CÀI ĐẶT VÀ CẤU HÌNH IPTABLES

Nguyễn Hồng Thái < *nhthai2005@gmail.com* >
Dept. of Telecommunication
Hô Chi Minh City University of Technology, South Vietnam

1. Giới thiệu về iptables

Iptables do Netfilter Organiztion viết ra để tăng tính năng bảo mật trên hệ thống Linux. Iptables cung cấp các tính năng sau:

- Tích hợp tốt với kernel của Linux.
- Có khả năng phân tích package hiệu quả.
- Lọc package dựa vào MAC và một số cờ hiệu trong TCP Header
- Cung cấp chi tiết các tùy chọn để ghi nhận sự kiện hệ thống
- Cung cấp kỹ thuật NAT
- Có khả năng ngăn chặn một số cơ chế tấn công theo kiểu DoS

2. Cài đặt iptables

Iptables được cài đặt mặc định trong hệ thống Linux, package của iptables là iptables-version.rpm hoặc iptables-version.tgz …, ta có thể dùng lệnh để cài đặt package này:

```
$ rpm –ivh iptables-version.rpm đối Red Hat
$ apt-get install iptables đối với Debian
```

- Khởi động iptables: `service iptables start`
- Tắt iptables: `service iptables stop`
- Tái khởi động iptables: `service iptables restart`
- Xác định trạng thái iptables: `service iptables status`

3. Cơ chế xử lý package trong iptables

Iptables sẽ kiểm tra tất cả các package khi nó đi qua iptables host, quá trình kiểm tra này được thực hiện một cách tuần tự entry đầu tiên đến entry cuối cùng.

Có ba loại bảng trong iptables:

Mangle table: chịu trách nhiệm biến đổi quality of service bits trong TCP header. Thông thường loại table này được ứng dụng trong SOHO (Small Office/Home Office).

Filter queue: chịu trách nhiệm thiết lập bộ lọc packet (packet filtering), có ba loại built-in chains được mô tả để thực hiện các chính sách về firewall (firewall policy rules).

- Forward chain: Cho phép packet nguồn chuyển qua firewall.
- Input chain: Cho phép những gói tin đi vào từ firewall.
- Output chain: Cho phép những gói tin đi ra từ firewall.

NAT queue: thực thi chức năng NAT (Network Address Translation), cung cấp hai loại built-in chains sau đây:

- Pre-routing chain: NAT từ ngoài vào trong nội bộ. Quá trình NAT sẽ thực hiện trước khi khi thực thi cơ chế routing. Điều này thuận lợi cho việc đổi địa chỉ đích để địa chỉ tương thích với bảng định tuyến của firewall, khi cấu hình ta có thể dùng khóa DNAT để mô tả kỹ thuật này.

- Post-routing chain: NAT từ trong ra ngoài. Quá trình NAT sẽ thực hiện sau khi thực hiện cơ chế định tuyến. Quá trình này nhằm thay đổi địa chỉ nguồn của gói tin. Kỹ thuật này được gọi là NAT one-to-one hoặc many-to-one, được gọi là Source NAT hay SNAT.
- OUPUT: Trong loại này firewall thực hiện quá trình NAT.

4. Target và Jumps

➢ *Jump* là cơ chế chuyển một packet đến một target nào đó để xử lý thêm một số thao tác khác.

➢ *Target* là cơ chế hoạt động trong iptables, dùng để nhận diện và kiểm tra packet. Các target được xây dựng sẵn trong iptables như:

- **ACCEPT**: iptables chấp nhận chuyển data đến đích.
- **DROP**: iptables khóa những packet.
- **LOG**: thông tin của packet sẽ gởi vào syslog daemon iptables tiếp tục xử lý luật tiếp theo trong bảng mô tả luật. Nếu luật cuối cùng không match thì sẽ drop packet. Với tùy chọn thông dụng là *--log-prefix="string"*, tức iptables sẽ ghi nhận lại những message bắt đầu bằng chuỗi *"string"*.
- **REJECT**: ngăn chặn packet và gởi thông báo cho sender. Với tùy chọn thông dụng là *--reject-with qualifier*, tức qualifier chỉ định loại reject message sẽ được gởi lại cho người gởi. Các loại qualifer sau: *icmp-port-unreachable (default), icmp-net-unreachable, icmp-host-unreachable, icmp-proto-unreachable, ...*
- **DNAT**: thay đổi địa chỉ đích của packet. Tùy chọn là *--to-destination ipaddress.*
- **SNAT**: thay đổi địa chỉ nguồn của packet. Tùy chọn là *--to-source <address>[-address][:<port>-<port>]*
- **MASQUERADING:** được sử dụng để thực hiện kỹ thuật NAT (giả mạo địa chỉ nguồn với địa chỉ của interface của firewall). Tùy chọn là *[--to-ports <port>[-<port>]]*, chỉ định dãy port nguồn sẽ ánh xạ với dãy port ban đầu.

