Manuale Turtle Firewall

Andrea Frigido
Friweb snc

Manuale Turtle Firewall

Andrea Frigido

Pubblicato 2002

Diario delle revisioni

Revisione 1.0 30 Mar 2002

Revisione 1.1 17 Dic 2002

Sommario

Capitolo 1. Introduzione

1.1. Cos'è Turtle Firewall

Turtle Firewall è un software per la configurazione di firewall basati su netfilter, quindi compatibile con le versioni del kernel Linux 2.4.x e 2.5.x.

Turtle firewall si attiva come un normale servizio Linux durante il boot e la sua configurazione è resa estremamente semplice dall'interfaccia web implementata come modulo per webmin.

1.2. Requisiti

- Linux kernel 2.4.x o 2.5.x.[1]
- iptables raggiungibile con il path predefinito.
- moduli netfilter, compresi ip_conntrack, ip_conntrack_ftp e ip_nat_ftp, compilati nel kernel o caricati con il comando insmod.
- Perl 5.6[2].
- Liberia expat (normalemente presente in tutte le distribuzioni Linux).
- Modulo perl XML::Parser[3].
- Webmin[4].

Note

1. http://www.kernel.org
2. http://www.perl.com
3. http://search.cpan.org
4. http://www.webmin.com

Capitolo 2. Elementi di un sistema protetto da firewall

2.1. Elementi del firewall

Un firewall è una macchina dotata di più interfaccie di rete connesse a diversi rami di differenti reti. Ogni interfaccia permette al firewall di comunicare con la zona a cui è connessa l'interfaccia. La zona può essere costituita da una o più sottoreti o addirittura da tutta la rete Internet.

Con le sue N interfacce il firewall può comunicare con N zone. Se un host di una zona vuole comunicare con un host di un'altra zona sarà costretto ad attraversare il firewall. Se esiste un percorso alternativo significa che abbiamo sbagliato a costruire la nostra rete poichè il firewall risulta inefficace.

Detto questo, nel firewall possiamo inpostare una serie di regole per determinare quali pacchetti debbano passare da una zona all'altra e queste regole possono tener conto di una serie di caratteristiche del pacchetto da filtrare: zona/sottorete/host di provenienza, zona/sottorete/host di destinazione, tipo di protocollo (tcp/udp/icmp), porta di destinazione e provenienza, ecc.

Con Turtle Firewall è possibile definire gli elementi che possono essere origine o destinazione di una connessione, assegnargli un nome e in seguito usare il nome per definire le regole del firewall.

Per visualizzare l'elenco degli elementi del firewall si deve cliccare sull'icona *Elementi* del menù principale del modulo Turtle Firewall di Webmin. Divisi in quattro tabella si possono vedere gli elementi di tipo ZONA (zone), RETE (net), HOST e GRUPPO (group).

2.2. Zona (Zone)

Una zona è definita da un nome arbitrario e dall'interfaccia di rete che usa il firewall per comunicare con gli hosts che si trovano in quella zona.

Una comune rete con firewall è normalmente costituita da 3 zone: la rete interna (comunemente chiamata *good* perchè ci stiamo noi, i buoni :-)) a cui assicurare un elevato livello di sicurezza, la rete esterna che di solito è Internet (comunemente chiamata *bad* perchè li fori ci stanno i cattivi che desiderano *"sicuramente"* entrare nella nostra zona *good* ;-( ) e in fine la zona dove abbiamo i nostri servers pubblici, come ad esempio il web server, che non può essere completamente blindata come la zona good perchè dobbiamo rendere accessibili i servizi pubblici agli utenti Internet (viene comunemente chiamata dmz, demilitarized zone). Naturalmente una rete può essere anche molto piu' complessa e contare quindi innumerevoli zone dmz o good ma le regole da seguire per impostare un buon firewall sono sostanzialmente le stesse.

Cliccando su *"Crea una nuova zona"* ci verrà visualizzato un breve modulo con il quale potremo indicare il nome che vogliamo assegnare alla nuova zona (good, bad, dmz, mia_zona, ecc.) e l'interfaccia di rete da utilizzare per raggiungerla (eth0, eth1, ppp0, ecc.).

Cliccando quindi sul bottone Crea registreremo in modo permanente la nuova zona.

In modo analogo, cliccando sul nome di una zona, avremo la possibilità di modificarla variando l'interfaccia di rete alla quale è associata.

2.3. Rete (Net)

Una rete è identificata da un indirizzo di rete e da una netmask che determina la dimensione della rete.

