Firewalling con Linux

Alessandro Barenghi

June 4, 2009

Summary

Cos'è/A cosa serve un firewall?

- Applicazione che controlla il flusso di dati da/per la rete
- Decide in base a informazioni sul flusso di dati se accettare o rifiutare i pacchetti
- Può essere integrata all' interno del kernel di un sistema operativo oppure essere un programma in userspace

Dove si trova un firewall?

- Il firewall deve essere l' *unico* punto di contatto tra le macchine al sicuro e la rete insicura
- ... ovvero, se la macchina è una sola finisce direttamente su quella
- Se si protegge una rete locale, l' unico punto di passaggio verso l' esterno deve essere il firewall

Dove si trova un firewall?

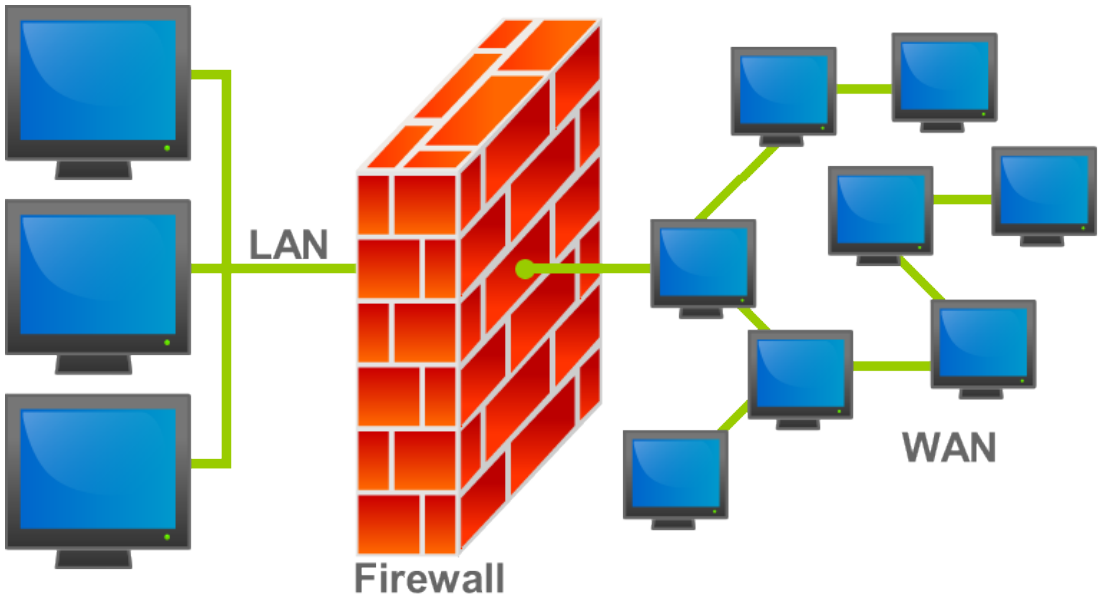

Figure: Firewall placing

Netfilter : cos'è?

- In ambiente Linux, il firewall è parte integrante del kernel
- Il vantaggio principale è che è agganciato esattamente alle funzioni che gestiscono lo stack ISO/OSI
- Per controllare il funzionamento del firewall sono disponibili diversi programmi in userspace
- Impareremo a usare iptables, l' interfaccia più diretta, le altre sono semplicemente zucchero sintattico

Tabelle

- Il sistema di gestione pacchetti di Iptables offre 5 punti d' aggancio (tabelle) standard
- Ogni tabella contiene una lista di regole con la rispettiva condizione di attivazione
- Per ogni pacchetto netfilter tenta di applicare le regole in ordine
- Nel caso nessuna delle regole venga attivata, è presente una prassi (Policy) che viene attuata

Input-Output-Forward

- Input, output e forward sono le 3 tabelle principali riguardanti i pacchetti che sono già stati instradati
- Input contiene la lista delle regole da applicare ai pacchetti in entrata nella macchina
- Output contiene la lista delle regole da applicare ai pacchetti in uscita dalla macchina
- Forward contiene la lista delle regole da applicare ai pacchetti inoltrati dalla macchina senza essere toccati

Prerouting e Postrouting

- Prerouting e postrouting sono le tabelle che si occupano dei pacchetti che devono ancora essere instradati ...
- ... o il cui instradamento deve essere corretto dopo che sono stati modificati
- Prerouting agisce su tutti i pacchetti che entrano nella macchina , indipendentemente dalla destinazione
- Postrouting agisce su tutti i pacchetti che escono dalla macchina , indipendentemente dalla sorgente.