5. Thực hiện lệnh trong iptables

Iptables command Switch	Mô tả
-t <table>	Chỉ định bảng cho iptables bao gồm: filter, nat, mangle tables.
-j <target>	Nhảy đến một target chain khi packet thỏa luật hiện tại.
-A	Thêm luật vào cuối iptables chain.
-F	Xóa tất cả các luật trong bảng lựa chọn.
-p <protocol-type>	Mô tả các giao thức bao gồm: icmp, tcp, udp và all
-s <ip-address>	Chỉ định địa chỉ nguồn
-d <ip-address>	Chỉ định địa chỉ đích

-i <interface-name>	Chỉ định "input" interface nhận packet
-o <interface-name>	Chỉ định "output" interface chuyển packet ra ngoài

Bảng 1: Bảng mô tả về iptables command Switch

Ví dụ 1: Firewall chấp nhận cho bất kỳ TCP packet đi vào interface eth0 đến địa chỉ 172.28.24.199

```
# iptables -A INPUT -s 0/0 -i eth0 -d 172.28.24.199 -p tcp -j ACCEPT
```

Ví dụ 2: Firewall chấp nhận TCP packet được định tuyến khi nó đi vào interface eth0 và đi ra interface eth1 để đến đích 172.28.2.2 với port nguồn bắt đầu 1024→65535 và port đích 8080

```
# iptables -A FORWARD -s 0/0 -i eth0 -o eth1 -d  172.28.2.2  -p tcp \
--sport 1024:65535 --dport  8080 -j ACCEPT
```

Ví dụ 3: Firewall cho phép gởi icmp echo-request và icmp echo-reply

```
# iptables -A OUPUT -p icmp --icmp-type echo-request -j ACCEPT
# iptables -A INPUT -p icmp --icmp-type echo-reply -j ACCEPT
```

Ví dụ 4: Chỉ định số lượng yêu cầu phù hợp cho một đơn vị thời gian theo dạng(/second, /minute, /hour. /day)

```
# iptables -A INPUT -p icmp -icmp-type echo-request -m limit --limit 1/s \
 -i eth0 -j ACCEPT
```

Ưu điểm của nó là giới hạn được số lượng kết nối, giúp cho ta chống được các cơ chế tấn công như DoS (Denial of Service attack).

Khóa chuyển (Switch)	Mô tả
-m multiport –sport<port,port>	Mô tả nhiều dãy sport, phải cách nhau bằng dấu "," và dùng tùy chọn –m
-m multiport –dport<port,port>	Mô tả nhiều dãy dport, phải cách nhau bằng dấu "," và dùng tùy chọn –m
-m multiport –ports<port,port>	Mô tả nhiều dãy port, phải cách nhau bằng dấu "," và dùng tùy chọn –m
-m –state<state>	Kiểm tra trạng thái: ESTABLISHED: đã thiết lập connection NEW: bắt đầu thiết lập connection RELATED: thiết lập connection thứ 2(FTP data transfer hoặc ICMP error)

Bảng 2: Mô tả một số thông số mở rộng

Ví dụ 5: Firewall chấp nhận TCP packet từ bất kỳ địa chỉ nào đi vào interface eth0 đến địa chỉ 172.28.24.195 qua interface eth1, source port từ 1024→65535 và destionation port là 8080 và 443 (dòng lệnh thứ 1). Packet trả về cũng được chấp nhận từ 172.28.2.2 (dòng lệnh thứ 2).

```
# iptables -A FORWARD -s 0/0 -i eth0 -d 172.28.24.195 -o eth1 -p tcp  \
--sport 1024:65535 -m multiport --dport 8080,443 -j ACCEPT
```

```
# iptables -A FORWARD -d 0/0 -i eth0 -s 172.28.2.2 -o eth1 -p tcp  \
-m state --state ESTABLISHED -j ACCEPT
```

