizzatore di log.

1.2.3.3 Conclusioni sull'installazione.

Conclusa l'installazione, si leggano attentamente i file *readme* nella *directory* “*ipfi*”, che riportano sempre delle informazioni molto importanti sull'utilizzo e le precauzioni da prendere quando si utilizza il software.

In particolare, il file *readme.txt* riporta la storia delle versioni di *IPFIRE-wall* nel tempo, e le correzioni apportate al software attraverso le successive edizioni.

Eventuali problemi di compilazione dei sorgenti o di installazione del pacchetto possono essere risolti leggendo *con cura* la sezione successiva, o, in caso contrario, contattando l'autore via email all'indirizzo delleceste@gmail.com.

¹⁶ *IPFIRE-wall* scrive su un file delle informazioni su quanto esegue come filtro di rete. Questo file è detto file di *log*.

1.3 Prerequisiti del sistema per l'installazione di IPFIRE-wall.

1.3.1 Compilazione dei sorgenti.

Per compilare i file sorgenti di *IPFIRE-wall*, ma in generale di tutto il codice che viene distribuito nella forma di file sorgente *c*, è necessario avere installato sul proprio PC un ambiente di sviluppo composto dal compilatore *c* (il *gcc*, *Gnu C Compiler*), e dalle librerie *c*: le *glibc*.

A seconda della distribuzione Linux in uso, installare l'ambiente di sviluppo può essere diverso dipendentemente dei casi.

Ad esempio, per quanto riguarda la distribuzione *debian GNU/Linux*, dovrebbe essere sufficiente impartire da *root* il comando

```
apt-get install gcc libc6
```

A questo punto dovrebbe essere stato installato il contesto per eseguire il setup di *IPFIRE-wall*.

Per quanto riguarda la compilazione dei sorgenti del *kernel* invece, è necessario disporre di qualcosa in più:

- *i sorgenti del kernel*;
- *l'abilitazione del supporto per i moduli del kernel e del supporto di netfilter*.

1.3.2 Ottenere e installare i sorgenti del kernel.

Per ottenere i sorgenti del *kernel* Linux, è necessario scaricarli dal sito ufficiale <http://www.kernel.org>, e scompattarli nel modo descritto in precedenza per i sorgenti di *IPFIRE-wall*.

I sorgenti del *kernel* di norma vengono scompattati nella cartella */usr/src/*.

Installare i sorgenti del *kernel* in questo modo comporta in genere la necessità di *ricompilare* il *kernel* stesso per essere successivamente in grado di *compilare IPFIRE-wall*.

Non ci soffermeremo in questa sede a descrivere la procedura di ricompilazione del *kernel*, essendo trattata approfonditamente altrove nel *web* e richiedendo comunque un minimo di conoscenza del sistema *Linux*.

Ad ogni modo, se disponete di una distribuzione installata da cd rom, dovrete trovare nello stesso supporto il pacchetto con i sorgenti del *kernel* che state usando. Il pacchetto in oggetto viene spesso chiamato *kernel-headers* o *kernel-sources*.

Installare siffatto pacchetto è del tutto analogo all'installazione di ogni altro software e il metodo è tipico della distribuzione in uso. Ad esempio per *debian* il pacchetto si chiama *linux-kernel-headers*, per cui il comando, da *root*,

```
apt-get install linux-kernel-headers
```

dovrebbe essere quello giusto.

1.3.2 Ottenere e installare i sorgenti del kernel.

Il pacchetto con i *kernel-headers* dovrebbe avere già certamente configurati il supporto ai moduli e quello a netfilter. In ogni caso, le voci che vanno abilitate nel menù di configurazione del *kernel*¹⁷, sono le seguenti:

- *Loadable module support* -> *Enable loadable module support* e *Module unloading*;
- *Networking* -> *Networking Support* -> *Networking Options* -> *Network Packet Filtering*.

Non sarà necessario, a parte l'opzione generale *Network Packet Filtering*, alcun altro supporto all'interno del menù *Network Packet Filtering* (ovvero *iptables*¹⁸).

Una volta installati e configurati correttamente i sorgenti del *kernel*, non dovrebbero esserci difficoltà nel compilare *IPFIRE-wall*.

1.3.3 Versioni del *kernel* supportate.

IPFIRE-wall è stato compilato e testato su numerosi *kernel* della serie 2.6, a partire dalla versione **2.6.12** fino alla **2.6.18** (versione di *IPFIRE-wall* 0.98).

Anche se non è escluso che funzioni con altre versioni meno recenti della serie **2.6**, ***IPFIRE-wall* non compilerà e non funzionerà su alcun *kernel* della serie 2.4.**

1.3.4 Risoluzione dei problemi.

Se la compilazione dei sorgenti dell'interfaccia di *IPFIRE-wall* non va a buon fine, è probabile che l'ambiente di sviluppo non sia configurato in modo corretto: rivedete la sezione 1.3.1 e riportate

17 La configurazione del *kernel* si effettua recandosi nella directory in cui sono stati scompattati i sorgenti, ad esempio */usr/src/linux-2.6.18*, e digitando, da root:

make xconfig, oppure

make menuconfig, a seconda delle preferenze. La prima opzione implica la presenza delle librerie grafiche *QT* assieme al loro ambiente di sviluppo (*libqt3*). La seconda invece necessita del pacchetto *ncurses* e *ncurses-dev* (di sviluppo).

Non ci si soffermerà più a lungo sulla configurazione e installazione di un *kernel* linux, rimandando alle numerose fonti di informazione disponibili in Rete.

18 Non solo, se il vostro sistema supporta *iptables* oppure carica all'avvio delle regole di *iptables*, potrebbe esserci conflitto tra queste e le regole applicate da *IPFIRE-wall*. *Iptables* e *IPFIRE-wall* infatti devono essere usati in modo esclusivo, mai assieme. In caso contrario, la rete potrebbe non funzionare o presentare un comportamento inaspettato. Al fine di vedere se nel vostro sistema sono caricate delle regole di *iptables*, è sufficiente digitare il comando

iptables -L e

iptables -t nat -L.

Se osservate delle regole elencate a seguito di uno o entrambi i comandi, allora ci sono delle regole di *iptables* che già effettuano il firewalling. A questo punto, non avviate o installate *IPFIRE-wall* oppure rimuovete prima le regole di *iptables* con il comando

iptables -F e

iptables -t nat -F,

e poi provate *IPFIRE-wall*.

In realtà, sarebbe bene anche rimuovere i *moduli* del *kernel* di *iptables*, ma per fare questo è necessaria una conoscenza approfondita del sistema. Rivolgetevi ad un esperto oppure documentatevi a fondo in *Internet*.

con un'*email* all'autore eventuali problemi o consultate un amico esperto.

D'altra parte, se la costruzione dei moduli del *kernel* presenta difficoltà, ciò è nella maggior parte dei casi dovuto alla non corrispondenza tra il *kernel* in uso e la configurazione dei sorgenti dello stesso. La rilettura dei due paragrafi precedenti dovrebbe costituire lo spunto di riflessione per risalire alla causa dell'insuccesso. Consultate ancora un amico esperto o al limite inviate un'*email* all'autore di *IPFIRE-wall* per alcuni consigli o per segnalare qualsiasi genere di problemi o *bug*.

1.3.4.1 Esempio di problema risolto.

Un problema comune nell'installazione dei moduli del *kernel* è il seguente, riportato come messaggio sulla *console*:

```
Make *** /lib/modules/2.6.15-26-386/build: no such file or directory: stop  
make: *** [all] Error2
```

In questo caso sarà necessario provvedere all'installazione degli *header* del *kernel*. In un sistema tipo *debian* (l'esempio riguarda una *kubuntu*), è sufficiente individuare il pacchetto degli *header* corrispondente alla versione del *kernel* in uso, in questo caso la *2.6.15-26-386*, e installarlo con il seguente comando (da *root*):

```
apt-get install linux-headers-2.6.15-26-386
```

e il problema è risolto. Riavviare l'installazione di *IPFIRE-wall* per concludere il *setup*.

1.3.5 Rimozione del software.

Per rimuovere il software in modo automatico, basta invocare lo *script* di installazione come indicato nel paragrafo 1.2.2.2 aggiungendo il parametro *uninstall* sulla riga del comando:

```
./install.sh uninstall
```

Verranno così rimossi l'eseguibile, le cartelle *IPFIRE* nella propria *home* e gli *script di avvio*.

2. Esecuzione di IPFIRE-wall.

2.1 Avvio di IPFIRE-wall.

IPFIRE-wall possiede un'interfaccia orientata alla semplicità d'uso che dialoga con il sistema operativo *Linux* e in particolare con la sezione che si occupa di processare i pacchetti che arrivano o sono destinati alla rete.

Tramite l'interfaccia utente è possibile caricare nuove regole di permesso, di negazione, di traduzione degli indirizzi (*NAT*), ovvero elencare le norme stesse, leggere alcune statistiche sul traffico di rete, farsi spedire delle *mail* periodicamente per monitorare un'istanza di *IPFIRE-wall* remota, e così via.

