Corso GNU/Linux Avanzato – Uso e configurazione di un firewall usando iptables

© Marco Papa (marco@netstudent.polito.it)

NetStudent
Politecnico di Torino

04 Giugno 2009

Cos'è un firewall?

Definition

Un modulo del Kernel che è in grado di identificare il traffico dati e sottoporlo a delle regole

Modulo del kernel Necessario dal momento che le operazioni devono essere eseguite rapidamente e gli stack di rete sono nel kernel

Identificazione Bisogna raggruppare il traffico in determinate categorie e applicare, per ogni categoria, delle regole. I dati possono essere in una, nessuna o molte categorie.

Applicazione delle regole Si può decidere di cancellare, modificare oppure marcare il traffico appartenente ad una determinata categoria

Cos'è un firewall?

- Il firewall, come modulo software, non è direzionale: tutte le interfacce di rete hanno pari "dignità"
- Le regole, tipicamente, rendono direzionale il firewall
- Un uso classico dei firewall consiste nel proteggere un'*area* interna dal resto della rete.
- Il firewall può essere installato su un router (area interna = una o più reti), su uno switch (area interna = alcune porte) o su un computer utente (area interna = computer stesso)

Cosa *NON* è un firewall?

- Il firewall NON prende decisioni di routing (ovvero non decide le interfacce di uscita dei pacchetti da inoltrare)
  - Tuttavia può essere usato per marcare determinati dati. Tale marchio può essere tenuto in conto dal modulo che prende le decisioni di routing.
- Il firewall NON è un sistema di intrusion detection

Premesse

Gli esempi faranno riferimento a reti

- Ethernet
- IPv4

Questo è lo scenario più comune oggi negli ambienti domestici quanto nei grandi data center (almeno per quanto riguarda le reti best effort)

Premesse

Come si vede chiaramente dalla tabella sottostante gli indirizzi IPv4 stanno finendo e, probabilmente, nei prossimi anni assisteremo ad un emergere di IPv6. Tuttavia, per quanto riguarda gli utenti dei firewall, la differenza è risibile.

Date	Addresses free	Used up
2006-01-01	1468.61 M	
2007-01-01	1300.65 M	167.96 M
2008-01-01	1122.85 M	177.80 M (with return of 16.78 M to IANA)
2009-01-01	925.58 M	197.27 M

Identificazione

L'identificazione puo' avvenire

- Sui singoli pacchetti (firewall stateless)
- Su interi flussi (firewall stetefull)

iptables è un firewall statefull che gestisce sia regole basate su flussi che sui singoli pacchetti (come se fosse un firewall steteless).

Identificazione

I firewall possono identificare i pacchetti/flussi in base al

- Livello “Data Link” (MAC, livello 2 ISO OSI)
- Livello rete (IP, livello 3 ISO OSI)
- Livello trasporto (TCP/UDP/ICMP/. . . livello 4 ISO OSI)
- Recentemente a livello applicativo (HTTP/eD2K/Kazaa/Jabber/. . . livello 7 ISO OSI)

iptables supporta nativamente i primi tre punti e, tramite classificatori esterni (per esempio quelli forniti dai progetti L7-filter e IPP2P) il livello applicativo.

Target

Le regole a cui si sottopongono i pacchetti identificati prendono il nome di target, i principali sono

DROP	Elimina il pacchetto
ACCEPT	Accetta il pacchetto
REJECT	Restituisci al mittente del pacchetto un errore
(CON)MARK	Marca il pacchetto (o la connessione) in modo tale che sia riconoscibile da altri processi, per esempio quello di routing (iproute2) o il gestore della Qality of Service (QoS) (e.g. tc). Si tenga presente che il pacchetto, di per se, non viene modificato quindi questo marchio ha effetto solo all'interno dello stesso router/pc che l'ha marchiato
LOG	Segnala al syslogger il pacchetto
MASQUERADE/SNAT	Modifica l'header IP in modo da implementare un Network Address Translation (NAT)