6. Sử dụng chain tự định nghĩa

Thay vì sử dụng các chain đã được xây dựng trong iptables, ta có thể sử dụng User Defined chains để định nghĩa một chain name mô tả cho tất cả protocol-type cho packet. Ta có thể dùng User Defined chains thay thế chain dài dòng bằng cách sử dụng chain chính chỉ đến nhiều chain con.

Ví dụ 6:

```
# iptables -A INPUT -i eth0 -d 172.28.24.198 -j fast-input-queue
# iptables -A OUTPUT -o eth0 -s 172.28.2.2 -j fast-output-queue
# iptables -A fast-input-queue -p icmp -j icmp-queue-in
# iptables -A fast-output-queue -p icmp -j icmp-queue-out
# iptables -A icmp-queue-out -p icmp --icmp-type echo-request \
-m state --state NEW -j ACCEPT
# iptables -A icmp-queue-in-p icmp --icmp-type echo-reply \
-m state --state NEW -j ACCEPT
```

7. Lưu iptables script

Lệnh service iptables save để lưu trữ cấu hình iptables trong file */etc/sysconfig/iptables*. Khi ta khởi động lại thì chương trình iptables-restore sẽ đọc lại file script này và kích hoạt lại thông tin cấu hình. Định dạng của file như sau:

```
# Generated by iptables-save v1.2.8 on Thu Nov  9 15:47:54 2006
*nat
:PREROUTING ACCEPT [4169:438355]
:POSTROUTING ACCEPT [106:6312]
:OUTPUT ACCEPT [22:1332]
-A PREROUTING -d 172.28.24.199 -i eth0 -p tcp -m tcp --dport 80 -j DNAT --to-destination
192.168.1.2:8080
-A PREROUTING -d 172.28.24.199 -i eth0 -p tcp -m tcp --dport 8888 -j DNAT --to-destination
192.168.1.3:80
-A PREROUTING -i eth0 -p tcp -m tcp --dport 20:21 -j DNAT --to-destination 192.168.1.2:21
-A PREROUTING -i eth0 -p tcp -m tcp --dport 2020:2121 -j DNAT --to-destination 192.168.1.3:21
-A POSTROUTING -o eth0 -j SNAT --to-source 172.28.24.199
COMMIT
# Completed on Thu Nov  9 15:47:54 2006
# Generated by iptables-save v1.2.8 on Thu Nov  9 15:47:54 2006
*filter
:INPUT DROP [4011:414080]
:FORWARD ACCEPT [552:57100]
:OUTPUT ACCEPT [393:43195]
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -i ! eth0 -m state --state NEW -j ACCEPT
-A FORWARD -d 192.168.1.3 -i eth0 -p tcp -m tcp --dport 80 -j ACCEPT
COMMIT
# Completed on Thu Nov  9 15:47:54 2006
# Generated by iptables-save v1.2.8 on Thu Nov  9 15:47:54 2006
*mangle
:PREROUTING ACCEPT [5114:853418]
:INPUT ACCEPT [4416:773589]
```

```
:FORWARD ACCEPT [552:57100]
:OUTPUT ACCEPT [393:43195]
:POSTROUTING ACCEPT [945:100295]
COMMIT
# Completed on Thu Nov  9 15:47:54 2006
```