Ogni utente può eseguire un'interfaccia, ma una sola di esse può essere in esecuzione in un determinato istante. Questo è correlato al fatto che il software è stato progettato per l'uso desktop, anche se ancora non è disponibile un'interfaccia grafica che si integri con l'ambiente a finestre.

L'avvio del filtro di rete deve avvenire secondo la seguente modalità:

- *deve essere avviata un'istanza del programma dall'utente root, atta a caricare il modulo del kernel e le regole imposte dall'amministratore;*

- ogni utente è quindi libero di avviare o meno la propria interfaccia con le proprie regole.

Il primo obiettivo si raggiunge in generale in modo automatico tramite lo *script* di avvio che l'*installer* copia durante la fase di *setup*. Tale *script* viene avviato automaticamente quando il computer si accende. Il comando eseguito in automatico, ovvero quello che va impartito nel caso in cui si scelga di avviare il software manualmente, è il seguente, e richiede le credenziali di *root*:

/etc/init.d/rc.ipfire start

in un sistema tipo *debian*, con avvio di tipo *SYSTEM V*, oppure con

/etc/rc.d/rc.ipfire start

in un sistema *Slackware Linux*.

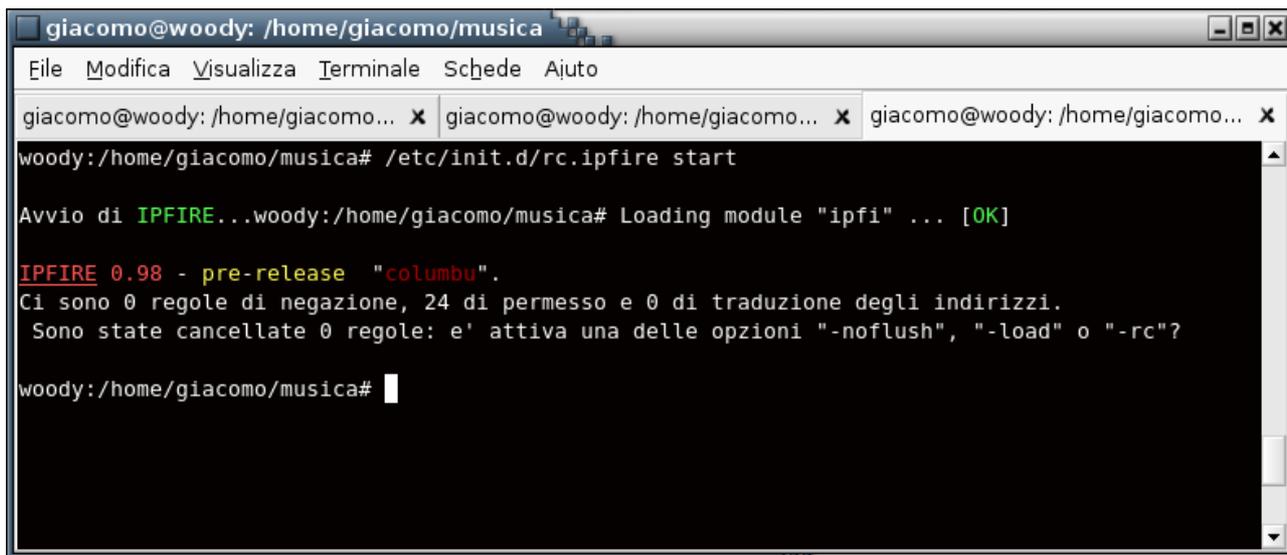
A questo punto si osserverà una schermata con un messaggio simile a quello riportato nella *figura 6*. Si noti che **il programma inserisce automaticamente il modulo nel kernel quando viene avviato da root, a meno che il modulo stesso non sia già stato in precedenza caricato in memoria.**

È necessario che sia *root* ad avviare il firewall la prima volta appunto perché solo egli ha i diritti per caricare un elemento del kernel.

Se l'amministratore non avrà caricato il modulo di IPFIRE-wall, allora nessun utente potrà utilizzare il filtro di pacchetti, né avviare la propria interfaccia.

Ciò avviene per motivi di sicurezza: è l'amministratore a dover decidere le regole fondamentali del firewall e nessun utente potrà modificare o rimuovere né alterare in alcun modo le impostazioni decise dall'amministratore del sistema. Ad esempio, se *root* inserisce una regola di negazione per il sito www.bruttosito.com, anche qualora l'utente *pippo* inserisse una regola di permesso esplicito per il medesimo sito, questa non verrebbe considerata dal filtro, che per sua natura **prima** analizza le regole di **negazione** (e la regola di negazione inserita da *root* non può essere cancellata) e **poi** quelle di **permesso**. D'altra parte, se l'amministratore non impone alcuna regola per il sito sopra citato, oppure ne inserisce una di permesso esplicito, allora l'utente *pippo* può aggiungere una norma di negazione esplicita che **blocca** il traffico verso o da quell'indirizzo. In questo senso **IPFIRE-wall dà libertà ad ogni utente di proteggersi dal traffico indesiderato, senza tuttavia compromettere il livello di sicurezza imposto dall'amministratore.**

2.1 Avvio di IPFIRE-wall.



```
giacomo@woody: /home/giacomo/musica
File Modifica Visualizza Terminale Schede Ajuto
giacomo@woody: /home/giacomo... x giacomo@woody: /home/giacomo... x giacomo@woody: /home/giacomo... x
woody:/home/giacomo/musica# /etc/init.d/rc.ipfire start
Avvio di IPFIRE...woody:/home/giacomo/musica# Loading module "ipfi" ... [OK]
IPFIRE 0.98 - pre-release "columbu".
Ci sono 0 regole di negazione, 24 di permesso e 0 di traduzione degli indirizzi.
Sono state cancellate 0 regole: e' attiva una delle opzioni "-noflush", "-load" o "-rc"?
woody:/home/giacomo/musica#
```

figura 6: il comando `/etc/init.d/rc.ipfire start` produce l'output mostrato sopra. Esso viene eseguito di norma all'avvio del sistema in modo automatico.

Questo costituisce un aspetto importante che differenzia il software qui descritto dagli altri comunemente usati in ambiente *Linux*.

A questo punto, un *utente qualsiasi* che abbia eseguito l'installazione del pacchetto può usare *IPFIRE-wall* digitando

ipfire

dalla linea di comando o cliccando sull'icona del *desktop* che l'*installer* crea durante il *setup*. Con questo comando l'interfaccia viene avviata nella modalità base. Esistono in realtà numerose opzioni che si possono specificare al momento dell'accensione del *firewall*, trattate in uno dei paragrafi a seguire.

2.2 Arresto del filtro di pacchetti.

Arrestare *IPFIRE-wall* significa da un lato terminare l'esecuzione della sua interfaccia utente e dall'altro rimuovere il modulo dal *kernel* quando l'azione di filtro di rete non sia più richiesta. I due paragrafi a seguire illustrano la metodologia da seguire per raggiungere l'uno e l'altro scopo.

2.2.1 Arresto dell'esecuzione dell'interfaccia utente.

L'utente che abbia in esecuzione *IPFIRE-wall* può arrestarlo in modi diversi:

- premendo il tasto "q" oppure "ESC";
- premendo la combinazione di tasti *CONTROL* e *C* contemporaneamente;
- chiudendo la console in cui viene eseguito il programma cliccando sulla *X* della finestra dell'interfaccia.

Ancorche` i primi due metodi siano quelli suggeriti, anche il terzo funziona correttamente poiché il programma intercetta i segnali di stop inviati dalla *shell* che esegue *IPFIRE*.

2.2.2 Rimozione del modulo del *kernel* (richiede i privilegi di *root*).

Il modulo del *kernel* di *IPFIRE-wall* può essere rimosso in qualsiasi momento, anche durante l'esecuzione di un'interfaccia utente, digitando il comando seguente, da *root*:

modprobe -r ipfi

Ovviamente un'eventuale istanza dell'interfaccia smetterà di funzionare dopo questa operazione.

Un altro metodo, che evita la rimozione del modulo mentre un utente sta in contemporanea eseguendo il suo *firewall*, è quello di utilizzare *IPFIRE-wall* stesso per rimuovere dalla memoria il modulo in spazio *kernel*. Ciò si effettua agevolmente invocando lo *script* di inizializzazione con l'opzione *stop*, nel modo che segue:

/etc/init.d/rc.ipfire stop

ovvero, su sistema *Slackware*:

/etc/rc.d/rc.ipfire stop

Se un'altra istanza dell'interfaccia utente è in esecuzione, il comando fallirà. In tale eventualità, l'utente deve arrestare l'esecuzione del filtro, oppure il processo attivo deve essere terminato con il comando

killall ipfire

al fine di sospendere l'esecuzione dell'interfaccia e poter rimuovere il modulo senza problemi.