Organizzazione

- Ogni riga di configurazione include una parte di identificazione e un target. Per esempio:
  - $\overbrace{\textit{Se il pacchetto utilizza il protocollo UDP}}^{\textit{Identificazione}}, \overbrace{\textit{eliminalo}}^{\textit{Target}}$
- Se il pacchetto corrisponde all'identificazione, viene applicato il target.
- Vi sono due tipi di target
  1. Terminali: la ricerca finisce e le righe successive sono ignorate.
  2. Non Terminali: il target viene eseguito e si continua con le righe successive
- In prima approssimazione l'unico target non terminale è LOG. Dunque è inutile mettere più di una riga pensata per lo stesso pacchetto: verrà eseguita solo la prima e le successive saranno ignorate

Esempio (1)

È corretta questa catena?

1. Se un pacchetto non è UDP, accettalo
2. Se un pacchetto è TCP, cancellalo

Risposta (1)

È corretta questa catena?

1. Se un pacchetto non è UDP, accettalo
2. Se un pacchetto è TCP, cancellalo

La catena di regole non è corretta, cerchiamo di capire cosa farebbe il firewall nel caso di un pacchetto UDP (che, leggendo la regola, sembrerebbe non dover essere accettato)

- Prima riga è ignorata (non rispetta la condizione pacchetto non UDP)
- Seconda regola è ignorata per lo stesso motivo

Mentre nel caso di un pacchetto TCP (che deve essere cancellato)

- Prima riga identifica il pacchetto quindi il pacchetto viene accettato
- Seconda riga non viene neanche valutata

Esempio (2)

Catena A

1. Se un pacchetto è UDP, accettalo
2. Se un pacchetto è UDP, “loggalo”

Catena B

1. Se un pacchetto è UDP, “loggalo”
2. Se un pacchetto è UDP, accettalo

Risposta (2)

Catena A

1. Se un pacchetto è UDP, accettalo
2. Se un pacchetto è UDP, “loggalo”

Non è corretta: i pacchetti UDP vengono accettati ma non vengono “loggati” (la prima riga “oscura” la seconda)

Catena B

1. Se un pacchetto è UDP, “loggalo”
2. Se un pacchetto è UDP, accettalo

La catena è corretta: la prima riga ha un target *non terminale* quindi si eseguirà sia il primo che il secondo target

Table

Ma le righe come sono organizzate? In base al tipo di regole vi sono varie tabelle (Table)

filter La tabella principale e di default: accetta target che NON modificano i pacchetti (e.g. ACCEPT, MARK, LOG)

nat Questa tabella viene consultata quando un pacchetto “inaugura” una nuova connessione e accetta target che modificano i pacchetti. Come il nome suggerisce, è particolarmente indicato per implementare una NAT

mangle Tabella pensata per la modifica dei pacchetti. Per esempio la si può utilizzare per incrementare il TTL dei pacchetti

raw Utilizzato principalmente per escludere alcuni pacchetti dall’analisi del firewall. Per esempio per evitare che alcuni pacchetti generino troppe informazioni di stato e facciano “scoppiare” la memoria

Table

- La tabella più utile è senz'altro la filter
- Se interessa il NAT, ovviamente serve anche la relativa tabella
- Si parlerà esclusivamente della tabella filter con qualche accenno alla tabella nat (sostanzialmente per implementare un NAT)

