Sicurezza dei Sistemi in Rete: seconda esercitazione firewall

Indice

Avvio e verifica della rete virtuale

L'esercitazione consiste nell'impostare regole di natting e firewalling in una rete composta da otto macchine virtuali. Lo schema della rete è rappresentato nell'immagine network_diagram.png, riportato anche nell'ultima pagina.

1. Visualizzare il diagramma di rete

   $ ristretto network_diagram.png

   il comando ristretto avvia un semplice programma di visualizzazione di immagini incluso di default nell'installazione di xfce.

La rete utilizzata per l'esercitazione è suddivisa in due sottoreti, DMZ e LAN. La rete LAN contiene la macchina locale *local1* e la macchina *dhcp* su cui è in esecuzione il server DHCP a servizio della LAN e alla DMZ. La rete DMZ contiene due server, un server HTTP (*www*) e un server FTP (*ftp*). Entrambi i server sono configurati con un indirizzo IP privato. Il firewall installato sulla macchina *int-firewall* separa la LAN dalla DMZ, mentre la DMZ è separata da Internet dal firewall *ext-firewall*. L'interfaccia di rete eth1 del firewall *ext-firewall* è l'unica interfaccia appartenente alla rete da proteggere configurata con un indirizzo IP pubblico, a cui corrispondono i nomi *www.fake.com* e *ftp.fake.com*.

Le macchine *www.google.com* e *remote1* rappresentano un server web e una macchina remota connesse ad Internet e dotate di IP pubblici. In questa semplice rete emulata, Internet è rappresentata da un singolo router, implementato dalla macchina *extrouter*.

Tutte le macchine virtuali eseguono il boot in parallelo, litigandosi la capacità computazionale disponibile. Il processo di avvio impiegherà qualche minuto. La rete è pronta per essere utilizzata quando tutte le macchine virtuali hanno completato il boot. Per ovvi motivi, l'esecuzione di iptables richiede i diritti di root, quindi l'account di root sarà quello più utilizzato. Verificare la connettività della rete:

remote1:~# ping www.google.com

per verificare la connettività della rete useremo il comando ping. Ping consente di inviare verso una macchina remota un pacchetto ICMP di tipo *echo-request*. Se non ci sono filtri di pacchetto attivi o configurazioni particolari, una macchina che riceve un *echo-request* risponde

inviando un *echo-reply* al mittente. Quando la macchina mittente riceve l'*echo-reply*, il comando ping stampa una riga sullo standard output. L'output prodotto da *ping* implica la capacità delle due macchine di scambiarsi pacchetti in entrambe le direzioni.

Dal nodo *remote1* provate a pingare www.google.com, www.fake.com, ftp.mail.com, *local1, ftp e www*.

Dal nodo *local1* provate a pingare www.google.com, www.fake.com, ftp.fake.com,*www, ftp e remote1*.

Notate che nodi con indirizzi privati non sono direttamente raggiungibili dall'esterno della rete protetta.

2. Verificare i servizi della rete

remote1:~# w3m www.google.com

local11:~# w3m www

w3m è un web browser con interfaccia testuale. Il comando *w3m nome_sito* visualizza la home page del sito web *nome_sito*. Per uscire da *w3m*, premere *q*, seguito da *y*. Se vengono visualizzate le homepage di *www.google.com* e di *www.fake.com*, significa che le macchine sono raggiungibili e che i loro web server (implementati con Apache2) sono avviati e configurati correttamente.

local1:~# ftp ftp

ftp è un client FTP con interfaccia testuale. Il comando *ftp nome_server* apre una sessione FTP verso il server *nome_server*. Alla richiesta di credenziali di accesso, accedere come l'utente alice (*username:alice password:alice*).

ftp> dir

usate il comando *dir* per ottenere la lista dei file dell'utente alice.

ftp> get file2

usate il comando *get nome_file* per scaricare dal server FTP il file nome_file.

Potete chiudere il comando *ftp* premendo la combinazione di tasti *Ctrl-d*

local1:~# ssh www

tutte le macchine della rete hanno server e client SSH. Potete aprire una connessione remota usando il comando *ssh nome_host*. Per terminare la connessione, utilizzare la combinazione di tasti *Ctrl-d*.