In alternativa, un altro modo può essere quello di forzare l'uscita di un'eventuale interfaccia al momento dell'invocazione dello *script* sopra riportato:

/etc/init.d/rc.ipfire force_stop

Questo manda un segnale di terminazione ad un'eventuale *IPFIRE-wall* acceso e poi rimuove il modulo dal *kernel*.

A questo punto ***il filtro non è più attivo e la rete non è più protetta!***

Si provveda quindi all'avvio di *iptables* per la sicurezza della propria *workstation Linux*.

Si segnala anche la possibilità per l'utente *root* di riavviare *IPFIRE-wall* “a fondo”, nel senso di rimuovere e poi reinserire il modulo nel *kernel* (ricaricando contestualmente le regole dell'amministratore):

/etc/init.d/rc.ipfire restart

Per chi non avesse lo *script rc.ipfire* installato nel proprio sistema, si ricorda che la sua invocazione con l'opzione *start* corrisponde all'avvio con l'opzione di seguito riportata:

ipfire -rc -user

2.2.2 Rimozione del modulo del kernel (richiede i privilegi di root).

mentre il comando di arresto equivale a questa riga:

ipfire -rc -rmmmod -flush

In particolare, *-rc* specifica che *IPFIRE-wall* va avviato, le regole caricate in memoria e poi l'interfaccia deve terminare, per lasciare spazio all'esecuzione di un'altra istanza utente.

L'opzione *-user* invece abilita l'utente all'inserimento delle proprie regole. Se tale opzione non è attiva, allora l'amministratore impedisce agli utenti non privilegiati di aggiungere le regole di protezione personali. Si valuti pertanto con accuratezza questa possibilità, che costituisce la caratteristica peculiare del software discusso.

-rmmmod istruisce *IPFIRE-wall* che all'uscita deve anche rimuovere il modulo dalla memoria del kernel.

-flush invece, esegue lo svuotamento delle regole del filtro all'uscita. E' evidente inoltre che *-rmmmod*, rimuovendo il modulo intero dalla memoria, porta via con esso tutte le regole presenti.

Tutte queste opzioni sono appannaggio del solo utente *root*.

2.2.3 Osservazione importante.

Quando l'utente *amministratore* avviasse *IPFIRE-wall* senza l'opzione *-rmmmod*, e senza alcuna delle opzioni *-rc* o *-load*¹⁹ tenga presente che allorchè ***l'interfaccia avrà terminato la sua esecuzione, il modulo del kernel rimarrà caricato in memoria, ma senza più le regole!***

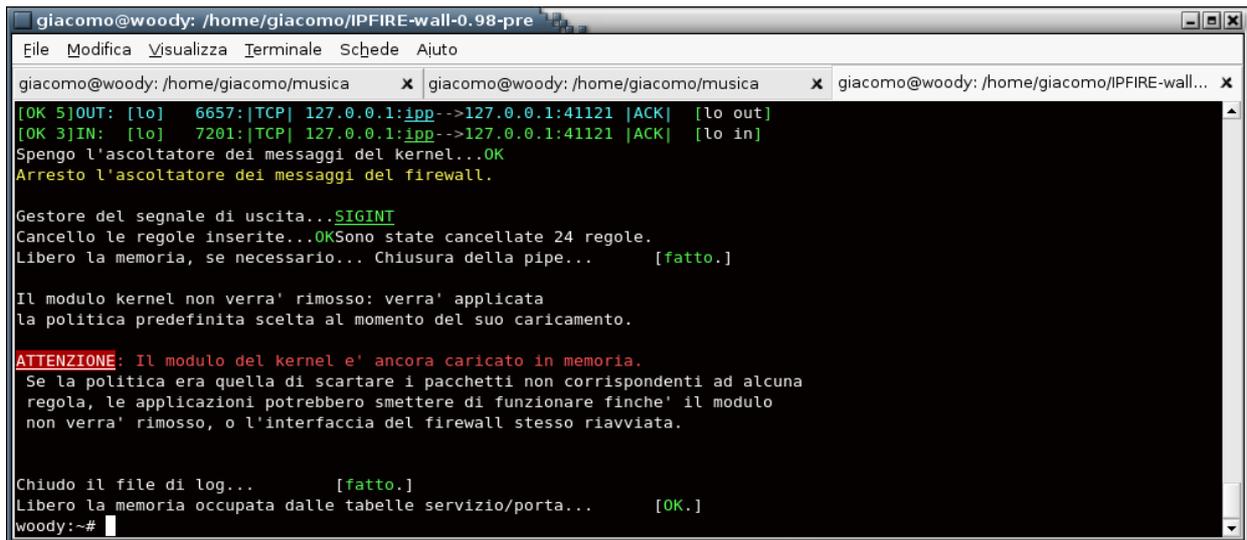
Ciò implica che a tutti i pacchetti verrà applicata la politica di valutazione predefinita, che di norma è quella di ***negare il passaggio di ogni pacchetto attraverso il kernel***²⁰. Questo significa che ***la rete smetterà di funzionare*** e con essa anche le applicazioni che la stavano utilizzando.

Pertanto si eviti di avviare *IPFIRE-wall* da *root* senza le opzioni opportune sopra descritte o senza invocare gli *script ad hoc*.

La circostanza appena descritta viene comunque segnalata da un messaggio in colore rosso al momento della terminazione dell'interfaccia utente (figura 7).

19 L'opzione *-load* è analoga all'opzione *-rc*. La prima tuttavia, produce meno messaggi sul terminale all'avvio.

20 La politica predefinita può essere cambiata agendo sul *proc file /proc/IPFIRE/policy*. Il metodo tuttavia non è descritto in questa sede.



```

giacomo@woody: /home/giacomo/IPFIRE-wall-0.98-pre
File Modifica Visualizza Terminale Schede Ajuto

giacomo@woody: /home/giacomo/musica x giacomo@woody: /home/giacomo/musica x giacomo@woody: /home/giacomo/IPFIRE-wall... x

[OK 5]OUT: [lo] 6657:|TCP| 127.0.0.1:ipp-->127.0.0.1:41121 |ACK| [lo out]
[OK 3]IN: [lo] 7201:|TCP| 127.0.0.1:ipp-->127.0.0.1:41121 |ACK| [lo in]
Spendo l'ascoltatore dei messaggi del kernel...OK
Arresto l'ascoltatore dei messaggi del firewall.

Gestore del segnale di uscita...SIGINT
Cancello le regole inserite...OKSono state cancellate 24 regole.
Libero la memoria, se necessario... Chiusura della pipe... [fatto.]

Il modulo kernel non verra' rimosso: verra' applicata
la politica predefinita scelta al momento del suo caricamento.

ATTENZIONE: Il modulo del kernel e' ancora caricato in memoria.
Se la politica era quella di scartare i pacchetti non corrispondenti ad alcuna
regola, le applicazioni potrebbero smettere di funzionare finche' il modulo
non verra' rimosso, o l'interfaccia del firewall stesso riavviata.

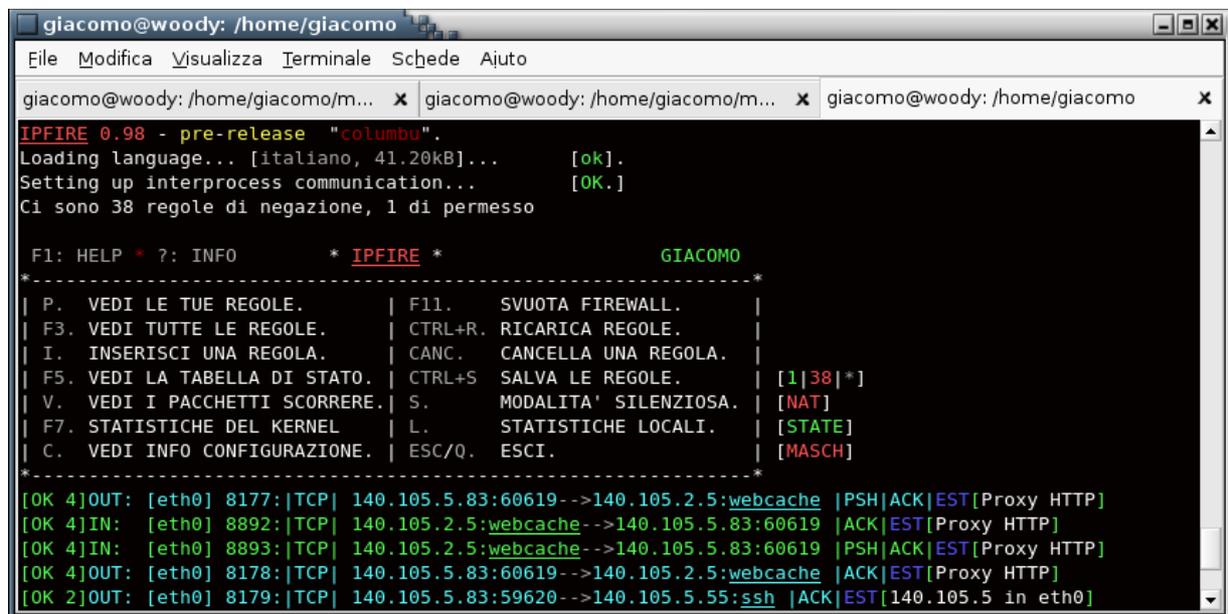
Chiudo il file di log... [fatto.]
Libero la memoria occupata dalle tabelle servizio/porta... [OK.]
woody:~#

```

figura 7: il messaggio di warning che viene emesso dall'interfaccia qualora root terminasse IPFIRE-wall senza rimuovere il modulo dal kernel, ovvero senza che la stessa interfaccia sia stata avviata con le opzioni adatte allo scaricamento del modulo nella fase di uscita.