Chain

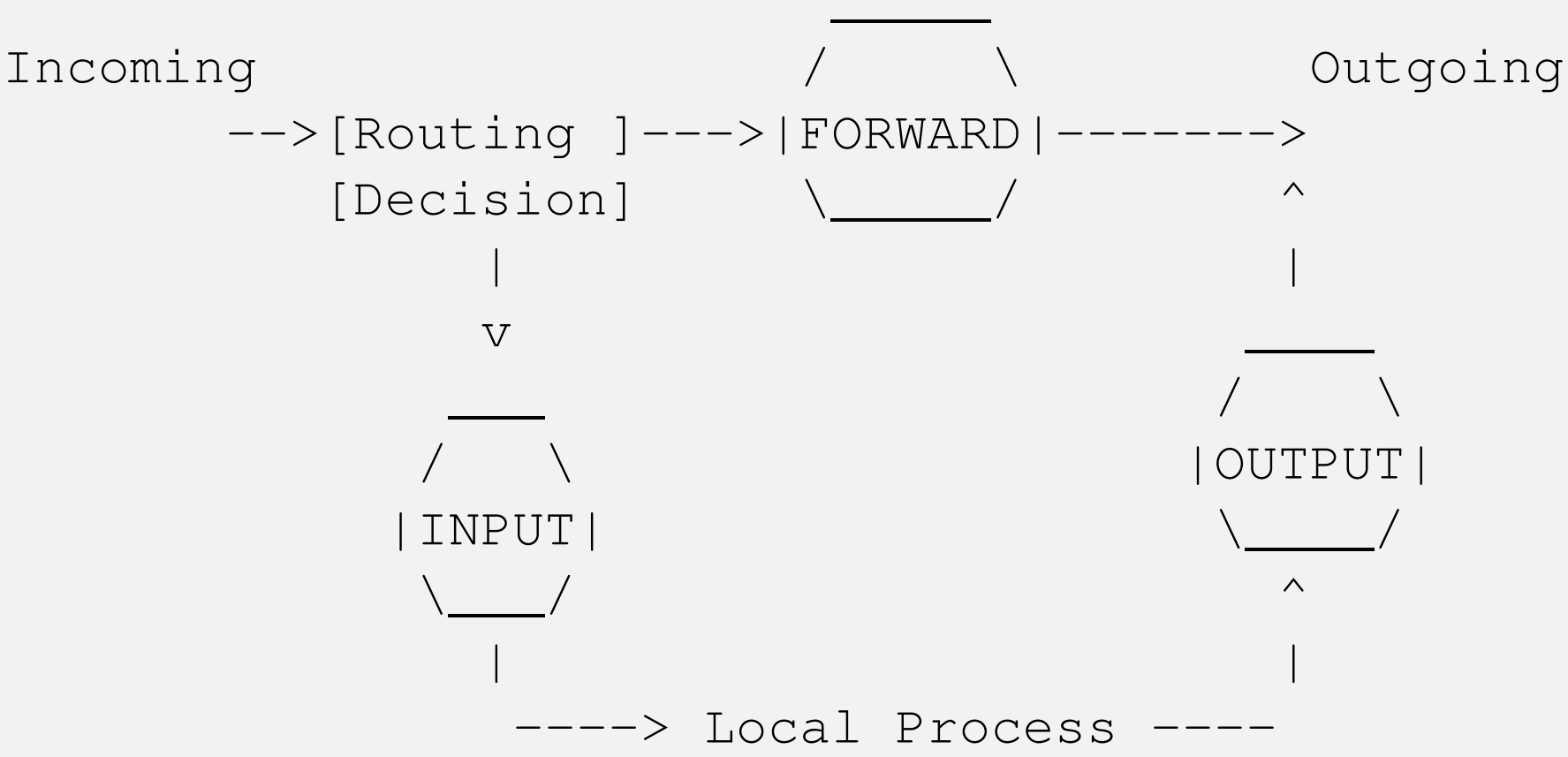
Le tabelle, a loro volta, sono divise in catene (chain) di linee di configurazione. Ogni tabella ha un determinato set di catene predefinite, per esempio la filter ha:

INPUT	Per i pacchetti destinati alla macchina su cui è presente il firewall
FORWARD	Per i pacchetti che sono inoltrati dalla macchina (router?) su cui gira il firewall
OUTPUT	Per i pacchetti che sono generati dalla macchina su cui è presente il firewall

Inoltre è possibile creare delle proprie catene. Il nome della catena personalizzata diventa un target.

Come sono richiamate le catene?