Politiche di filtraggio e di nat

La rete da proteggere appartiene all'azienda FAKE. Tale azienda ha due server pubblicamente accessibili: un server web (che dovrà essere raggiungibile tramite il nome www.fake.com) e un server ftp (che dovrà essere raggiungibile tramite il nome ftp.fake.com). Entrambi i server sono dotati di indirizzi IP privati e sono connessi alla DMZ. Il server web dispone anche di un web proxy, in ascolto sulla porta 8080. Fake ha inoltre una rete locale (LAN) con due macchine (*local1* e *dhcp*) dotata di indirizzo IP privato. Due firewall (*int-firewall e ext-firewall*) separano la LAN dalla DMZ e la DMZ da Internet. A causa della crisi economica, FAKE ha deciso che un solo indirizzo IP pubblico è più che sufficiente. Tale indirizzo è assegnato all'interfaccia di rete di *ext-firewall* direttamente connessa ad Internet.

Int-firewall e *ext-firewall* applicano le regole di filtraggio per limitare il traffico di rete tra LAN e DMZ e tra DMZ e Internet. Entrambi i firewall applicano anche regole di nat. *Ext-firewall* deve consentire agli host in internet di accedere al server Web e al server FTP utilizzando gli indirizzi *www.fake.com* e *ftp.fake.com*, che si risolvono entrambi nell'unico indirizzo IP pubblico disponibile.

La macchina *local1* è usata dall'amministratore di rete, ed è l'unica macchina che deve essere in grado di accedere da remoto (tramite **ssh**) a *int-firewall* e a *ext-firewall*.

Politiche di filtraggio dei pacchetti

- Utilizzare una policy di negazione implicita per tutti i pacchetti in transito su entrambi i firewall
- Utilizzare una policy di negazione implicita per tutti i pacchetti in ingresso in entrambi i firewall
- Utilizzare una policy di negazione implicita per tutti i pacchetti in uscita da entrambi i firewall
- Consentire flussi di comunicazione UDP provenienti dalla DMZ e diretti al server DHCP relay in esecuzione sul *int-firewall.*
- Consentire flussi di comunicazione UDP provenienti dal server DHCP relay in esecuzione su *int-firewall* verso la DMZ.
- Consentire flussi di comunicazione UDP provenienti da *int-firewall* e diretti al server DHCP in esecuzione su *dhcp.*
- Consentire flussi di comunicazione UDP provenienti dal *dhcp* e diretti verso il server DHCP relay in esecuzione su *int-firewall.*
- Consentire flussi di comunicazione UDP provenienti dalla DMZ verso la rete LAN relative al traffico DNS (porta 53).
- Consentire le risposte delle richieste DNS generate dalla DMZ.
- Consentire a *local1* di aprire connessioni ssh verso *int-firewall* ed *ext-firewall*
- Consentire le risposte di *int-firewall* ed *ext-firewall* alle connessioni ssh generate da *local1*
- Consentire connessioni TCP sulla porta 80 dalla LAN verso il server web *www*
- Consentire le risposte del server web *www* a connessioni originate dalla LAN
- Consentire connessioni TCP sulla porta 21 dalla LAN verso il server FTP *ftp.fake.com*
- Consentire le risposte di *ftp.fake.com* a connessioni originate dalla LAN
- Consentire connessioni TCP sulla porta 80 da Internet verso il server web

- Consentire le risposte del server web a connessioni originate da Internet
- Consentire connessioni TCP sulla porta 21 da Internet verso il server FTP *ftp.fake.com*
- Consentire le risposte di *ftp.fake.com* a connessioni originate da Internet

Politiche di Network Address Translation

- Consentire agli host in Internet di accedere al sito web installato nella DMZ utilizzando l'indirizzo *www.fake.com*
- Consentire agli host in Internet di accedere al server FTP installato nella DMZ utilizzando l'indirizzo *ftp.fake.com*

Politiche di filtraggio dei protocolli

- Consentire agli host della LAN di accedere a server web in Internet solo utilizzando il server web installato nella DMZ di FAKE come proxy (non trasparente)

Implementazione delle politiche di filtraggio e di nat

Prima di iniziare

Segue un piccolo elenco di comandi fondamentali. Questi comandi NON fanno parte della soluzione dell'esercitazione, servono solo per ricordare alcuni comandi utili e la loro sintassi.