8. Phục hồi script khi mất script file

Để có thể phục hồi script khi mất script file. Đầu tiên, ta phải lưu script lại dùng lệnh*: iptables-save > script_du_phong*. Sau đó, ta có thể xem lại *script_du_phong* vừa lưu, dùng lệnh cat script_du_phong. Kết quả như sau:

```
# Generated by iptables-save v1.2.8 on Thu Nov  9 15:47:54 2006
*nat
:PREROUTING ACCEPT [4169:438355]
:POSTROUTING ACCEPT [106:6312]
:OUTPUT ACCEPT [22:1332]
-A PREROUTING -d 172.28.24.199 -i eth0 -p tcp -m tcp --dport 80 -j DNAT --to-destination
192.168.1.2:8080
-A PREROUTING -d 172.28.24.199 -i eth0 -p tcp -m tcp --dport 8888 -j DNAT --to-destination
192.168.1.3:80
-A PREROUTING -i eth0 -p tcp -m tcp --dport 20:21 -j DNAT --to-destination 192.168.1.2:21
-A PREROUTING -i eth0 -p tcp -m tcp --dport 2020:2121 -j DNAT --to-destination 192.168.1.3:21
-A POSTROUTING -o eth0 -j SNAT --to-source 172.28.24.199
COMMIT
# Completed on Thu Nov  9 15:47:54 2006
# Generated by iptables-save v1.2.8 on Thu Nov  9 15:47:54 2006
*filter
:INPUT DROP [4011:414080]
:FORWARD ACCEPT [552:57100]
:OUTPUT ACCEPT [393:43195]
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -i ! eth0 -m state --state NEW -j ACCEPT
-A FORWARD -d 192.168.1.3 -i eth0 -p tcp -m tcp --dport 80 -j ACCEPT
COMMIT
# Completed on Thu Nov  9 15:47:54 2006
# Generated by iptables-save v1.2.8 on Thu Nov  9 15:47:54 2006
*mangle
:PREROUTING ACCEPT [5114:853418]
:INPUT ACCEPT [4416:773589]
:FORWARD ACCEPT [552:57100]
:OUTPUT ACCEPT [393:43195]
:POSTROUTING ACCEPT [945:100295]
COMMIT
# Completed on Thu Nov  9 15:47:54 2006
```

Sau đó, sửa file *script_du_phong* và nạp lại iptables thông qua lệnh *iptables-restore*

```
# iptables-restore < script_du_phong
```

Cuối cùng, ta dùng lệnh để lưu trữ lại các luật vào file cấu hình:

```
# service iptables save
```

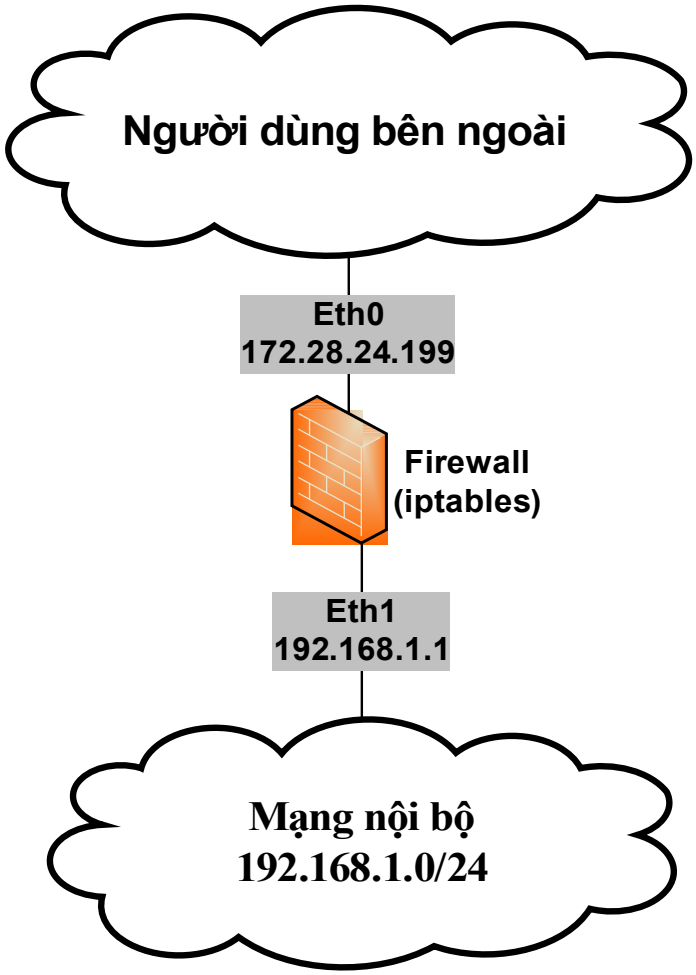
9. Load kernel module cần cho iptables

Ứng dụng iptables yêu cầu load một số module sau:

- *iptable_nat* module cho NAT.
- *ip_conntrack_ftp* module cần cho FTP support
- *ip_conntrack* module để theo dõi trạng thái của TCP connect.
- *ip_nat_ftp* module cần cho việc load FTP servers sau NAT firewall.