Come è chiaramente illustrato da [1] le catene della tabella filter sono richiamate come segue



iptables

- Il firewall di Linux ha, ovviamente, una componente (operativa) nel kernel.
- L'utente si deve interfacciare con tale componente per configurarlo.
- Per tale compito esiste il comando iptables
- Attenzione: iptables è solo un'interfaccia utente, se il kernel non supporta una caratteristica (e.g. un target non è disponibile nel kernel), anche se la pagina di manuale di iptables la nomina, ovviamente la caratteristica non funzionerà.

Tabella

L'opzione "-t" consente di scegliere la tabella su cui agire

Example

- iptables -t nat
- iptables
- iptables -t filter (equivalente alla riga precedente)
- iptables -t raw

Visualizzare le catene

L'opzione "-L" consente di visualizzare una catena (spesso abbinata con l'opzione "-v" che consente di visualizzare maggiori informazioni e "-n" per non fare reverse DNS di tutti gli IP)

Example

```
# iptables -vnL INPUT
Chain INPUT (policy DROP 1702K packets, 119M bytes)
 pkts bytes target     prot opt in     out     source               destination
 430K   65M ACCEPT     0    --  *      *       0.0.0.0/0            0.0.0.0/0  state RELATED,
                                                                                 ESTABLISHED
    0     0 ACCEPT     0    --  lo     *       0.0.0.0/0            0.0.0.0/0
   24  1520 ACCEPT     0    --  vlan92 *       0.0.0.0/0            0.0.0.0/0
    0     0 ACCEPT     tcp  --  eth1   *       0.0.0.0/0            0.0.0.0/0  tcp dpt:22
 267K   18M ACCEPT     udp  --  *      *       10.133.0.0/16        0.0.0.0/0  udp dpt:53
    0     0 ACCEPT     udp  --  *      *       10.133.0.0/16        0.0.0.0/0  udp dpt:123
16455 6068K ACCEPT     udp  --  !eth1  *       0.0.0.0/0            0.0.0.0/0  udp dpt:67
 1038 59179 ACCEPT     icmp --  !eth1  *       0.0.0.0/0            0.0.0.0/0  icmp type 8
```

Non specificando alcuna catena (e.g. iptables -vnL) si visualizzano tutte le catene della tabella

Policy

- Come si può osservare dalla slide precedente, vicino al nome della catena c'è una policy con un target (nell'esempio DROP)
- Serve a identificare un comportamento di "default" per tutti i pacchetti che non sono stati identificati in precedenza
- Equivale a dire "se il pacchetto non è stato identificato da nessuna linea della catena (oppure è stato identificato solo da linee con target non terminali) applica il target della policy"
- La policy è del tutto equivalente ad aggiungere una regola al fondo della catena che identifica TUTTI i pacchetti e che abbia lo stesso target della policy
- La policy si setta con l'opzione "-P" e può essere solo ACCEPT o DROP

Example

```
# iptables -P INPUT DROP
```

Aggiunta di una linea

- Si può utilizzare sia l'opzione "-A" che l'opzione "-I"
- "-A" aggiunge la riga in coda alla catena
- "-I" inserisce la riga nella posizione specificata della catena (Se non si mette alcun numero di posizione si intende in cima alla catena)

Example

```
# iptables -F INPUT
# iptables -A INPUT -p udp -j DROP
# iptables -A INPUT -p icmp -j DROP
# iptables -I INPUT 2 -p tcp -s 129.0.0.1 -j DROP
# iptables -vnL INPUT
Chain INPUT (policy ACCEPT 792K packets, 235M bytes)
 pkts bytes target     prot opt in     out     source               destination
   24  2932 DROP       udp  --  *      *       0.0.0.0/0            0.0.0.0/0
    0     0 DROP       tcp  --  *      *       129.0.0.1            0.0.0.0/0
    0     0 DROP       icmp --  *      *       0.0.0.0/0            0.0.0.0/0
```

Il significato delle opzioni "-p", "-s" e "-j" si vedrà successivamente (ma è facilmente immaginabile)

Rimozione di una o più linee

Si può utilizzare l'opzione "-D" per cancellare selettivamente una riga mentre "-F" cancella tutte le righe.