2.2.4 Interagire con IPFIRE-wall tramite il menù principale.

Una volta avviata, l'interfaccia presenta un menù attraverso il quale è semplice configurare e interagire con il filtro di pacchetti. Il menù presenta diverse voci, come in figura 8, e ogni opzione viene attivata premendo il tasto ad essa corrispondente, senza bisogno di dare conferma con l'invio.



```

giacomo@woody: /home/giacomo
File Modifica Visualizza Terminale Schede Ajuto

giacomo@woody: /home/giacomo/m... x giacomo@woody: /home/giacomo/m... x giacomo@woody: /home/giacomo x

IPFIRE 0.98 - pre-release "columbu".
Loading language... [italiano, 41.20kB]... [ok].
Setting up interprocess communication... [OK.]
Ci sono 38 regole di negazione, 1 di permesso

F1: HELP * ?: INFO * IPFIRE * GIACOMO
*-----*
| P. VEDI LE TUE REGOLE. | F11. SVUOTA FIREWALL. |
| F3. VEDI TUTTE LE REGOLE. | CTRL+R. RICARICA REGOLE. |
| I. INSERISCI UNA REGOLA. | CANC. CANCELLA UNA REGOLA. |
| F5. VEDI LA TABELLA DI STATO. | CTRL+S SALVA LE REGOLE. | [1|38|*]
| V. VEDI I PACCHETTI SCORRERE. | S. MODALITA' SILENZIOSA. | [NAT]
| F7. STATISTICHE DEL KERNEL | L. STATISTICHE LOCALI. | [STATE]
| C. VEDI INFO CONFIGURAZIONE. | ESC/Q. ESCI. | [MASCH]
*-----*

[OK 4]OUT: [eth0] 8177:|TCP| 140.105.5.83:60619-->140.105.2.5:webcache |PSH|ACK|EST[Proxy HTTP]
[OK 4]IN: [eth0] 8892:|TCP| 140.105.2.5:webcache-->140.105.5.83:60619 |ACK|EST[Proxy HTTP]
[OK 4]IN: [eth0] 8893:|TCP| 140.105.2.5:webcache-->140.105.5.83:60619 |PSH|ACK|EST[Proxy HTTP]
[OK 4]OUT: [eth0] 8178:|TCP| 140.105.5.83:60619-->140.105.2.5:webcache |ACK|EST[Proxy HTTP]
[OK 2]OUT: [eth0] 8179:|TCP| 140.105.5.83:59620-->140.105.5.55:ssh |ACK|EST[140.105.5 in eth0]

```

figura 8. Il menù principale di IPFIRE-wall e le 14 opzioni disponibili.

Una descrizione più dettagliata delle singole voci è data nella documentazione web. Si sottolinea in

2.2.4 Interagire con IPFIRE-wall tramite il menù principale.

questa sede la differenza tra i tasti *P* e *F3*: il primo produce la stampa delle sole regole dell'utente (indicato con il suo nome in verde stampatello in alto a destra, sopra la cornice a trattini del menù), mentre il secondo elenca *tutte* le regole caricate nel filtro di pacchetti, cioè quelle dell'utente e quelle dell'amministratore.

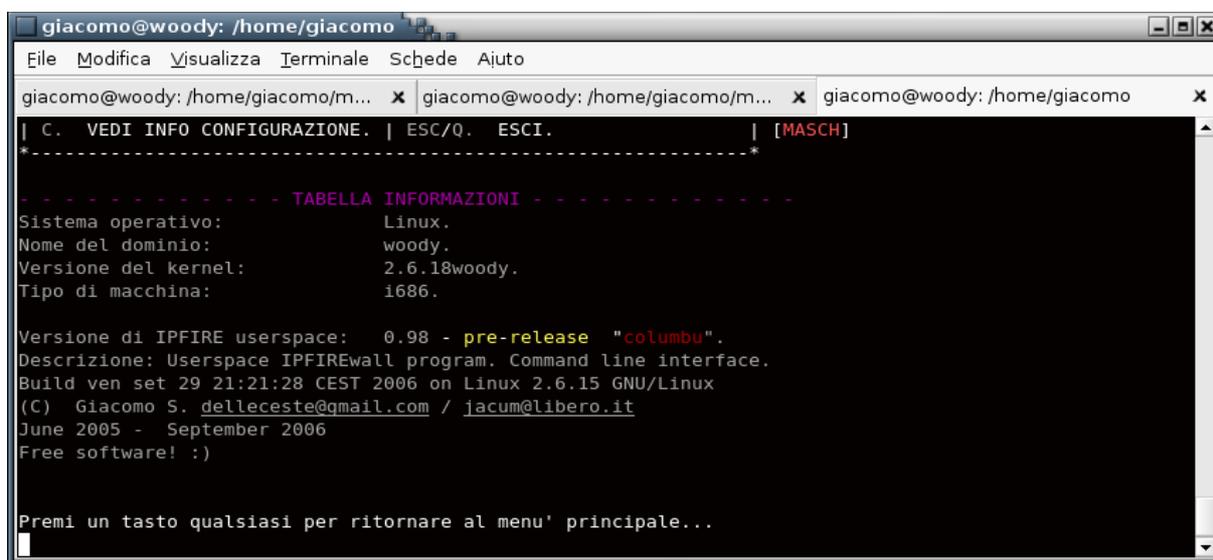
Si notino anche le indicazioni tra parentesi quadre nella seconda metà in basso a destra rispetto alla cornice del menù: la prima riga indica il numero di regole inserite dall'utente (senza quelle di *root*). Nell'ordine, quelle di permesso (in verde), quelle di negazione (in rosso) e quelle di traduzione degli indirizzi (in giallo). Si rammenta che l'utente normale non possiede i privilegi di introdurre regole di traduzione degli indirizzi (*NAT*).

Nel caso rappresentato, le regole di *stato*²¹ sono abilitate (in verde la parola chiave *stato*, che altrimenti sarebbe stata di colore rosso), mentre sono disabilitati il *NAT* e il *mascheramento degli indirizzi* (in rosso).

Il tasto *F5* stampa le tabelle riguardanti le connessioni di stato presenti nel *kernel* in un dato istante. L'elenco rappresenta sostanzialmente le connessioni *in corso* o comunque *recentemente filtrate*.

Il tasto *F7* presenta le statistiche precise fornite dal *kernel* sulle operazioni effettuate sul traffico di rete, mentre il tasto *L* rappresenta le statistiche dal punto di vista dell'*interfaccia* utente, le quali sono solo una stima del traffico sulla base della comunicazione tra lo spazio dell'*interfaccia utente* e quello *kernel*, e in generale dipendono dall'istante a partire dal quale l'interfaccia è stata avviata e dal livello di *log* impostato dall'utente.

La pressione del tasto *?* (punto interrogativo) ha come conseguenza la visualizzazione di alcune informazioni sul software, come mostrato in *figura 9*.



```
giacomo@woody: /home/giacomo
File Modifica Visualizza Terminale Schede Ajuto
giacomo@woody: /home/giacomo/m... x giacomo@woody: /home/giacomo/m... x giacomo@woody: /home/giacomo x
| C. VEDI INFO CONFIGURAZIONE. | ESC/Q. ESCI. | [MASCH]
*-----*
- - - - - TABELLA INFORMAZIONI - - - - -
Sistema operativo: Linux.
Nome del dominio: woody.
Versione del kernel: 2.6.18woody.
Tipo di macchina: 1686.

Versione di IPFIRE userspace: 0.98 - pre-release "columbu".
Descrizione: Userspace IPFIREwall program. Command line interface.
Build ven set 29 21:21:28 CEST 2006 on Linux 2.6.15 GNU/Linux
(C) Giacomo S. delleceste@gmail.com / jacum@libero.it
June 2005 - September 2006
Free software! :)

Premi un tasto qualsiasi per ritornare al menu' principale...
```

21 Si veda la documentazione *online* per una descrizione circa le regole *di stato*: in breve, inserire una regola di stato che permetta il traffico di un pacchetto in *una direzione*, consente in modo automatico di garantire il traffico in risposta *a quel pacchetto anche nell'altra direzione*. Ad esempio, permettere il traffico *verso* la porta 80 con una regola *di stato*, significa automaticamente permettere l'accesso *anche* ai pacchetti che provengono *dalla* porta 80. Con una sola regola, si ottiene ciò che si otterrebbe con due regole, una di permesso *verso* la porta 80, un'altra di permesso *dalla* porta 80.

figura 9. Informazioni sul sistema operativo, sulla versione del kernel e sulla versione di IPFIRE-wall.