10. Một số giá trị khởi tạo của iptables

```
######## Internal-Firewall.sh cript
######## Cho phép tự chạy script bằng shell
#!/bin/sh
#### Gán lệnh vào biến
IPTABLES=/sbin/iptables
######### Các giá trị khởi tạo
INTERNAL_LAN="192.168.1.0/24"    # Địa chỉ mạng LAN
INTERNAL_LAN_INTERFACE="eth1"  # Interface nối đến mạng LAN
INTERNAL_LAN_INTERFACE_ADDR="192.168.1.1" ##Địa chỉ int eth1
EXTERNAL_INTERFACE="eth0"                ## Interface public
EXTERNAL_INTERFACE_ADDR="172.28.24.199"    ## Địa chỉ eth0
$IPTABLES -F FORWARD                ## Xóa các luật của FORWARD chain
$IPTABLES -F INPUT            ## Xóa các luật của INPUT chain
$IPTABLES -F OUTPUT                ## Xóa các luật của OUTPUT chain
$IPTABLES -P FORWARD DROP  ## Mặc định FORWARD chain là DROP
$IPTABLES -P OUPUT ACCEPT    ## Mặc định OUTPUT chain là ACCEPT
$IPTABLES -P INPUT DROP   ## Mặc định INPUT chain là DROP
#++++++++++++++++++++++++++++++++++++++++++++++++++++++
## Cho phép tất cả các packet đi vào loopback với tất cả các protocol
$IPTABLES -A INPUT -i lo -p all -j ACCEPT
## Cho phép các gói tin đi vào firewall chỉ với icmp protocol
$IPTABLES -A INPUT -p icmp -j ACCEPT
## Cho phép các packet đi vào eth1 có địa chỉ nguồn là địa chỉ của LAN
$IPTABLES -A INPUT -i $INTERNAL_LAN_INTERFACE -s $INTERNAL_LAN -j ACCEPT
# Cho phép các packet ra từ eth1 có địa chỉ đích là địa chỉ của LAN
$IPTABLES -A OUTPUT -o $INTERNAL_LAN_INTERFACE \
-d $INTERNAL_LAN -j ACCEPT
# Thực hiện NAT bằng cách đổi địa chỉ nguồn của gói tin trước khi định tuyến,
#####đi ra từ eth0 với bất kỳ địa chỉ nào khác địa chỉ của LAN
$IPTABLES -A -t nat -A POSTROUTING -o $EXTERNAL_LAN_INTERFACE \
-d ! $INTERNAL_LAN -j MASQUERADE
## Cho phép các gói tin đi qua firewall có địa chỉ nguồn hoặc địa chỉ đích
########là địa chỉ của LAN
$IPTABLES -A FORWARD -s $INTERNAL_LAN -j ACCEPT
$IPTABLES -A FORWARD -d $INTERNAL_LAN -j ACCEPT
```



Hình 1: Mô hình mạng mô tả cho script internal-firewall.sh

11. Một số ví dụ về Firewall

Ví dụ 7: Cho phép truy xuất DNS đến Firewall

```
# iptables -A  OUTPUT -p udp -o eth0 --dport 53 –sport 1024:65535 -j ACCEPT
# iptables -A  INPUT -p udp -i eth0 --dport 53 –sport 1024:65535 -j ACCEPT
```

Ví dụ 8: Cho phép www và ssh truy xuất tới Firewall

```
# iptables -A OUTPUT -o eth0 -m state --state ESTABLISHED, RELATED  -j ACCEPT
# iptables -A INPUT -p  tcp -i eth0 --dprt 22 --sport 1024:65535 -m state \
--state NEW -j ACCEPT
# iptables -A INPUT -p  tcp -i eth0 --dport 80 --sport 1024:65535 -m state \
--state NEW -j ACCEPT
```