Example

```
# iptables -vnL INPUT
Chain INPUT (policy ACCEPT 792K packets, 235M bytes)
 pkts bytes target     prot opt in     out     source               destination
   24  2932 DROP       udp  --  *      *       0.0.0.0/0            0.0.0.0/0
    0     0 DROP       tcp  --  *      *       129.0.0.1            0.0.0.0/0
    0     0 DROP       icmp --  *      *       0.0.0.0/0            0.0.0.0/0
# iptables -D INPUT 1
# iptables -vnL INPUT
Chain INPUT (policy ACCEPT 792K packets, 235M bytes)
 pkts bytes target     prot opt in     out     source               destination
    0     0 DROP       tcp  --  *      *       129.0.0.1            0.0.0.0/0
    0     0 DROP       icmp --  *      *       0.0.0.0/0            0.0.0.0/0
# iptables -F INPUT
# iptables -vnL INPUT
Chain INPUT (policy ACCEPT 793K packets, 235M bytes)
 pkts bytes target     prot opt in     out     source               destination
```

Rimozione di una o più linee

Attenzione! l'opzione "-F" cancella tutte le linee ma NON setta la policy su ACCEPT, quindi se avete settato come policy DROP molto probabilmente bloccherete TUTTE le connessioni. Compresa la vostra sessione SSH con cui state configurando il firewall, chiudendovi allegramente FUORI (questo può essere un problema non trascurabile, specie nei sistemi embedded)

Definizione dei target

Il target viene indicato con l'opzione "-j"

Example

- # iptables -I INPUT -j LOG # logga tutto il traffico in ingresso (è la prima regola e non ha alcuna parte di identificazione)
- # iptables -A INPUT -j DROP # simile a iptables -P INPUT DROP
- # iptables -A INPUT # Identifica tutto il traffico e non fare nulla (ovviamente non è terminale)

Definizione delle caratteristiche da identificare

Alcune opzioni comunemente utilizzate per identificare il traffico sono:

- L'opzione "-i" consente di specificare l'interfaccia da cui si è ricevuto il pacchetto (e.g. eth0, lo). Ovviamente non è valido in catene come la OUTPUT
- L'opzione "-o" è la duale della "-i" per l'interfaccia di uscita. Ovviamente non è valido in catene come la INPUT
- L'opzione "-s" consente di specificare l'indirizzo o la rete (classless) della sorgente dei pacchetti (e.g. 130.192.27.216, 130.192.0.0/16).
- L'opzione "-d" è la duale di "-s" per l'indirizzo/rete della destinazione
- L'opzione "-p" consente di specificare il protocollo (e.g. UDP/TCP/ICMP)
- Caricando il modulo mac (con il comando "-m mac") è possibile specificare l'indirizzo MAC sorgente con l'opzione "–mac-source"

Esempio

Example

```
# iptables -I FORWARD -m mac -i eth0 -o eth1 -s 130.192.27.216 -d
130.192.3.24 -p udp –mac-source 00:11:22:33:44:55 -j DROP
```

Ovvero per i paccetti

- Che si stiano inoltrando dall'interfaccia eth0
- All'interfaccia eth1
- Il cui indirizzo sorgente sia 130.192.27.216
- Il cui indirizzo destinazione sia 130.192.3.24
- Che contengano un datagram UDP
- E il cui indirizzo mac sorgente del precedente router (o della sorgente) sia uguale a 00:11:22:33:44:55
- Devono essere cancellati

Livello Trasporto

Se il protocollo è UDP o TCP spesso si utilizzano le porte sorgente e destinazione per distinguere il protocollo di livello applicativo

- I server HTTP tipicamente sono in ascolto sulla porta 80/tcp
- I server DNS tipicamente sono in ascolto sulla porta 53/udp
- Questa è un'assunzione non sempre vera
  - Posso avviare un server HTTP su una qualunque porta TCP
  - Freenet, skype, kad, il trasferimento dati di quasi tutti i protocolli P2P che porte usano?
  - Skype usa (tra l'altro) la porta 443/tcp (assegnata all'HTTPS) per instaurare telefonate
- Per un approccio base può andare bene lo stesso
- Si utilizzano le opzioni "–sport" e "–dport" (disponibili sono con "-p udp" e "-p tcp")

Esempio

Example

```
# iptables -A OUTPUT -p udp –dport 53 -j ACCEPT
# iptables -A OUTPUT -p tcp –dport 80:443 -j ACCEPT
# iptables -P OUTPUT DROP
```

Consenti connessioni generate dalla macchina locale verso le porte 53/udp (DNS?) 80/tcp (HTTP?) e 443/tcp (HTTPS?).