Mappa

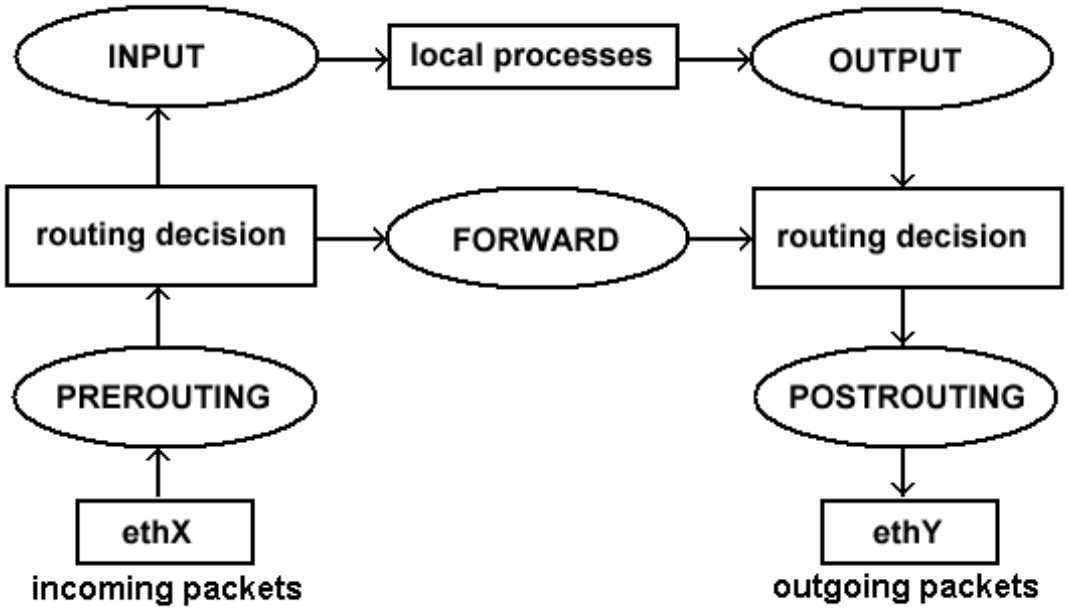

Figure: Netfilter Hook Map

Regole

- In ogni tabella elencata si possono aggiungere regole che consentono o bloccano il passaggio di pacchetti
- I criteri più comuni sono indirizzo/porta di destinazione/sorgente o tipo di protocollo
- Iptables consente anche di modificare il contenuto dei pacchetti o di effettuare logging ogni volta che scatta una regola

Source NAT

- La Source Network Address Translation consente di utilizzare un solo ip pubblico per mascherare una rete
- Una singola macchina si occupa di "fingersi" la sorgente delle comunicazioni e agisce da concentratore
- Dall' esterno le connessioni sembrano partire dalla macchina gateway (l' indirizzo sorgente viene cambiato)
- L' intera rete mascherata dal NAT risulta invisibile [1] dall' esterno

[1] o quasi

Destination NAT e NPT

- La Destination Network Address Translation (DNAT) consente di cambiare la destinazione di un canale TCP
- È in qualche modo il simmetrico della SNAT : una sola macchina smista le richieste dall' esterno su oggetti diversi all' interno
- Utile se avete più macchine in casa , una sola connessione e volete accedere a tutte
- Implicitamente viene operata anche una Network Port Translation (per ragioni di disambiguazione)