Ví dụ 9: Masquerading (many to One NAT) là kỹ thuật NAT Many to One để cho phép nhiều máy cục bộ có thể sử dụng địa chỉ IP chính thức (được cung cấp từ ISP) để truy cập internet.

```
#########Cho phép script tự khởi động với shell
#! /bin/sh
######### Nạp module iptable_nat
modprobe iptable_nat
```

```
######## Bật chức năng định tuyến
echo 1 > /proc/sys/net/ipv4/ip_forward
##### Cho phép sử dụng NAT giả mạo trong đó
###### - Interface eth0 là interface liên kết mạng internet
###### - Interface eth1 liên kết đến mạng nội bộ
iptables -A POSTROUTING -t nat -o eth0 -s 192.168.1.0/24 -d 0/0 -j MASQUERADE
# Cho phép đi qua firewall trong trường các trường hợp các kết nối là mới,
###đã thiết lập hoặc có liên hệ
iptables -A FORWARD -t  filter -o eth0  -m state  \
--state NEW, ESTABLISHED, RELATED -j ACCEPT
iptables -A FORWARD -t  filter -i eth0  -m state  \
--state NEW, ESTABLISHED, RELATED -j ACCEPT
```

Ví dụ 10: Thực hiện Port Forwarding với DHCP DSL. Trong trường hợp ta nhận 1 địa chỉ IP động từ ISP và ta muốn sử dụng địa chỉ này để cung cấp cho tất cả địa chỉ trong mạng nội bộ và public các server nội bộ ra bên ngoài internet. Tất cả các yêu cầu trên có thể giải quyết bằng cách sử dụng kỹ thuật Port Forwarding.

```
######### Cho script chạy với shell
#!/bin/sh
##### Nạp module iptable_nat
modprobe iptable_nat
##### Gán eth0 lên biến external_int
external_int = "eth0"
##### Thực hiện lấy ip mà DHCP cấp cho máy này
external_ip = " `ifconfig $external_int | grep 'inet addr' | awk '{print $2}' |  \
sed –e 's/.*://'` "
##### Cho phép các interface forward với nhau
echo 1 > /proc/sys/net/ipv4/ip_forward
##### Thực hiện đổi địa chỉ đích trước khi thực hiện routing
iptables - nat - PREROUTING - tcp -ieth0 - $external_ip --dport 80 \
--sport 1024:65535 - DNAT –to 192.168.1.2:8080
# Cho phép các packet FORWARD qua firewall trong các trường hợp dưới đây
iptables -A FORWARD -p tcp -i eth0 -o eth1 -d 192.168.1.2 -dport 8080  \
-sport 1024:65535 -m state --state NEW -j ACCEPT
iptables -A FORWARD -t filter -o eth0 -m state \
--state NEW, ESTABLISHED, RELATED -j ACCEPT
iptables -A FORWARD -t filter -i eth0 -m state \
--state NEW, ESTABLISHED, RELATED -j ACCEPT
```

Ví dụ 11: Thực hiện NAT với ip tĩnh.

- Sử dụng one to one NAT để cho phép server có địa chỉ 192.168.1.2 trên mạng nội bộ truy xuất ra ngoài internet thông qua địa chỉ 172.28.24.199.
- Tạo many to one NAT để cho mạng 192.168.1.0 có thể truy xuất đến tất cả các server trên internet thông qua địa chỉ 172.28.24.199.