- *firewall:~# man iptables*

  l'unico comando di cui avete veramente bisogno. Visualizza la pagina di manuale del comando iptables. Per uscire dalla pagina di manuale, premere *q*

- *firewall:~# iptables -t filter -L -v -n*

  visualizza le regole attualmente incluse nelle catene appartenenti alla tabella filter e le loro policy di default. L'opzione *-n* evita che iptables provi ad eseguire il reverse lookup degli indirizzi IP

- *firewall:~# iptables -t filter -P FORWARD DROP*

  imposta la policy di negazione implicita (*DROP*) sulla catena *FORWARD* della tabella *filter*

- *firewall:~# iptables -t filter -A FORWARD -p tcp --dport 22 -i eth0 -j DROP*

  esempio di comando utilizzato per aggiungere una regola di packet filtering statico. Questo comando aggiunge una regola alla catena *FORWARD* della tabella filter. La regola inserita blocca (*-j DROP*) tutti i pacchetti *TCP* aventi *22* come numero di porta di destinazione e che hanno eth0 come interfaccia di ingresso

- *firewall:~# iptables -t filter -D FORWARD 2*

  esempio di eliminazione selettiva di una singola regola. Questo comando elimina la seconda regola della catena *FORWARD* nella tabella filter

- *firewall:~# iptables -t filter -F FORWARD*

  eliminazione di tutte le regole appartenenti alla catena *FORWARD*. Questo comando non modifica la policy di default della catena.

- *firewall:~# iptables -t filter -F*

  eliminazione di tutte le regole appartenenti a tutte le catene della tabella filter. Le

policy di default delle catene non vengono modificate

Dopo l'inserimento di una nuova regola di filtraggio o di nat, verificate che la regola sia stata effettivamente aggiunta nella tabella e nella catena corrette utilizzando l'opzione -L. Verificatene inoltre gli effetti sulla raggiungibilità delle macchine e sulla fruibilità dei loro servizi.

Nel corso dell'esercitazione sarà necessario modificare dei file di testo. In tutte le macchine virtuali è installato l'editor di testo *vim* (che, in caso qualcuno avesse dubbi, è meglio di emacs...). Per aprire un file di testo da modificare potete utilizzare il comando

- *nome-macchina:~#vim nome-file*

Dopo aver aperto il file, premere *i* per entrare in modalità di inserimento di testo, ed effettuare le modifiche. Al termine delle modifiche premere *ESC* per uscire dalla modalità di inserimento testo, e la sequenza di caratteri *:wq* per salvare le modifiche e chiudere il file.

In seguito è proposta una possibile implementazione delle policy di filtraggio e di nat. La soluzione proposta non è l'unica implementazione possibile.

Implementazione delle politiche di NAT

Policy:

Consentire agli host in Internet di accedere al sito web installato nella DMZ utilizzando l'indirizzo *www.fake.com*

Prima di implementare la policy, verificare che è impossibile per *remote1* accedere al server web nella DMZ di fake

remote1:~# w3m www.fake.com

Implementazione:

ext-firewall:~# iptables -t nat -A PREROUTING -p tcp -i eth1 --dport 80 -j DNAT --to-destination 192.168.1.1

Verificare che www.fake.com è diventato accessibile da *remote1*

Policy:

Consentire agli host in Internet di accedere al server FTP installato nella DMZ utilizzando l'indirizzo ftp.fake.com

Prima di implementare la policy, verificare che è impossibile per remote1 accedere al server FTP nella DMZ di fake

remote1:~# ftp ftp.fake.com

Implementazione:
ext-firewall:~# iptables -t nat -A PREROUTING -p tcp -i eth1 --dport ftp -j DNAT --to-destination 192.168.1.2

Verificare che ftp.fake.com è diventato accessibile da remote1, e che è possibile scaricare file

Implementazione delle politiche di filtraggio

Policy:

Utilizzare una policy di negazione implicita per tutti i pacchetti in transito su entrambi i firewall

Prima di implementare la policy, verificare che è possibile per *local1* accedere al server web e al server FTP nella DMZ di FAKE

local1:~# w3m www

local1:~# ftp ftp

Implementazione:

int-firewall:~# iptables -t filter -P FORWARD DROP

ext-firewall:~# iptables -t filter -P FORWARD DROP

Verificare l'impossibilità di comunicare tra le macchine nella DMZ e *local1*

Verificare l'impossibilità di comunicare tra le macchine nella DMZ e Internet

Verificare che è ancora possibile aprire connessioni ssh verso i firewall

Policy:

Utilizzare una policy di negazione implicita per tutti i pacchetti in ingresso su entrambi i firewall

Utilizzare una policy di negazione implicita per tutti i pacchetti in uscita su entrambi i firewall

Implementazione:

int-firewall:~# iptables -t filter -P INPUT DROP

int-firewall:~# iptables -t filter -P OUTPUT DROP

ext-firewall:~# iptables -t filter -P INPUT DROP

ext-firewall:~# iptables -t filter -P OUTPUT DROP

Verificare che non è più possibile aprire connessioni ssh verso i firewall

Policy:

Consentire flussi di comunicazione UDP provenienti dalla DMZ e diretti al server DHCP relay in esecuzione sul *int-firewall.*

Consentire flussi di comunicazione UDP provenienti dal server DHCP relay in esecuzione su *int-firewall* verso la DMZ.

Consentire flussi di comunicazione UDP provenienti da *int-firewall* e diretti al server DHCP in esecuzione su *dhcp.*

Consentire flussi di comunicazione UDP provenienti dal *dhcp* e diretti verso il server DHCP relay in esecuzione su *int-firewall.*

Implementazione:

int-firewall:~# iptables -t filter -A INPUT -i eth1 -p udp --sport 68 --dport 67 -j ACCEPT

int-firewall:~# iptables -t filter -A OUTPUT -o eth1 -p udp --sport 67 --dport 68 -j ACCEPT

int-firewall:~# iptables -t filter -A OUTPUT -o eth0 -p udp -s 192.168.1.253 --sport 67 -d 192.168.2.253 --dport 67 -m state --state NEW,ESTABLISHED -j ACCEPT
int-firewall:~# iptables -t filter -A INPUT -i eth0 -p udp -s 192.168.2.253 --sport 67 -d 192.168.1.253 --dport 67 -m state --state ESTABLISHED -j ACCEPT

Verificare che ora è possibile effettuare richieste DHCP attraverso il DHCP relay in esecuzione su *int-firewall.*

Policy:

Consentire flussi di comunicazione UDP provenienti dalla DMZ verso la rete LAN relative al traffico DNS (porta 53).

Consentire le risposte delle richieste DNS generate dalla DMZ.

Implementazione:

int-firewall:~# iptables -A FORWARD -p udp --dport 53 -i eth1 -o eth0 -s 192.168.1.0/24 -d 192.168.2.253 -m state --state NEW,ESTABLISHED -j ACCEPT

int-firewall:~# iptables -A FORWARD -p udp --sport 53 -o eth1 -i eth0 -d 192.168.1.0/24 -s 192.168.2.253 -m state --state ESTABLISHED -j ACCEPT

Verificare se è possibile effettuare richieste di risoluzione DNS dagli host presenti nella DMZ al server DNS in esecuzione su *dhcp* (es: *nslookup local1*).

Policy:

Consentire a *local1* di aprire connessioni ssh verso *int-firewall* ed *ext-firewall*

Consentire le risposte di *int-firewall* ed *ext-firewall* alle connessioni ssh generate da *local1*

Implementazione:

Iniziamo con il consentire le connessioni ssh da *local1* a *int-firewall*

int-firewall:~# iptables -t filter -A INPUT -p tcp --dport ssh -s 192.168.2.1 -m state --state NEW,ESTABLISHED -j ACCEPT

int-firewall:~# iptables -t filter -A OUTPUT -p tcp --sport ssh -d 192.168.2.1 -m state --state ESTABLISHED -j ACCEPT

Verificare che è ora possibile aprire una connessione ssh da *local1* a 192.168.2.254

Ora introduciamo le regole necessarie per consentire connessioni ssh da *local1* a *ext-firewall*

int-firewall:~# iptables -t filter -A FORWARD -p tcp --dport ssh -s 192.168.2.1 -d 192.168.1.254 -m state --state NEW,ESTABLISHED -j ACCEPT

int-firewall:~# iptables -t filter -A FORWARD -p tcp --sport ssh -d 192.168.2.1 -s 192.168.1.254 -m state --state ESTABLISHED -j ACCEPT

ext-firewall:~# iptables -t filter -A OUTPUT -p tcp --sport ssh -d 192.168.2.1 -m state --state ESTABLISHED -j ACCEPT

ext-firewall:~# iptables -t filter -A INPUT -p tcp --dport ssh -s 192.168.2.1 -m state --state NEW,ESTABLISHED -j ACCEPT