La netmask si può esprimere con 4 numeri (0-255) separati dal punto (es. 255.255.255.0) oppure semplicemente dal numero di bit più significativi da consideare a 1 (es. 24).

Analogamente a quanto abbiamo visto per le zone, anche per creare una nuova rete è necessario cliccare su *"Crea una nuova rete"*.

A questo punto potremo dare un nome alla rete (es. rete_interna, sottorete1, ecc.), indicare un indirizzo di rete (es. 192.168.0.0), indicare la netmask della rete (es. 255.255.0.0) e, per finire, indicare la zona alla quale questa sottorete appartiene (es. good).

Sia chiaro che prima di poter definire una rete deve essere stata dichiara la zona alla quale la rete appartiene.

2.4. Host

Un host è un singolo computer o dispositivo (router o altro) raggiungibile dal nostro firewall. Un host ha quindi un indirizzo *(ip)* e fa parte di una zona *(zone)*.

Per creare un nuovo host si deve cliccare su *"Crea un nuovo host"*, assegnargli un nome (es. PC_Andrea, host1, WebServer, ecc.), indicare un indirizzo IP (es. 192.168.1.123) e, in fine, indicare a che zona appartiene l'host.

2.5. Gruppo (Group)

Con questo particolare elemento si può raggruppare una serie di elementi, precedentemente definiti, che possono essere delle zone, delle reti o degli host. Un Gruppo è identificato da un nome e può essere usato nelle regole del firewall come un normale elemento del sistema. Questo semplifica molto la definizione di regole comuni a più hosts (o reti o zone) rendendo più sintetica, leggibile e mantenibile la configurazione del firewall. Se, ad esempio, vogliamo dare l'accesso ssh verso un server ai nostri 3 amministratori di sistema, sarà sufficiente creare un gruppo chiamto *"amministratori"* composto dai 3 host degli amministratori e poi applicare un'unica regola ssh all'intero gruppo.

Per creare un nuovo Gruppo si deve cliccare su *"Crea un nuovo gruppo"*, assegnargli un nome (es. servers, host_privilegiati, ecc.) e, in fine, indicare attraverso l'attivazione di una serie di check-box quali elementi faranno parte del gruppo che stiamo definendo.

2.6. L'elemento FIREWALL

Turtle Firewall considera, oltre alle zone definite da noi, un'ulteriore zona predefinita chiamata FIREWALL che identifica la nostra linux-box dove abbiamo installato il firewall. Tutti i pacchetti che non sono in transito sul firewall ma che invece partono da, o sono destinati al firewall stesso, implicano l'uso della zona FIREWALL. FIREWALL non è un host perchè il firewall non ha un solo ip (ne ha uno per ogni interfaccia) ed è quindi più opportuno considerarlo una zona a se stante.

E' estremamente importante definire regole per la protezione del firewall. Il buon senso ci dovrebbe spingere a rendere completamente isolato il firewall negando l'accesso alla zona FIREWALL a chiunque. Spesso però vogliamo poter lavorare sulla configurazione del firewall comodamente seduti alla nostra postazione (la pigrizia è la causa più frequente di compromissione di un sistema di sicurezza ;-)) e allora consiglio di rendere disponibile il minor numero possibile di servizi al minor numero possibile di host, il solo servizio ssh al solo host dell'amministratore di rete è un buon compromesso.

Nota: Turtle Firewall non permette di definire 2 elementi con lo stesso nome, anche se di tipo diverso. Non ci può essere, ad esempio, un host che ha lo stesso nome di una zona. Per lo stesso motivo non è possibile definire un elemento che si chiami "FIREWALL".

Capitolo 3. Regole del firewall

La policy di base di Turtle Firewall è quella di vietare per default ogni tipo di connessione intercettabile dal firewall. A questa politica di base si aggiungono poi delle regole che determinano quali tipi di connessioni vogliamo permettere. Questa non è l'unica logica applicabile su un firewall ma è quella generalmente considerata più prudente e quindi è quella che ho scelto per Turtle Firewall.

Per permettere una connessione elementare tra 2 host (A, B), prendiamo per esempio una connessione http, dobbiamo definire 2 regole di filtro: una che permette il passaggio dei pacchetti che vanno dall'host A all'host B e una che permette il ritorno dei pacchetti di risposta da B ad A.