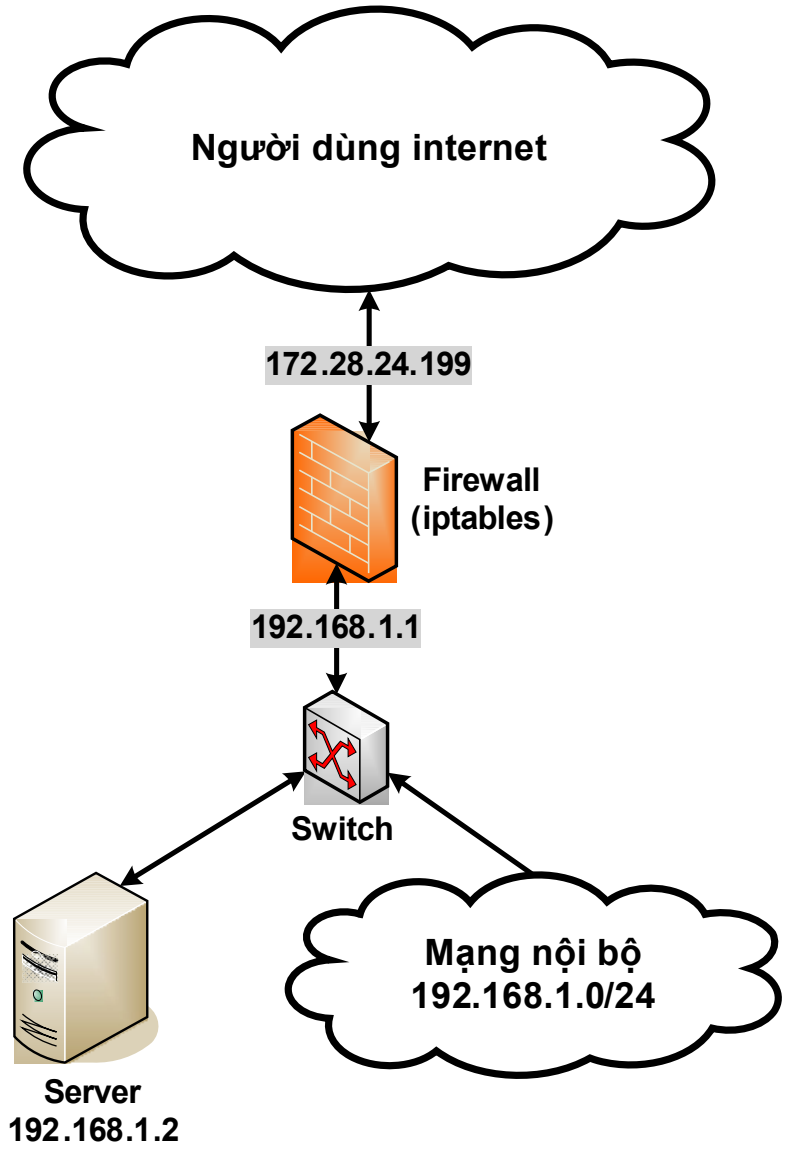
```
##### Cho script chạy với shell
#! /bin/sh
## Load module và cho phép forward giữa các card mạng
modprobe iptable_nat
echo 1 > /proc/sys/net/ipv4/ip_forward
```

```
# Thực hiện DNAT để đổi địa chỉ đích thành địa chỉ của server
#### mạng nội bộ (192.168.1.2) khi truy cập đến 172.28.24.199
iptables -t nat -A PREROUTING -d 172.28.24.199 -i eth0 \
-j DNAT to-destination 192.168.1.2
## Thực hiện SNAT để đổi địa chỉ nguồn từ 192.168.1.2
########################## → 172.28.24.199
iptables -t nat -A POSTROUTING -s 192.168.1.2 -o eth0  \
-j SNAT --to-source 172.28.24.199
## Tương tự như trên, cho phép máy từ LAN truy cập đến các server
iptables -t nat -A POSTROUTING -s 192.168.1.0/24 -o eth0 \
-j SNAT --to-source 172.28.24.199
## Cho phép bên ngoài truy xuất vào server (192.168.1.2)
#####thông qua các port 80, 443, 22
iptables -A FORWARD  -p tcp -i eth0 -o eth1 -d 192.168.1.2  \
-m multiport --dport 80,443,22 -m state --state NEW -j ACCEPT
# Cho phép chuyển tất cả các NEW, ESTABLISHED SNAT connections
#### bắt đầu từ homework và thực sự đã thiết lập trước đó với DNAT connections
iptables -A  FORWARD -t filter -o eth0 -m state \
--state NEW, ESTABLISHED, RELATED -j ACCEPT
# Cho phép chuyển tất cả các connections bắt đầu từ internet đã được thiết lập
##########thông qua từ khóa NEW
iptables -A FORWARD -t filter -i eth0 -m state   \
--state ESTABLISHED, RELATED -j ACCEPT
```