Verificare che è ora possibile aprire una connessione ssh da local1 a 192.168.1.254

Policy:

Consentire connessioni TCP sulla porta 80 dalla LAN verso il server web *www*

Consentire le risposte del server web *www* a connessioni originate dalla LAN

Implementazione:

int-firewall:~# iptables -t filter -A FORWARD -s 192.168.2.0/24 -d 192.168.1.1 -p tcp --dport 80 -m state --state NEW,ESTABLISHED -j ACCEPT

int-firewall:~# iptables -t filter -A FORWARD -d 192.168.2.0/24 -s 192.168.1.1 -p tcp --sport 80 -m state --state ESTABLISHED -j ACCEPT

Verificare che è ora possibile per local1 accedere al server web nella DMZ di FAKE

Policy:

Consentire connessioni TCP sulla porta 21 dalla LAN verso il server FTP ftp

Consentire le risposte di ftp a connessioni originate dalla LAN

Implementazione:

int-firewall:~# iptables -t filter -A FORWARD -s 192.168.2.0/24 -d 192.168.1.2 -p tcp --dport ftp -m state --state NEW,ESTABLISHED -j ACCEPT

int-firewall:~# iptables -t filter -A FORWARD -d 192.168.2.0/24 -s 192.168.1.2 -p tcp --sport ftp -m state --state ESTABLISHED -j ACCEPT

int-firewall:~# iptables -t filter -A FORWARD -d 192.168.2.0/24 -s 192.168.1.2 -p tcp --sport ftp-data -m state --state RELATED,ESTABLISHED -j ACCEPT

int-firewall:~# iptables -t filter -A FORWARD -s 192.168.2.0/24 -d 192.168.1.2 -p tcp --dport ftp-data -m state --state ESTABLISHED -j ACCEPT

Verificare che è ora possibile per local1 accedere al server ftp nella DMZ di FAKE

Policy:

Consentire connessioni TCP sulla porta 80 da Internet verso il server web

Consentire le risposte del server web a connessioni originate da Internet

Implementazione:

ext-firewall:~# iptables -t filter -A FORWARD -i eth1 -p tcp --dport www -d 192.168.1.1 -m state --

state NEW,ESTABLISHED -j ACCEPT

ext-firewall:~# iptables -t filter -A FORWARD -o eth1 -p tcp --sport www -s 192.168.1.1 -m state --state ESTABLISHED -j ACCEPT

Verificare che è possibile per *remote1* accedere a *www.fake.com*

Policy:

Consentire connessioni TCP sulla porta 21 da Internet verso il server FTP *ftp.fake.com*

Consentire le risposte di ftp.fake.com a connessioni originate da Internet

Implementazione:

ext-firewall:~# iptables -t filter -A FORWARD -i eth1 -p tcp --dport ftp -d 192.168.1.2 -m state --state NEW,ESTABLISHED -j ACCEPT

ext-firewall:~# iptables -t filter -A FORWARD -o eth1 -p tcp --sport ftp -s 192.168.1.2 -m state --state ESTABLISHED -j ACCEPT

ext-firewall:~# iptables -t filter -A FORWARD -o eth1 -p tcp --sport ftp-data -s 192.168.1.2 -m state --state RELATED,ESTABLISHED -j ACCEPT

ext-firewall:~# iptables -t filter -A FORWARD -i eth1 -p tcp --dport ftp-data -d 192.168.1.2 -m state --state ESTABLISHED -j ACCEPT

Verificare che è ora possibile per remote1 accedere al server *ftp.fake.com*

Implementazione delle politiche di filtraggio dei protocolli

Policy:

Consentire agli host della LAN di accedere a server web in Internet solo utilizzando il server web installato nella DMZ di FAKE come proxy (non trasparente).