Schema Grafico

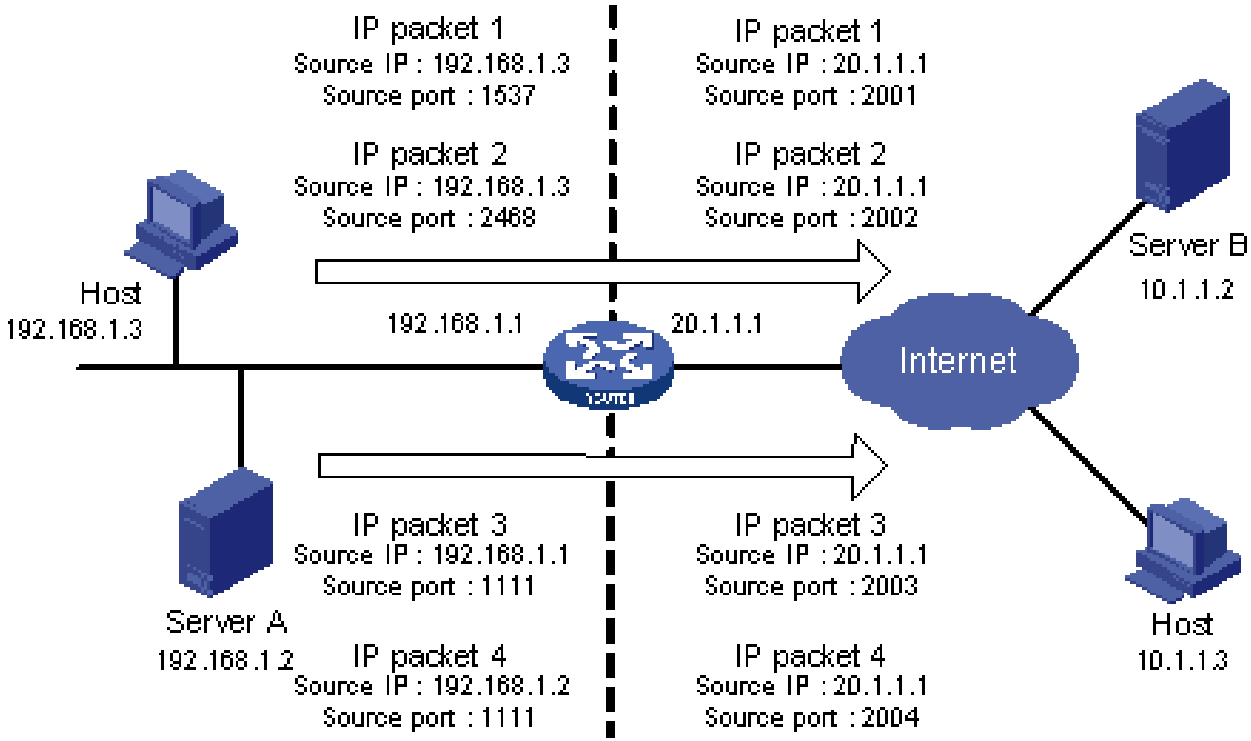

Figure: Struttura di una NA(P)T

Casi di studio

- Finalmente un po' di pratica : affrontiamo 3 configurazioni tipiche in ordine di difficoltà crescente :
  - Macchina singola: firewalling di una macchina direttamente connessa
  - Gateway con Source NAT: realizzazione di un semplice gateway per casa
  - Destination NAT: quando la rete di casa inizia a complicarsi

Macchina Singola

- Obiettivi: tutto chiuso , salvo qualche servizio che vogliamo offrire
- Assumiamo che la macchina abbia una sola scheda di rete
- La configurazione fornirà la base per le due successive

Step 1 - Policies

- Prima di tutto impostiamo le policies tramite la regola : `iptables -P <TABLE> <POLICY>`
- Di prassi la tabella di INPUT e di FORWARD vanno impostate con policy DROP (scarta il pacchetto)
- La tabella di output può essere impostata allo stesso modo, se necessario oppure lasciata in ACCEPT
- La prima scelta è la più sicura, ma spesso è fonte di difficoltà di configurazione [2] inutili

[2] quasi

Step 2 - loopback

- Impostate le policy iniziamo a occuparci di quello che vogliamo accettare
- La prima cosa che dobbiamo lasciar passare incondizionatamente[3] è l' interfaccia di loopback
- Usiamo una regola fatta così :`iptables -A <TABELLA> <condizioni> -j <AZIONE>`
- Usando la condizione `-i <interfaccia>` che indica l' interfaccia diventa
- `iptables -A INPUT -i lo -j ACCEPT`

[3] al solito , a meno di insoliti trick

Step 3 - Connessioni stabilite

- A questo punto la nostra macchina è perfettamente sicura^{TM}.[4].
- Questo grazie al fatto che è perfettamente sigillata.
- Volendola utilizzare, il primo passo da fare è autorizzare le connessioni provenienti dall' interno
- Nel caso la policy della tabella di OUTPUT sia ACCEPT non c'è altro da fare
- Vogliamo anche che i pacchetti di risposta alle connessioni in uscita siano autorizzati a entrare

[4] Ok , *quasi* perfettamente sicura^{TM}

Step 3 - Connessioni stabilite

- Per ottenere tutto questo facciamo uso dell' opzione di iptables che controlla lo stato della connessione : `-m state`
- La regola che autorizza le connessioni in uscita diventa :
- `iptables -A OUTPUT -m state --state NEW,RELATED,ESTABLISHED -j ACCEPT`
- Quella che autorizza il rientro dei pacchetti delle connessioni vive è :
- `iptables -A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT`

Step 4 - Porte Aperte

- Applicate le regole precedenti la macchina è pronta per un uso normale senza esporre servizi
- Nel caso vogliate esporre un ftp server, un client BitTorrent o serve specificare l' apertura di una porta
- Questo viene realizzato con una regola che matcha la porta di destinazione : opzione `--dport`
- È possibile anche specificare un protocollo specifico tramite l' opzione `-p [tcp|udp|icmp]`
- Per aprire la porta 22 (SSH) per esempio la regola è :
- `iptables -A INPUT -p tcp --dport 22 -j ACCEPT`