Il tasto *s* abilita la modalità silenziosa: il kernel non manda più informazioni sui pacchetti filtrati e sul verdetto che li riguarda²². Al contrario, il tasto *v* ripristina la modalità verbosa²³.

La procedura di inserimento di una regola è guidata e chiede all'utente ogni parametro da configurare. Completata la costruzione di una regola, è possibile rinunciare alla sua introduzione nell'insieme, oppure aggiungerla ad esso, in coda o anche in una precisa posizione. Una particolare posizione per una determinata regola può influenzare il comportamento del filtro, poiché esso si ferma nella scansione dell'elenco qualora trovasse una corrispondenza di un pacchetto con una norma: questo lascia intravedere all'utente accorto la possibilità di *ottimizzare* l'applicazione delle regole del *firewall*.

Il metodo di rimozione invece richiede la posizione esatta della regola da cancellare: è pertanto bene, *prima* di procedere alla cancellazione di una norma, effettuare una *stampa* di tutte le regole, al fine di memorizzare la posizione di quella da togliere. In ogni caso, prima di confermare l'operazione, viene stampata sulla console la regola interessata.

Alcune opzioni non sono direttamente indicate sul menù principale, benché ci sia traccia di esse sulla documentazione *web*.

Ad esempio, la pressione del tasto *b* disabilita la risoluzione dei numeri di porta nei rispettivi *servizi* che essi rappresentano. Le coppie *servizio-numero di porta* vengono caricate *in memoria* dal file */etc/services* e la consultazione dell'insieme delle corrispondenze viene reso in questo modo efficiente, evitando la lettura da file ad ogni interrogazione. D'altro canto, il tasto *u* riabilita la risoluzione del numero di porta nel servizio corrispondente.

Il lettore interessato alla conoscenza approfondita di tutti gli aspetti di *IPFIRE-wall* consulti la documentazione *web*, all'indirizzo <http://www.giacomos.it/ipfire/index.html>.

Ivi potranno trovarsi esplicitate tutte le opzioni di avvio, quelle rappresentate nel file di configurazione utilizzato dal software, le indicazioni su come eseguire il *mailer*, nonché numerosi *scenari* di topologie di rete e di utilizzo di *IPFIRE-wall* ad esse correlato.

22 Questo avviene a partire dalla versione 0.98. In quelle precedenti, il *kernel* spedisce i pacchetti all'interfaccia utente anche nella modalità silenziosa, ma è in questo caso l'interfaccia stessa a non stamparli sulla console. Con la versione 0.98, si è pensato che sarebbe stato più efficiente evitare la comunicazione *kernel/utente* per spedire dei pacchetti che comunque non venivano visti.

La scelta originaria era giustificata dall'intenzione di inviare i pacchetti all'interfaccia per mantenere il *log* sul file di testo. Ma quella scelta non aumentava comunque il livello di sicurezza, essendo comunque il file di *log* di proprietà dell'utente che eseguiva l'interfaccia e quindi alla fine in balia di esso e passibile di modifiche manuali o addirittura di rimozione.

23 In realtà la modalità verbosa può essere configurata per assumere diversi livelli di intensità. Si assume che il livello predefinito sia quello ottimale, rimandando alla lettura del manuale *online* il lettore interessato ad alterare il grado di loquacità dell'interfaccia.

2.2.5 Conclusioni.

Le pagine di questa guida, lungi dalla pretesa di dare una visione completa ed esaustiva dell'utilizzo del *filtro di pacchetti di rete IPFIRE-wall*, hanno descritto la procedura di *installazione* del software su un sistema *GNU/Linux* e le modalità per avviare una *prima esecuzione* dello stesso, con le opzioni e le configurazioni di base, che garantiscono un livello di protezione fondamentale.

Si rimanda ad una letteratura più specifica per apprendere il funzionamento e l'architettura delle reti di computer, nonché della sua implementazione in un sistema *Linux*.

Si consiglia infine al lettore interessato la consultazione della documentazione *online* all'indirizzo <http://www.giacomos.it/ipfire/index.html> affinché approfondisca la conoscenza di *IPFIRE-wall*. Presso tale sito *web* vengono ringraziate le persone che hanno aiutato l'autore nella realizzazione del progetto che, principalmente, è stato realizzato a scopo didattico, nella speranza che esso fornisca alcuni esempi utili di programmazione in spazio *kernel* e, perché no, anche in quello utente.

Segue un'appendice che introduce alcuni concetti del protocollo internet *TCP/IP* in modo semplificato, atta a chiarire alcuni concetti che aiutano l'utente a comprendere il funzionamento dei filtri di pacchetto, e in particolare di *IPFIRE-wall*.

Buon divertimento con il software libero e con il mondo *Linux*!

Appendice.

A.1 Introduzione al protocollo internet TCP/IP.

A.1.1 La rete e l'instradamento dei pacchetti in Linux (cenni).

I pacchetti di rete percorrono il cuore del sistema operativo *linux (kernel)* seguendo due possibili strade e attraversando alcuni punti specifici. Un pacchetto *in ingresso* nel sistema deve essere ispezionato dal *kernel* per conoscerne la destinazione: questa potrebbe essere *locale (INPUT)*, oppure il pacchetto potrebbe essere *inoltrato* verso un altro nodo della rete (*FORWARD*). Per quanto riguarda invece i pacchetti *generati localmente*, ad esempio dalla nostra applicazione browser web preferita (ad esempio *mozilla* oppure *firefox*), questi saranno destinati ad *uscire (OUTPUT)* dal nostro sistema verso un computer remoto.

I pacchetti che raggiungono il nostro sistema possono attraversare il percorso di INPUT oppure di FORWARD, a seconda che sono destinati a noi oppure ad un altro computer. Quelli generati dal nostro sistema, attraverseranno la catena di OUTPUT. In ogni caso, i pacchetti che attraversano la catena di FORWARD (inoltre verso altre macchine) mai attraverseranno quella di OUTPUT o di INPUT. Quindi all'arrivo di un pacchetto al nostro *firewall linux*, esso si trova ad un bivio: direzione *INPUT* se il pacchetto e' per noi (ad esempio una risposta all'applicativo *firefox* contenente una pagina web), direzione *FORWARD* se esso non contiene il nostro come indirizzo di destinazione (ad esempio un *firewall* di instradamento di pacchetti (*router*) inoltra quelli dei computer da esso protetti verso la rete Internet restituendo in seguito le risposte verso le corrispondenti macchine della rete interna). Solo nelle cosiddette "catene" elencate fino ad ora è possibile effettuare il *filtraggio* dei pacchetti di rete: quella di *INPUT*, *OUTPUT* o *FORWARD*. Questo significa che una regola di blocco sulla catena di *INPUT* blocca un pacchetto *in arrivo e destinato a noi*, una regola sul *FORWARD* invece viene applicata a un pacchetto *in arrivo ma non destinato a noi*, mentre una regola in *OUTPUT* ne blocca uno *generato da una applicazione sul nostro sistema*, e quindi diretta verso un computer esterno.

Fino a questo punto le cose risultano piuttosto semplici. A queste considerazioni tuttavia vanno aggiunte alcune precisazioni. Un *firewall* o *filtro di pacchetti* spesso non si limita a lasciare passare o meno il traffico attraverso di esso. A volte è necessario che alcuni pacchetti in arrivo e destinati a una macchina *X* siano reindirizzati verso un altro computer *Y*. È il caso rappresentato ad esempio da un server web, che ospita le pagine di un'università: esso di norma non viene installato sul computer di confine della rete, poiché ivi sarebbe esposto direttamente agli attacchi provenienti dalla grande *Internet*. Il server web trova piuttosto spazio su una macchina interna alla rete, protetta dal *firewall* o anche da un *indirizzo privato* (un indirizzo privato appartiene per definizione ad una classe di indirizzi che non può essere raggiunta dall'esterno di una rete privata). Allora un nodo della rete che volesse visitare il sito web dell'università non potrebbe indirizzare il server web direttamente, essendo l'indirizzo di quest'ultimo privato, bensì deve indirizzare quella macchina

dell'università che risiede al confine tra la rete interna e quella esterna ed effettuare ad essa la richiesta della pagina web desiderata. A questo punto, il computer *firewall* di confine non possiede un server web in esecuzione, tuttavia è a conoscenza che quest'ultimo risiede presso la macchina Z della rete interna. Il *firewall* inoltra quindi la richiesta verso la macchina Z (facendo transitare i pacchetti ad essa reindirizzati attraverso la catena di *FORWARD*), e le risposte da Z saranno naturalmente fatte pervenire al computer remoto in attesa del documento web. Il procedimento viene attuato tracciando il traffico con l'ausilio di tabelle allocate nella memoria del *kernel Linux*. Appare evidente quindi che il destino di un pacchetto in arrivo (*INPUT* oppure *FORWARD*) può essere cambiato dopo il suo ingresso nel nostro sistema, ma prima del bivio *INPUT/FORWARD*. Il punto in cui avviene questo cambiamento è il cosiddetto punto di *PRE-ROUTING*, cioè, letteralmente, “prima dell'indirizzamento”.