Esempio

Ma se la mia macchina non è un computer utente ma un router?

- Voglio consentire agli utenti della mia rete (eth0) di accedere ad internet (con l'interfaccia eth1) SOLO sulle porte 53/udp (DNS?) 80/tcp (HTTP?) e 443/tcp (HTTPS?)
- Non voglio che dall'esterno si possano aprire connessioni verso le macchine della mia rete
- Potrei riproporre lo stesso meccanismo di prima? (con le opportune modifiche)

Funziona?

```
# iptables -A FORWARD -i eth0 -o eth1 -p udp –dport 53 -j ACCEPT
# iptables -A FORWARD -i eth0 -o eth1 -p tcp –dport 80:443 -j ACCEPT
# iptables -P FORWARD DROP
```

Esempio

Funziona?

```
# iptables -A FORWARD -i eth0 -o eth1 -p udp –dport 53 -j ACCEPT
# iptables -A FORWARD -i eth0 -o eth1 -p tcp –dport 80:443 -j ACCEPT
# iptables -P FORWARD DROP
```

- Non fa ciò che voglio: consente ai pacchetti di uscire ma non consente alle risposte di tornare al richiedente.
- Potrei aggiungere regole per risolvere questo problema:

Ho risolto in questo modo?

```
# iptables -A FORWARD -i eth0 -o eth1 -p udp –dport 53 -j ACCEPT
# iptables -A FORWARD -i eth0 -o eth1 -p tcp –dport 80:443 -j ACCEPT
# iptables -A FORWARD -i eth1 -o eth0 -p udp –sport 53 -j ACCEPT
# iptables -A FORWARD -i eth1 -o eth0 -p tcp –sport 80:443 -j ACCEPT
# iptables -P FORWARD DROP
```

Esempio

Ho risolto in questo modo?

```
# iptables -A FORWARD -i eth0 -o eth1 -p udp –dport 53 -j ACCEPT
# iptables -A FORWARD -i eth0 -o eth1 -p tcp –dport 80:443 -j ACCEPT
# iptables -A FORWARD -i eth1 -o eth0 -p udp –sport 53 -j ACCEPT
# iptables -A FORWARD -i eth1 -o eth0 -p tcp –sport 80:443 -j ACCEPT
# iptables -P FORWARD DROP
```

La comunicazione funziona ma

- Una persona che si trova all'esterno e vuole collegarsi ad una macchina interna può farlo!!!
- Basta che i pacchetti che manda abbiano come porta sorgente una tra 80/tcp, 443/tcp e 53/udp
- Non è rispettato uno dei requisiti del sistema!
- Bisogna sfruttare le potenzialità del firewall statefull

Esempio

Soluzione

```
# iptables -A FORWARD -m state -i eth1 -o eth0 –state
RELATED,ESTABLISHED -j ACCEPT
# iptables -A FORWARD -i eth0 -o eth1 -p udp –dport 53 -j ACCEPT
# iptables -A FORWARD -i eth0 -o eth1 -p tcp –dport 80:443 -j ACCEPT
# iptables -P FORWARD DROP
```

- La prima riga accetta tutti i pacchetti il cui stato sia
  - “ESTABLISHED” (ovvero non sono i primi pacchetti della connessione) oppure
  - “RELATED” (ovvero connessione legata alla precedente, a rigore non servirebbe)
- Questo consente la ricezione delle risposte dall’esterno
- Non consente che dall’esterno si aprano connessioni verso l’interno

Introduzione

Assumendo di avere a disposizione meno indirizzi IP di quanto si necessita cosa si può fare?