Turtle Firewall semplifica questa operazione definendo dei servizi per i quali si fa carico di impostare tutte le regole di filtering che sono necessarie per garantire quel servizio. In questo modo la regola di filtro di Turtle Firewall è più semplice da usare perchè si basa su un servizio internet anzichè sulle caratteristiche dei singoli pachetti. Per indicare che vogliamo permettere all'host A di comunicare con l'host B attraverso il protocollo http basterà definire una sola regola Turtle Firewall che specifica il nome dell'elemento sorgente (A), il nome dell'elemento destinazione (B) e il servizio da usare (http). Sia chiaro che se definisco una regola per il servizio http da A a B, questo non significa che anche B può usufruire dello stesso servizio verso A perchè B può solo accettare e rispondere alle richieste http che provengono da A.

Per visualizzare l'elenco delle regole attive bisogna cliccare sull'icona "Regole" del menù principale del modulo Turtle Firewall.

Per creare una nuova regola si deve selezionare "Crea una nuova regola". A questo punto bisogna indicare l'elemento sorgente della comunicazione, l'elemento destinatario, il servizio da usare (es. http, ssh, ftp, ecc.), un'eventuale porta (solo per i servizi generici come tcp o udp) e, per finire, selezionare il check-box "Attiva" che rende per l'appunto attiva la regola.

Capitolo 4. Regole NAT (Network Address Translation) e Mascheramento

4.1. Introduzione

Con i nuovi kernel 2.4.x e iptables la gestione del NAT e del mascheramento avviene all'interno del modulo netfilter del kernel. Con le regole NAT possiamo cambiare l'elemento sorgente o l'elemento destinatario di una connessione.

Questo è utile quando vogliamo dirottare tutto il traffico destinato ad un host verso un altro host. Se ad esempio, il nostro provider ci ha assegnato alcuni indirizzi pubblici ma il nostro web server si trova in una dmz con un indirizzo riservato, per rendere accessibile il web server all'esterno ci basterà impostare una regola NAT che ridirezioni tuttle le connessioni destinate all'indirizzo pubblico, che abbiamo scelto per il nostro web server, verso l'indirizzo privato (reale) del nostro webserver. Il client che si trova all'altro capo della connessione non si accorgerà di nulla e crederà di comunicare effettivamente con un webserver il cui indirizzo IP è pubblico.

E' possibile anche specificare a quale servizio vogliamo applicare il NAT. In questo modo possiamo impostare Turtle Firewall in modo che smisti il traffico ricevuto su un indirizzo IP pubblico verso differenti server serali, in funzione del servizio utilizzato. Avremo quindi un unico indirizzo IP pubblico che potrà rendere disponibili una serie di servizi (http, ftp, smtp, pop3, ecc.) che poi verranno gestiti da differenti server interni (Web Server, Ftp Server, Mail Server, ecc.).

Il mascheramento è un caso particolare di NATting e si applica quando vogliamo che le connessioni che da una zona arrivano ad un'altra zona appaiano a quest'ultima come provenienti dal nostro firewall. Il caso più consueto è quello delle connessioni verso internet: non vogliamo che gli host interni vengano identificati dagli host di internet e così diciamo al nostro firewall di mascherare tutte le comunicazioni verso l'esterno. Per gli host internet la connessione apparirà provenire solo dal firewall.

In questa sezione è anche possibile definire delle regole di Redirezione local, essenziali per la realizzazione di un Transparent Prixy.

Per visualizzare l'elenco delle regole di NAT, Mascheramento e Redirezione bisogna cliccare sull'icona "NAT e Mascheramento" del menù principale del modulo Turtle Firewall.

4.2. Nat (Network Address Translation)

Per creare una nuova regola NAT si deve selezionare *"crea una nuova regola NAT"*. A questo punto è necessario indicare l'host virtuale (normalmente un host con ip pubblico), l'host reale (normalmente un host che si trova in dmz) e il servizio per il quale applicare la regola NAT.

Gli elementi degli host indicati come "Host virtuale" e "Host reale" devono, naturalmente, essere precedentemente definiti.

Come host virtuale è anche possibile indicare una zona, in questo caso particolare il sistema considera come indirizzo dell'host virtuale qualsisti indirizzo IP assegnato all'interfaccia di rete connessa con la zona indicata. E' necessario adottare una soluzione simile nelle situazioni in cui il nostro IP pubblico è dinamico, ad esempio nel caso di una connessione ppp.

4.3. Mascheramento (Masquerade)

Il mascheramento prevede che tutti i pacchetti che, attraverso il firewall, escono verso una determinata zona appaiano come inviati dal firewall stesso.