Affinché un pacchetto veda il proprio percorso cambiare nel punto di *PRE-ROUTING*, è necessario che ivi il filtro di pacchetti modifichi il campo dell'indirizzo del pacchetto stesso: è in *PRE-ROUTING* che un pacchetto destinato alla macchina X viene reindirizzato verso una macchina Y, oppure anche a noi stessi, o ancora che un pacchetto destinato a noi (alla catena di *INPUT*) sia inoltrato ad una macchina Z (quindi attraverso la catena di *FORWARD*). In modo del tutto analogo, un pacchetto che sta per lasciare la macchina A, può essere modificato poco prima di uscire, ad esempio per farlo apparire come proveniente da una macchina diversa da quella dalla quale è effettivamente stato generato. È il caso del lavoro svolto dal *firewall* quando deve permettere ad una macchina interna alla rete privata (con indirizzo privato e quindi invisibile dall'esterno) di accedere alla rete pubblica *Internet*. Questo tipo di intervento viene svolto nel punto di *POST-ROUTING*.

La *figura 10* rappresenta il modello descritto finora.

Si osservi lo schema a blocchi piuttosto che i fumetti, che sono tratti da una figura atta a descrivere il flusso del codice sorgente di *IPFIRE-wall*, che avviene in alcune pagine della documentazione *online*. Le operazioni di cui si è accennato nei punti di *PRE* e *POST routing* prendono il nome di operazioni di *NAT*, ovvero *Network Address Translation*, traduzione degli indirizzi di rete. In effetti, vengono proprio manipolati gli indirizzi di rete contenuti nei pacchetti per consentirne il corretto instradamento. In particolare, la traduzione degli indirizzi può riguardare il cambiamento dell'indirizzo sorgente del pacchetto (l'indirizzo della macchina che lo spedisce), oppure di quello destinazione (l'indirizzo verso cui è diretto). Si parla quindi di *NAT sorgente (SNAT)* e destinazione (*DNAT*). Il *DNAT* si esegue in *PRE-ROUTING*, per cambiare la destinazione del pacchetto, mentre il *Source NAT* si esegue in *POST-ROUTING*, per cambiare l'indirizzo sorgente del pacchetto. In realtà con *IPFIRE-wall* si può eseguire *DNAT* anche in *OUTPUT*, sotto particolari condizioni²⁴. Il filtraggio dei pacchetti avviene invece nelle catene di *INPUT*, *OUTPUT* o *FORWARD*, e in tale contesto essi non vengono manipolati, ma semplicemente accettati o scartati, a seconda della regola che li riguarda.

²⁴ Si veda la documentazione *online*.

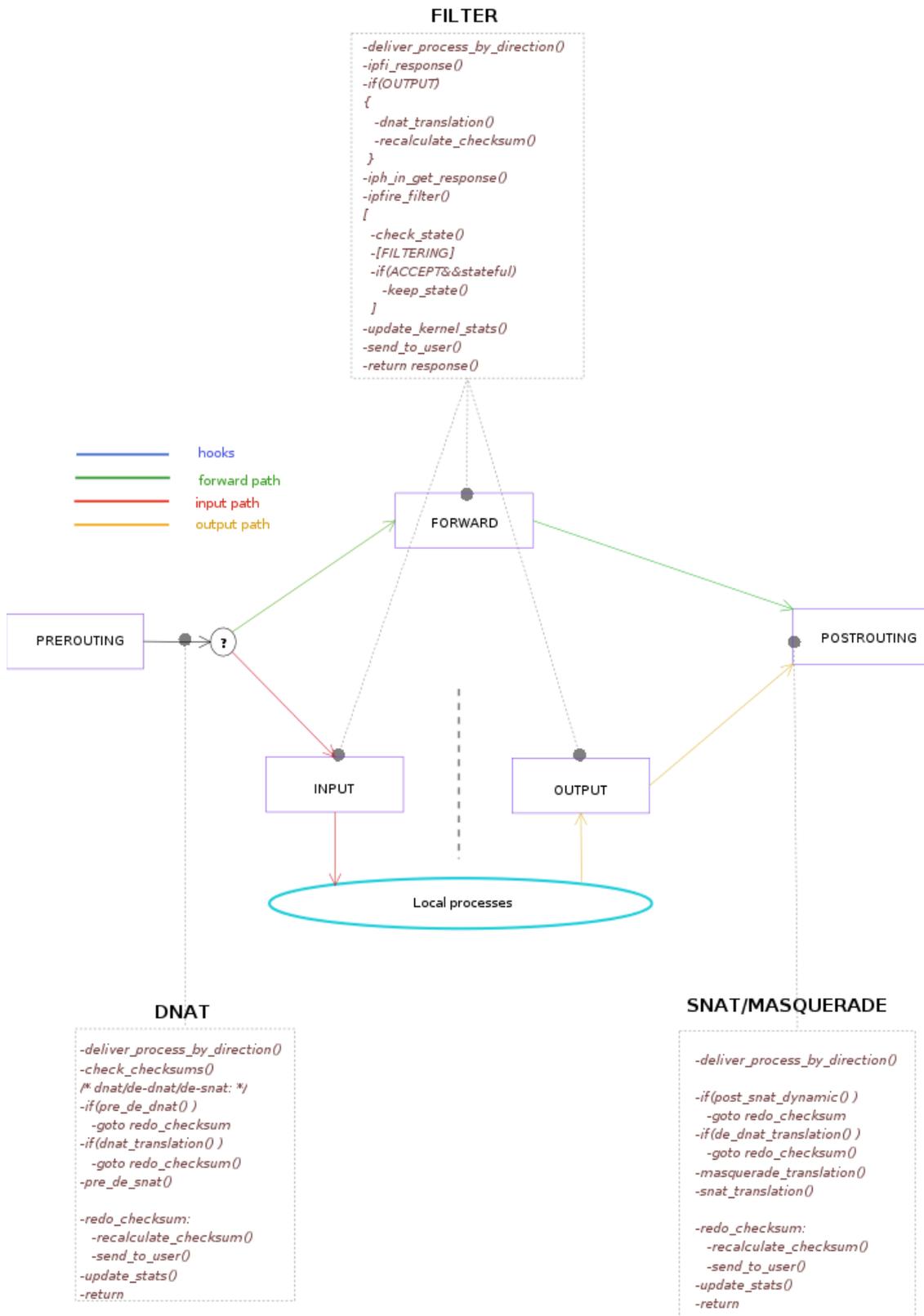


Figura 10: si osservi lo schema a blocchi che esemplifica i punti in cui opera il filtro e quelli in cui avviene la traduzione degli indirizzi di rete.

A.1.2 Regole di IPFIRE-wall e protocollo di rete TCP/IP.

IPFIREwall può essere istruito per filtrare pacchetti di rete sulla base delle loro proprietà caratteristiche del protocollo di rete internet *TCP/IP* (*Transmission Control Protocol/Internet Protocol*).

A.1.2.1 Identificazione di una macchina in rete e di un servizio al suo interno.

Per identificare un computer in rete, è necessario che a questo sia associato un indirizzo univoco. Un indirizzo di rete è costituito da un numero di 32 bit, che permette di assegnare quindi un totale di 2 elevato alla 32 valori diversi. Una volta identificata una particolare macchina nella rete, questa può avere in esecuzione più servizi: è quindi necessario indicare a quale dei servizi disponibili si è interessati. Ad ogni servizio è associato un numero, detto numero di porta, o semplicemente *porta*. Due servizi diversi saranno contraddistinti da due numeri di porta diversi.

Quindi la coppia indirizzo-porta è sufficiente a identificare un computer in rete e, all'interno di esso, un particolare servizio in esecuzione.

Ad esempio, il servizio di recupero di risorse web è in ascolto sulla porta 80, mentre il servizio di risoluzione dei nomi Internet sulla porta 53, o ancora il recupero della posta elettronica avviene contattando un server in ascolto sulla 110 mentre il deposito della posta in uscita viene effettuato su un server associato alla 25. Questi costituiscono solo alcuni esempi di coppie servizio/porta, ma leggendo il file */etc/services* (si provi il comando

cat /etc/services

da una shell linux), se ne possono osservare molti altri.