- Si assegna uno o più indirizzi IP forniti dall'ISP ad un'interfaccia del router (assumiamo eth1)
- Si assegnano degli IP privati alle altre interfacce (assumiamo una sola interfaccia, eth0)
- Si abilita il forwarding in /proc/sys/net/ipv4/ip_forward
- Si configura il firewall in modo da consentire il forwarding dei pacchetti da eth0 a eth1 (e i pacchetti di risposta)
- Si configura il firewall in modo da modificare l'header IP e fare da NAT

Esempio funzionante

Example

Abilito il Forwarding
iptables -A FORWARD -i eth0 -o eth1 -j ACCEPT
iptables -A FORWARD -m state -i eth1 -o eth0 –state RELATED,ESTABLISHED -j ACCEPT
Attivo il NAT
iptables -t nat -A POSTROUTING -o eth1 -j MASQUERADE

Esempio funzionante

- Molto semplice ma con basse prestazioni
- Se si conosce l'indirizzo o gli indirizzi a disposizione è molto più performante sostituire l'ultima riga. (Si assuma di avere a disposizione gli indirizzi 130.192.225.5-130.192.225.10)

Example

```
# iptables -t nat -D POSTROUTING -o eth1 -j MASQUERADE
# iptables -t nat -A POSTROUTING -o eth1 -j SNAT –to-source
130.192.225.5-130.192.225.10
# # Se avessi avuto a disposzione solo un indirizzo avrei scritto –to-source
130.192.225.5
```

Considerazioni varie

- iptables <opzioni> –help è vostro amico ☺.
- Può sembrare una banalità ma tutti i moduli (e.g. mac, state) hanno loro opzioni e anche solo indicare “-p icmp” attiva delle opzioni specifiche, il comando –help mostra tutte le opzioni disponibili con i moduli e i flag specificati fino a quel punto
- Se volete fare da router ricordate di abilitare l’ip forwarding in /proc/sys/net/ipv4/ip_forward!!!
- Ovviamente vi sono altri aspetti da considerare quando si configura un router (per esempio server DHCP o algoritmi di routing)
- Spesso è utile identificare i pacchetti in base all’utente (locale) che li ha generati, per questo scopo c’è il modulo owner
- Quasi tutti i campi si possono negare con il carattere “!”, per esempio se voglio selezionare tutti i pacchetti che non provengono dall’host 130.192.225.79 posso indicare “-i ! 130.192.225.79”

Valutazione difficoltà della lezione

?

Bibliografia I


Rusty Russel.
Linux 2.4 packet filtering howto.
Technical report, Netfilter, 2002.

Copyright

Copyright

Tu sei libero:

di riprodurre, distribuire, comunicare al pubblico, esporre in pubblico, rappresentare, eseguire e recitare quest'opera

Alle seguenti condizioni:

Attribuzione. Devi attribuire la paternità dell'opera nei modi indicati dall'autore o da chi ti ha dato l'opera in licenza e in modo tale da non suggerire che essi avallino te o il modo in cui tu usi l'opera.

Non commerciale. Non puoi usare quest'opera per fini commerciali.

Non opere derivate. Non puoi alterare o trasformare quest'opera, ne' usarla per crearne un'altra.

Copyright

- Ogni volta che usi o distribuisci quest'opera, devi farlo secondo i termini di questa licenza, che va comunicata con chiarezza.
- In ogni caso, puoi concordare col titolare dei diritti utilizzi di quest'opera non consentiti da questa licenza.
- Questa licenza lascia impregiudicati i diritti morali.

Le utilizzazioni consentite dalla legge sul diritto d'autore e gli altri diritti non sono in alcun modo limitati da quanto sopra.
Questo è un riassunto in linguaggio accessibile a tutti del Codice Legale (la licenza integrale) disponibile all'indirizzo:

`http://creativecommons.org/licenses/by-nc-nd/2.5/it/legalcode`