Per creare una nuova regola di Mascheramento si deve selezionare "Crea nuovo Mascheramento". A questo punto è sufficiente indicare la zona sulla quale applicare il mascheramento dei pacchetti.

Nota: Questa funzionalità è attualmente limitata ;-(, in futuro conto di estenderla introducendo differenti mascheramenti in funzione degli host di origine e destinazione.

4.4. Redirezione su porta locale

Le regole di Redirezione permettono di catturare i pacchetti tcp/ip in transito sul Firewall e redirigerli ad una porta locale del Firewall stesso. Il client dal quale provengono i pacchetti crederà di comunicare con l'host desiderato, senza accorgersi che la comunicazione viene invece catturata e gestita dal Firewall. Anche i pacchetti tcp/ip di risposta appariranno al client come provenienti dall'host contattato.

Questa è una funzionalità essenziale per la realizzazione di un *Transparent Proxy*. Un Transparen Proxy è u proxy cache server, come ad esempio Squid, che gestisce in modo automatico la navigazione dei suoi client. I client non sanno di utilizzare un Proxy, richiedono normalmente le loro connessioni http ed ftp utilizzando il Firewall come gateway Internet. E' a questo punto che il Firewall, grazie alla redirezione della comunicazione, può inviare le richieste al Proxy Server in esecuzione sulla stessa macchina.

Per creare una nuove regola di redirezione è necessario cliccare su *"crea una nuova Redirezione"*. Le informazioni da indicare sono: Sorgente, Destinazione, Servizio e Porta locale.

Sorgente è l'elemento dal quale proviene la comunicazione da redirezionare. Può essere uno qualsiasti degli elementi del Firewall. Grazie a questa opzione è possibile definire una redirezione selettiva che ha effetto solo su determinate sorgenti della connessione.

Destinazione è l'obiettivo della connessione, può essere uno qualsiasi degli elementi del Firewall escluse le zone. Indicando * si ottiene il redirezionamento per tutte le destinazioni possibili.

Servizio è il servizio tcp/ip per il quale applicare la redirezione. Potrebbe essere, ad esempio, il servizio http.

Porta locale è la porta del Firewall verso la quale verrà dirottato il servizio. Nel caso di un Proxy potrebbe essere la porta 3128 oppure la 8080.

Capitolo 5. I servizi

5.1. Introduzione

Abbiamo già visto in precedenza che Turtle Firewall utilizza regole basate sui servizi (http, ftp, ecc.). Questi sono dei servizi predefiniti contenuti nel file */etc/turtlefirewall/fwservices.xml*.

E' comunque possibile definire i propri servizi da affiancare a quelli predefiniti. Per definire nuovi servizi non esiste interfaccia web, bisogna quindi operare direttamente sul file */etc/turtlefirewall/fwuserdefservices.xml*, riservato alla definizione dei servizi personalizzati.

Ora vedremo come definire i nostri servizi per poterli poi attivare nel Firewall. Il file *fwuserdefservices.xml* deve essere scritto seguendo una codifica XML che permette di definire le caratteristiche che ogni pacchetto tcp/ip deve possedere per essere considerato valido per lo specifico servizio. La logica da seguire per definire un servizio è simile a quella usata per definire una regola del firewall. In sostanza si tratta di dichiarare un certo numero di filtri che permettono il passaggio dei pacchetti, se un pacchetto non è conforme a alcun filtro verrà respinto.

Il file XML di definizione dei servizi deve avere il tag radice *services* (<services>) che contiene al suo interno i tags *service* che definiscono il singolo servizio. Il tag *service* ha gli attributi name e description per definire rispettivamente il nome e una breve descrizione del servizio. All'interno del tag *service* vanno posti filtri *filter* che indicano come vengono selezionati i pacchetti validi per il servizio.

Il tag filter

Il tag filter possiede una serie di attributi che vengono usati per definire le caratteristiche che il pacchetto deve possedere per essere considerato valido. Se il paccetto non rientra nei parametri definiti dal filtro allora il controllo passa al filto successivo e così via fino all'ultimo filtro. Se nessun filtro risulta verificato il pacchetto viene respinto.