A.1.2.2 Affidabilità della connessione di rete e servizi orientati alla connessione.

Il protocollo in uso per la trasmissione dei dati in rete è chiamato *TCP/IP* (Protocollo di Controllo della Trasmissione / Protocollo Internet).

La parte *IP* rappresenta quella sezione di protocolli atta a gestire principalmente l'instradamento dei pacchetti di rete da sorgente a destinazione, mentre la parte di controllo (*TCP*) si preoccupa dell'affidabilità e della gestione della ritrasmissione e del riordinamento dei pacchetti giunti corrotti o in ordine errato. Il controllo della trasmissione fornisce vari livelli di affidabilità ed efficienza. Ancora, si può dire che, in *TCP/IP*, *TCP* rappresenta il cosiddetto *livello di trasporto*, mentre *IP* rappresenta il *livello di rete*.

Il livello di trasporto fornisce, come già accennato, diversi livelli di affidabilità ed efficienza. Due concetti sono fondamentali per caratterizzare un protocollo di trasporto: esso può essere *orientato* o *non orientato* alla connessione, *affidabile* o *non affidabile*. Un altro protocollo spesso utilizzato nella comunicazione di rete è quello *ICMP*, *Internet Control Message Protocol*, che appunto viene utilizzato per scambiare messaggi di controllo.

Il programma *ping* rappresenta un esempio d'uso del protocollo *ICMP*. Anche i *firewall* spesso rispondono tramite messaggi *ICMP* qualora vogliano restituire qualche codice d'errore o la non disponibilità di qualche servizio. *IPFIRE-wall* supporta i protocolli *TCP*, *UDP* e *ICMP*.

Una regola deve *sempre indicare esplicitamente la direzione* a cui va applicata. Secondo la trattazione sopra, le direzioni saranno quelle di ingresso (*INPUT*), uscita (*OUTPUT*), inoltra (*FORWARD*), *pre-routing* (*PRE*) o *post-routing* (*POST*).

I concetti rappresentati dalle direzioni sono stati illustrati poco sopra: i più importanti sono costituiti dalle direzioni di ingresso, uscita e inoltra. Il lettore a cui non risultassero ancora chiari i significati di *pre* e *post-routing* può tranquillamente proseguire la trattazione, che non farà ulteriore riferimento ad essi; chi invece non avesse compreso il significato di direzione di ingresso e uscita, rilegga attentamente i paragrafi precedenti.

Al momento dell'aggiunta di una nuova regola è sempre bene specificare il *tipo di protocollo* a cui pertiene.

Inoltre, possono essere specificati gli *indirizzi sorgente e destinazione*. Gli indirizzi di rete vengono espressi nella forma decimale a punti: quattro numeri da 0 a 255 sono separati da un punto (notazione IP versione 4). Ad esempio, sono validi indirizzi di rete IP

192.168.0.1

o

158.110.28.25.

IPFIRE-wall consente inoltre di specificare un *intervallo di indirizzi*, indicando i due estremi, come ad esempio

192.168.1.0-192.168.1.254,

oppure, utilizzando la notazione equivalente,

192.168.1.0/23 o

192.168.1.0/255.255.254.0,

dove si può osservare la coppia indirizzo/maschera di *sottorete* (si veda la bibliografia per gli approfondimenti).

Gli indirizzi di rete possono essere introdotti in *IPFIRE-wall* anche in *forma negata*, preponendo il carattere di negazione *!* all'indirizzo che deve essere negato. La direttiva "**!192.168.0.2**" significa ad esempio che l'indirizzo di rete deve essere diverso da **192.168.0.2**. Ancora, il carattere *!* può venire anteposto anche ad un intervallo, intendendo così che gli indirizzi al suo interno sono esclusi dalla corrispondenza.

In modo del tutto analogo *IPFIRE-wall* consente di esprimere direttive riguardanti le *porte* di una connessione. Il numero di porta è semplicemente un intero minore di 65536 (la porta è un intero senza segno su 16 bit, e quindi un numero di 16 bit può assumere un valore da zero a 65535, pari a 2 elevato alla 16 meno 1). Come per gli indirizzi, è possibile esprimere una *singola porta* o un *intervallo* di porte, *anche in forma negata*.

IPFIRE-wall consente di determinare anche a quale *interfaccia di rete* va applicata una regola: un

A.1.2.2 Affidabilità della connessione di rete e servizi orientati alla connessione.

computer connesso alla ad una *internet* può possedere più di un dispositivo di rete e pertanto una regola si può applicare a tutte ovvero a una sola delle interfacce di collegamento presenti. Un caso comune è costituito da un PC con un *modem* e una *scheda di rete (ethernet)*: in tal caso i nomi delle interfacce potrebbero essere *ppp0*, se il modem effettua una connessione punto a punto *Point to Point* con il provider Internet remoto, e *eth0* per la scheda di rete *ethernet*.

La procedura di inserimento di una regola è guidata passo passo: per ogni campo viene richiesto il valore desiderato dall'utente, che alla fine confermerà o annullerà l'aggiunta della nuova prescrizione per il filtro dei pacchetti.

L'ultima fase della procedura chiede di dare un nome semplice (al più di venti caratteri) alla regola, per permetterne il riconoscimento immediato al momento della sua applicazione e visualizzazione sulla console dei messaggi di IPFIRE-wall.

Quando una nuova regola viene aggiunta, l'utente ha la facoltà di deciderne la posizione: o in coda alle altre (e quindi il *firewall* la consulerà per ultima), opzione predefinita, oppure in una particolare posizione a scelta.

Si osservi che questo passo richiede particolare attenzione: è sempre buona norma conoscere approfonditamente le regole già presenti insieme con il loro ordine e ricordare che *quando una regola viene applicata con successo ad un pacchetto di rete, la scansione della lista ha termine*. Questo significa ad esempio che, se la terza regola consente *tutto* il traffico *in uscita del protocollo TCP*, è inutile inserire una regola più specifica al quinto posto che permetta i pacchetti *in uscita, di protocollo TCP e verso la porta 80*.

Se tuttavia le due regole sopra appaiono in ordine inverso, il traffico TCP in uscita verso la porta 80 verrà permesso esplicitamente dalla regola per la porta 80, mentre tutto il traffico in uscita, TCP, verso una porta diversa dalla 80 sarà permesso dalla regola generica a seguire. Si riporta infine un esempio di una regola così come viene salvata da *IPFIRE-wall*. Anche se l'utente non dovrà mai preoccuparsi di scriverne una manualmente, essendo disponibile un'interfaccia interattiva, si può notare la chiarezza e la semplicità della rappresentazione delle direttive che costituiscono le politiche di gestione del filtro di pacchetti.

```
RULE
NAME=Telnet->192.168.0.2
POSITION=10
DIRECTION=OUTPUT
OUTDEVICE=eth0
MYSRCADDR
DSTADDR=192.168.0.2
PROTOCOL=6
DSTPORT=23
KEEP_STATE=YES
```

A.1.2.3 Regole di stato.

Dalla trattazione svolta finora dovrebbe risultare chiaro che una comunicazione via rete avviene tra due computer che devono dialogare tra loro conoscendo i rispettivi indirizzi di rete. Non solo,

all'arrivo di un pacchetto alla destinazione giusta, questo deve anche essere consegnato all'applicazione corretta, in ascolto su una determinata porta. Se ad esempio la macchina A con indirizzo 212.255.12.120 manda un pacchetto al server web per scaricare una pagina al computer B 158.110.28.25, dovrà indirizzare la richiesta all'applicativo server web, in ascolto sulla porta 80 TCP.

A questo punto, la macchina B risponderà alla macchina A conoscendo l'indirizzo di rete di quest'ultima e alla porta aperta a sua volta dalla stessa macchina A per identificare la connessione con il server web. Infatti anche il richiedente il servizio deve aprire una porta affinché il server possa identificare il punto d'accesso del destinatario a livello trasporto.

La situazione è quindi la seguente:

A: <212.255.12.120, porta 4096> -> B: <158.110.28.25, porta 80>

essendo A il mittente che inizia la connessione e B il destinatario, ed essendo, nel pacchetto nella direzione A -> B, l'indirizzo di A (212.25...) quello sorgente e l'indirizzo di B (158.110...) quello destinazione (insieme alle rispettive porte). Quando B risponde ad A, è evidente che il pacchetto da B ad A avrà l'indirizzo di B, porta di B (80) come sorgente e l'indirizzo di A, porta 4096 come destinazione.

Supponiamo dunque di avere installato *IPFIRE-wall* sul computer A: se noi inserissimo una regola di permesso per i pacchetti *in uscita* verso l'*indirizzo destinazione* 158.110.28.25, porta 80, certamente i pacchetti dal nostro PC A uscirebbero verso B. Questo è sufficiente affinché possiamo visualizzare la pagina web, ovvero affinché la connessione sia stabilita?