Allo stato attuale, la rete LAN è completamente isolata da Internet. Questo garantisce ottimi livelli di sicurezza, a scapito di una scarsa usabilità (per *local1* è possibile accedere solo ai servizi erogati dalle macchine nella DMZ). Per consentire alle macchine della LAN di accedere a siti web in Internet senza rendere necessario un contatto diretto tra macchine della LAN e Internet è possibile configurare i client in modo da usare il server *www* installato nella DMZ di FAKE come proxy. Questo significa che tutte le connessioni verso un server web aperte da un client nella LAN verranno inoltrate verso la porta 8080 di *www* (dove è in ascolto il software *apache2 (mod_proxy)*). *Apache2 mod_proxy* aprirà le connessioni HTTP a server web in Internet per conto dei client nella LAN, mediando (e loggando, ed eventualmente filtrando) tutte le comunicazioni. Come conseguenza, è inoltre necessario:

- configurare *ext-firewall* in modo da consentire a *www* di aprire connessioni verso server web in Internet, e di ricevere le relative risposte
- configurare *ext-firewall* in modo da effettuare SNAT (in quanto *www* ha un indirizzo IP privato)
- configurare *int-firewall* in modo da consentire le connessioni da host nella LAN alla porta 8080 di www, e le relative risposte
- configurare *local1* in modo da usare *www* come proxy
- configurare tinyproxy in modo da accettare le connessioni provenienti dalle macchine della LAN

Implementazione:

ext-firewall:~# iptables -t nat -A POSTROUTING -p tcp --dport 80 -s 192.168.1.1 -o eth1 -j MASQUERADE

ext-firewall:~# iptables -A FORWARD -p tcp -i eth0 -s 192.168.1.1 --dport 80 -o eth1 -m state --state NEW,ESTABLISHED -j ACCEPT

ext-firewall:~# iptables -A FORWARD -p tcp -o eth0 -d 192.168.1.1 --sport 80 -i eth1 -m state --state ESTABLISHED -j ACCEPT

A questo punto *www.fake.com* deve essere in grado di raggiungere *www.google.com*

int-firewall:~# iptables -t filter -A FORWARD -i eth0 -o eth1 -s 192.168.2.0/24 -d 192.168.1.1 -p tcp --dport 8080 -m state --state NEW,ESTABLISHED -j ACCEPT

int-firewall:~# iptables -t filter -A FORWARD -o eth0 -i eth1 -d 192.168.2.0/24 -s 192.168.1.1 -p tcp --sport 8080 -m state --state ESTABLISHED -j ACCEPT

local1:~# export HTTP_PROXY=http://192.168.1.1:8080/

A questo punto, le richieste di *local1* vengono dirottate verso il transparent proxy. Provate ad eseguire il comando:

local1:~# w3m www.google.com

Per configurare Apache2 in modo tale che agisca come proxy ed essere grado di servire le richieste provenienti dalla LAN, in particolare da *local1*, occorre modificare nell' host *www* i seguenti files:

- */etc/apache2/mods-enabled/proxy.conf*
- */etc/apache2/sites-available/forward_proxy.conf*
- */etc/apche2/ports.conf*

Per ulteriori dettagli relativi alla precisa configurazione è possibile consultare link.

Per concludere

Verificate la lista di regole di filtraggio con i comandi

int-firewall:~# iptables -t filter -L -v -n

ext-firewall:~# iptables -t filter -L -v -n

Analizzate la lista e ricostruite i comandi di iptables necessari per creare le regole di filtraggio

Verificate la lista di regole di nat con i comandi

int-firewall:~# iptables -t nat -L -v -n

ext-firewall:~# iptables -t nat -L -v -n

Analizzate la lista e ricostruite i comandi di iptables necessari per creare le regole di nat

Errori comuni

State attenti a non commettere i seguenti errori nella scrittura dei comandi di iptables. Sono errori veniali e abbastanza comuni, ma spesso difficili da trovare.

I nomi delle interfacce di rete devono essere scritti correttamente. Se scrivete una regola con una interfaccia di rete inesistente, iptables non genera nessun messaggio di errore. Quindi:

- scrivete *eth0*, e non *etho* o *ethO*
- scrivete *eth1*, e non *ethl*
- scrivete *eth*, e non *eht*

Quando scrivete le regole di filtro, ricordatevi sempre di aggiungere l'obiettivo con l'opzione -j. Le regole con solo l'espressione di confronto e senza l'obiettivo sono ancora sintatticamente valide (iptables non si lamenta) ma ovviamente non hanno l'effetto desiderato.

Quando scrivete regole che contengono i qualificatori relativi alle interfacce di rete di ingresso e di uscita dei pacchetti, verificate sempre il percorso dei pacchetti utilizzando il diagramma di rete.

Diagramma di rete