Attributi:

- *direction*: definisce la direzione del pacchetto in transito. Può assumere il valore *go* o il valore *back*. Con go vengono considerati i soli pacchetti di andata da un ipotetico host sorgente ad un host destinazione, con back si considerano invece i pacchetti di ritorno.
- *p*: protocollo, può assumere uno dei seguenti valori: tcp, udp o icmp.
- *sport*: porta usata dal socket dell'host sorgente.
- *dport*: porta di destinazione del pacchetto.
- *icmptype*: tipo di messaggio ICMP (da usare solo con p="icmp", vedi doc. iptables).
- *state*: stato del pacchetto (vedi direttiva state di iptables).
- *jump*: salto forzato ad una catena o ad un esito.

  Può assumere i seguenti valori: *ACCEPT* (il pacchetto viene accettato, è la catena di default se direction è *go*), *DROP* (il pacchetto viene considerato non valido, di solito non si usa mai direttamente questo jump perchè non viene segnalato sul file di log), *BACK* (il controllo passa ad una speciale catena che lascia passare solo i pacchetti di ritorno di connessioni già aperte, è la catena di default nel caso sia impostato l'attributo direction a *back*), *ICMP-ACC* (altra speciale catena da usare con p="icmp", lascia passare solo i messaggi icmp standard considerati sicuri).

Nota: Se come sport o dport si indica "PORT" allora la porta verrà impostata durante la generazione dello script al valore indicato dall'attributo *port* del tag *rule* della regola del firewall. In questo modo si possono definire servizi parametrizzabili.

Sintassi:

```
<filter direction="go/back" p="tcp/udp/icmp" sport="nPortaSrc" dport="nPortaDst"
  icmptype="tipoMsgIcmp" jump="ACCEPT/DROP/BACK/ICMP-ACC"/>
```

Esempio 5-1. Definizione di 3 servizi

```
<services>
  <service name="http" description="Servizio www o http">
    <filter direction="go" p="tcp" dport="www"/>
    <filter direction="back" p="tcp" sport="www"/>
  </service>

  <service name="tcp" description="Servizio TCP generico">
    <filter direction="go" p="tcp" dport="PORT"/>
    <filter direction="back" p="tcp" sport="PORT"/>
  </service>

  <service name="ping" description="icmp message echo-request and echo-
reply">
    <filter direction="go" p="icmp" ICMPTYPE="echo-request"/>
    <filter direction="back" p="icmp" ICMPTYPE="echo-reply"/>
  </service>
</services>
```

5.2. Servizi Predefiniti

Il file *fwservices.xml* è il file di definizione dei servizi usato da Turtle Firewall. Contiene la definizione dei servizi più comunemente usati su internet e può essere ovviamente essere integrato con i propri servizi specifici modificando direttamente il codice XML.

Servizi predefiniti

- *all*: Permette ogni tipo di accesso (USARE CON CAUTELA)
- *tcp*: Servizio TCP generico
- *udp*: Servizio UDP generico
- *ftp*: File Transfer Protocol
- *dns*: Domain Name Service
- *www*: Servizio www o http
- *http*: Servizio www o http
- *https*: HTTP + SSL
- *auth*: Servizio di autenticazione
- *smtp*: Simple Mail Transfer Protocol
- *pop3*: Servizio per il download dei messaggi e-mail con POP3
- *imap*: Servizio per il download dei messaggi e-mail con IMAP
- *ssh*: Secure Shell
- *ntp*: Networs Time Protocolo
- *icmp_acc*: Filtra i messaggi icmp pericolosi lasciando passare quelli standard
- *ping*: icmp message echo-request and echo-reply
- *netbios_ns*: Servizio NETBIOS Name Service

- *netbios*: Servizio NETBIOS Completo
- *netbios_ssn*: Servizio NETBIOS Sessions Service
- *cvs*: CVS Server Service
- *nntp*: NNTP Network News Transport Protocol
- *telnet*: Telnet Protocol
- *webmin*: Webmin (port 10000)
- *h323*: H323 Protocol (NetMeeting), sperimentale
- *ipsec-ESP*: VPN IPSec protocol con IKE e ESP
- *ipsec-AH*: VPN IPSec protocol con IKE e AH
- *ipsec-ESP-AH*: VPN IPSec protocol con IKE, ESP e AH
- *afp-over-tcp*: AFP (Apple Filing Protocol) over TCP
- *nfs*: NFS, sperimentale
- *mysql*: MySQL-Server
- *kazaa*: KaZaa Filesharing
- *pptp*: PPTP VPN Service
- *rdp*: Remote Desktop Protocol
- *pc-anywhere*: PC-Anywhere service (da admin agli hosts)
- *x11*: X Window System service (dal client all'XServer)