Il lettore attento avrà già intuito il problema: i pacchetti di risposta da B verso di noi (A) presentano l'indirizzo di B come sorgente, e quello di A come destinazione, esattamente il caso opposto del nostro primo pacchetto in uscita da A (sorgente), verso B (destinazione).

Di norma quindi, è necessaria una regola per lasciare uscire un pacchetto verso la destinazione, e una regola con i campi inversi per consentire le risposte.

In IPFIRE-wall, grazie alle regole di stato, è possibile definire una regola per un pacchetto o un insieme di pacchetti verso una direzione, e consentire il loro riconoscimento quando viaggiano nella direzione opposta : è sufficiente che la regola sia dichiarata di stato.

In questo modo, oltre ad evitare di scrivere per ogni regola di permesso la sua complementare, si risolve un ulteriore problema: in generale, se con una regola si permette la comunicazione tra A e B e quella tra B ed A con la complementare, si permette implicitamente che in ogni caso B possa mandare pacchetti ad A con successo. Questo può non essere sempre auspicabile. Mentre si potrebbe desiderare di contattare un sito web per leggere una pagina, si potrebbe ritenere non altrettanto lecito che il server web inicializzasse di sua spontanea volontà una connessione verso il mio computer (in cerca di che cosa?).

A.1.2.3 Regole di stato.

Le regole di stato considerano che *l'utente ha consentito che il primo pacchetto che inizia la connessione sia uscito dal firewall e associano a questa connessione una tabella*, in modo tale che *i pacchetti di ritorno nella direzione opposta* siano riconosciuti come *correlati* al primo pacchetto. Ad esempio siano N1:p1 indirizzo e porta associati al pc 1, N2 e p2 quelli associati al pc 2. La regola di stato che consente

N1:p1 -> N2:p2

consentirà anche il traffico

N2:p2 -> N1:p1,

almeno fintanto che le tabelle di stato non saranno andate in timeout dopo la fine della connessione. Si osservi che, dopo lo scadere dei citati timeout, non saranno più accettati pacchetti provenienti da **N2:p2** (sorgente), poiché non c'è alcuna regola che consenta **N2:p2** come indirizzo e porta sorgenti!

Si consiglia di prendere dimestichezza con le regole di stato e con il loro utilizzo leggendo il documento web <http://www.giacomos.it/ipfirw/stateful.html>.

A.1.2.4 Opzioni specifiche del protocollo TCP.

Essendo *TCP* un protocollo *orientato alla connessione*, esso memorizza nei suoi pacchetti informazioni concernenti la connessione in alcuni campi detti *flag*. Anche questi flag possono essere controllati da IPFIRE-wall.

Segue la lista dei campi che caratterizzano un pacchetto TCP e la loro spiegazione in vista anche di un possibile utilizzo nel contesto di un filtro di pacchetti.

SYN

Il flag SYN viene utilizzato per instaurare una nuova connessione.

Quando un computer desidera iniziare una nuova connessione, esso invia un pacchetto con questo flag a uno (i flag sono costituiti da un solo bit e pertanto possono assumere solo i valori zero e uno). Al contempo, il flag ACK deve essere a zero. La risposta ad una nuova connessione accettata è caratterizzata da entrambi i flag SYN e ACK a uno. In pratica, SYN a uno denota un messaggio di tipo *richiesta di connessione* oppure *connessione accettata*.

Uno scenario interessante è costituito dalla presenza di regole *di stato* e da una regola che vieti esplicitamente i pacchetti con il flag SYN impostato e il flag ACK a zero. In questo caso, sarebbe possibile osservare colorati di rosso tutti i tentativi di creazione di una nuova connessione.

ACK

In aggiunta alla funzione descritta sopra, il flag ACK indica che il campo *acknowledgement number* nel pacchetto TCP è valido. Se ACK vale zero, tale campo è ignorato.

PSH

Il flag PSH (*Push*) indica dati di tipo *push*. In questo modo, il trasmettitore è richiesto di consegnare il pacchetto all'applicazione al momento d'arrivo, senza salvarlo prima in una memoria temporanea. Probabilmente, per ragioni di efficienza, il salvataggio in una memoria temporanea (*buffer*) sarebbe inevitabile.

RST

Il campo RST (*Reset*) viene utilizzato per riinizializzare una connessione divenuta instabile per un problema all'host o per qualche altro motivo. Viene anche adoperato per rifiutare un pacchetto TCP malformato o una nuova connessione. Se si riceve un reset, si è certi che da qualche parte si è verificato un problema.

FIN

Il flag FIN e' impostato al valore uno per chiudere una connessione e specifica che il trasmittente non ha altri dati da inviare.

URG

Il campo URG (*Urgent pointer*) indica che è presente uno scostamento (*offset*) in bytes, dal numero di sequenza, a partire dal quale il ricevente dovrebbe trovare dati urgenti. Per il significato dei numeri di sequenza, si veda la bibliografia.

A.1.2.5 Altri campi delle regole.

A.1.2.5.1 Dispositivo di rete

IPFIRE-wall può filtrare pacchetti anche in dipendenza del dispositivo di rete dal quale essi sono in arrivo o da cui sono in partenza.

Nella direzione di *INPUT*, *PREROUTING* o *FORWARD* si può specificare un *dispositivo d'ingresso*, mentre in *OUTPUT*, *POSTROUTING* o ancora in *FORWARD* è possibile indicare un *dispositivo d'uscita*.

Se nessun dispositivo di rete è presente in una regola, allora il *kernel* non controllerà il nome del dispositivo coinvolto nel pacchetto considerato.

A.1.2.5.2 Assegnare un nome ad una regola.

L'utente, al momento dell'inserimento di una nuova regola in *IPFIRE-wall*, può assegnare a quest'ultima un semplice nome (di venti caratteri al massimo), che verrà stampato dalla console quando un pacchetto corrisponde alla regola stessa. Questo rende possibile all'utilizzatore ricordare facilmente la regola che viene utilizzata, senza dover ispezionare attentamente tutti i campi tipici del protocollo di rete per risalire al significato della norma applicata.

A.2 Bibliografia.

Si veda l'ottimo testo di *Andrew S. Tanenbaum* “*Reti di Computer*” (“*Computer Networks*”), Terza edizione, per una trattazione chiara e al contempo approfondita circa le reti di computer.

A.2 Bibliografia.

Indice generale

Introduzione.....	2
Cos'è IPFIRE-wall.....	2
Cosa non è IPFIRE-wall.....	3
Attenzione.....	5
1. Installazione del software.....	6
1.1 Ottenere il pacchetto per l'installazione.....	6
1.2 Installazione di IPFIRE-wall.....	6
1.2.1 Estrazione dei file compressi.....	6
1.2.2 Installazione automatica.....	8
1.2.2.1 Installazione da root.....	10
1.2.2.2 Installazione da utente non privilegiato.....	11
1.2.3 Installazione manuale.....	12
1.2.3.1 Compilazione e installazione dei moduli del kernel.....	12
1.2.3.2 Compilazione dell'interfaccia utente e installazione dei file.....	13
1.2.3.3 Compilazione dell'analizzatore dei log, analyzer.....	15
1.2.3.3 Conclusioni sull'installazione.....	15
1.3 Prerequisiti del sistema per l'installazione di IPFIRE-wall.....	15
1.3.1 Compilazione dei sorgenti.....	15
1.3.2 Ottenere e installare i sorgenti del kernel.....	16
1.3.3 Versioni del kernel supportate.....	17
1.3.4 Risoluzione dei problemi.....	17
1.3.4.1 Esempio di problema risolto.....	17
1.3.5 Rimozione del software.....	17
2. Esecuzione di IPFIRE-wall.....	19
2.1 Avvio di IPFIRE-wall.....	19
2.2 Arresto del filtro di pacchetti.....	20
2.2.1 Arresto dell'esecuzione dell'interfaccia utente.....	20
2.2.2 Rimozione del modulo del kernel (richiede i privilegi di root).....	21
2.2.3 Osservazione importante.....	22
2.2.4 Interagire con IPFIRE-wall tramite il menù principale.....	22
2.2.5 Conclusioni.....	25
Appendice.....	27
A.1 Introduzione al protocollo internet TCP/IP.....	27
A.1.1 La rete e l'instradamento dei pacchetti in Linux (cenni).	27
A.1.2 Regole di IPFIRE-wall e protocollo di rete TCP/IP.....	30
A.1.2.1 Identificazione di una macchina in rete e di un servizio al suo interno.....	30
A.1.2.2 Affidabilità della connessione di rete e servizi orientati alla connessione.....	30
A.1.2.3 Regole di stato.....	32
A.1.2.4 Opzioni specifiche del protocollo TCP.....	33

A.2 Bibliografia.

A.1.2.5 Altri campi delle regole.	34
A.1.2.5.1 Dispositivo di rete.....	34
A.1.2.5.2 Assegnare un nome ad una regola.....	34
A.2 Bibliografia.....	35