Step 5 - ICMP

- Onde evitare malfunzionamenti in alcuni comodi tool di rete, è opportuno accettare anche alcuni pacchetti ICMP
-
```
iptables -A INPUT -p icmp --icmp-type
destination-unreachable -j ACCEPT
iptables -A INPUT -p icmp --icmp-type
source-quench -j ACCEPT
iptables -A INPUT -p icmp --icmp-type
time-exceeded -j ACCEPT
iptables -A INPUT -p icmp --icmp-type
echo-request -j ACCEPT
iptables -A INPUT -p icmp --icmp-type echo-reply
-j ACCEPT
```

Macchina Singola

- Abbiamo completato la configurazione di una macchina singola che può:
  - Navigare liberamente (connessioni in uscita sbloccate)
  - Operare liberamente sul loopback locale (127.0.0.1)
  - Ha alcune porte aperte verso l' esterno a seconda di cosa vogliamo esporre
  - Cestina qualunque altra cosa
  - Usi : avere una macchina con solo le porte necessarie aperte

Due interfacce

- La configurazione come gateway di una macchina prevede (almeno) 2 interfacce di rete
- Per convenzione eth0 sarà quella rivolta all' interno e eth1 quella rivolta all' esterno
- È possibile specificare le regole per una sola interfaccia con il flag `-i <interfaccia>`
- Il gateway dovrà far passare i pacchetti destinati all' esterno nella sua catena di forward
- Al momento dell' uscita sarà necessario effettuare il Source NAT

Forwarding e Masquerading

- Per ogni host/sottorete a cui vogliamo fare da SNAT servono 2 regole
- La prima per accettare i suoi pacchetti aggiunta alla tabella di forward
- `iptables -A FORWARD -i eth0 -s <ipsorgente> -j ACCEPT`
- La seconda per effettivamente mascherare l' indirizzo (aggiunta alla tabella POSTROUTING)
- `iptables -t nat -A POSTROUTING -o eth1 -s <ipsorgente> -j MASQUERADE`

Forwarding e Masquerading - 2

- L' unica cosa che ci resta da fare a questo punto è accettare i pacchetti related
- `iptables -t nat -A POSTROUTING -m state --state RELATED,ESTABLISHED -j ACCEPT`
- A questo punto abbiamo completamente configurato il nostro Source NAT Gateway
- Nel caso non l' abbiate fatto prima , vanno impostate le routing tables :)

Source NAT

- Abbiamo completato la configurazione di SNAT Gateway che può:
  - Fare tutto quello che faceva la macchina singola
  - Mascherare altre macchine con il suo indirizzo di rete in modo opaco
  - Gestire in modo trasparente (per le macchine interne) la connessione
  - Usi : usare una sola connessione per molte macchine
  - Per chiunque avesse una connessione Fastweb : è così che siete connessi alla rete :)

Piccole LAN crescono

- L' ultimo obiettivo è : rendere visibile una porta di una macchina dietro un NAT all' esterno
- Per esempio, volete mantenere un webserver interno a casa o avere accesso diretto a SSH del vostro computer
- Il principio che applicheremo è l' opposto, si tratta di Destination NAT
- Questa volta modifichiamo i pacchetti *prima* che vengano passati agli algoritmi di routing.

Destination NAT

- Quello che vogliamo fare è modificare la destinazione dei pacchetti che arrivano su una determinata porta
- Redirigiamo quei pacchetti verso una macchina interna ben precisa creando un legame tra la porta esterna e quella interna
- La regola necessaria per fare questo è :
- `iptables -t nat -A PREROUTING -i eth1 -m multiport -p tcp --dport 42 -j DNAT --to-destination <ipinterno:porta>`

Source NAT

- Abbiamo completato la configurazione di un S+DNAT Gateway che può:
  - Fare tutto quello che faceva la macchina singola
  - Fare tutto quello che faceva la macchina SNAT
  - Offre all' esterno la possibilità di connettersi a una macchina interna al NAT
  - Usi : Tunnel grafici dall' esterno , esporre server interni

Tips

- Nel caso abbiate sbagliato una regola potete rimuoverla sostituendo all' opzione `-A` l' opzione `-D`
- Per spianare tutte le regole potete usare `iptables -F`
- Il comando precedente non azzera le policy , quindi attenzione a non chiudervi fuori :)
- Il comando `iptables -L` vi offre una lista della configurazione attuale di iptables

HowTos e guide

- Netfilter homepage : `http://www.netfilter.org`
- In particolare la parte di documentazione contiene un HowTo base e uno specifico per il NAT
- Nel caso vogliate una GUI spiccia per macchina singola Gufw è una buona idea