Ví dụ 12: Tạo một proxy

```
########### Cho phép script chạy với sh
#!/bin/sh
INTIF="eth1"  ## Gán chuỗi "eth1" vào INTIF
EXTIF="eth0"  ## Gán chuỗi "eth0" vào EXTIF
######## Thực hiện lấy địa chỉ ip mà DHCP cấp
EXTIP="`/sbin/ifconfig eth0 | grep 'inet addr' | awk '{print $2}'  | sed -e 's/.*://'`"
###### Load module cần thiết
/sbin/depmod -a
/sbin/modprobe ip_tables
/sbin/modprobe ip_conntrack
/sbin/modprobe ip_conntrack_ftp
/sbin/modprobe ip_conntrack_irc
/sbin/modprobe iptable_nat
/sbin/modprobe ip_nat_ftp
## Cho phép các card mạng có thể forward được với nhau
echo "1" > /proc/sys/net/ipv4/ip_forward
###### Cho phép thực hiện với ip động
echo "1" > /proc/sys/net/ipv4/ip_dynaddr
iptables -P INPUT ACCEPT     ## Mặc định INPUT chain là ACCEPT
iptables -F INPUT            ## Xóa các luật trong INPUT chain
iptables -P OUTPUT ACCEPT  ## Mặc định OUTPUT chain là ACCEPT
iptables -F OUTPUT           ## Xóa các luật trong OUTPUT chain
iptables -P FORWARD DROP  ## Mặc định FORWARD chain là DROP
iptables -F FORWARD                ## Xóa các luật trong FORWARD chain
iptables -t nat -F                 ## Xóa tất cả các luật của bảng nat
```

```
## Cho phép FORWARD đi vào  eth0 đi ra eth1 trong trường hợp
#####các connection là  ESTABLISHED, RELATED
iptables -A FORWARD -i $EXTIF -o $INTIF -m state  \
--state ESTABLISHED,RELATED -j ACCEPT
######## Và ngược lại
iptables -A FORWARD -i $INTIF -o $EXTIF -j ACCEPT
## Thực hiện đổi địa chỉ nguồn trong trường hợp đi ra từ eth0
iptables -t nat -A POSTROUTING -o $EXTIF -j MASQUERADE
```



Hình 2: Mô hình mạng LAN với server

Kết quả của việc cấu hình proxy trên, như sau:

```
# Generated by iptables-save v1.2.8 on Thu Nov  9 10:02:42 2006
*nat
:PREROUTING ACCEPT [536:76253]
:POSTROUTING ACCEPT [2:119]
:OUTPUT ACCEPT [15:909]
-A POSTROUTING -o eth0 -j MASQUERADE
COMMIT
```

```
# Completed on Thu Nov  9 10:02:42 2006
# Generated by iptables-save v1.2.8 on Thu Nov  9 10:02:42 2006
*filter
:INPUT ACCEPT [132:12857]
:FORWARD DROP [0:0]
:OUTPUT ACCEPT [0:0]
:RH-Firewall-1-INPUT - [0:0]
-A FORWARD -i eth0 -o eth1 -m state --state RELATED,ESTABLISHED -j ACCEPT
-A FORWARD -i eth1 -o eth0 -j ACCEPT
-A RH-Firewall-1-INPUT -i lo -j ACCEPT
-A RH-Firewall-1-INPUT -i eth0 -j ACCEPT
-A RH-Firewall-1-INPUT -p icmp -m icmp any -j ACCEPT
-A RH-Firewall-1-INPUT -p esp -j ACCEPT
-A RH-Firewall-1-INPUT -p ah -j ACCEPT
-A RH-Firewall-1-INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A RH-Firewall-1-INPUT -p tcp -m state --state NEW -m tcp --dport 80 -j ACCEPT
-A RH-Firewall-1-INPUT -p tcp -m state --state NEW -m tcp --dport 21 -j ACCEPT
-A RH-Firewall-1-INPUT -p tcp -m state --state NEW -m tcp --dport 22 -j ACCEPT
-A RH-Firewall-1-INPUT -p tcp -m state --state NEW -m tcp --dport 23 -j ACCEPT
-A RH-Firewall-1-INPUT -p tcp -m state --state NEW -m tcp --dport 25 -j ACCEPT
-A RH-Firewall-1-INPUT -j REJECT --reject-with icmp-host-prohibited
COMMIT
# Completed on Thu Nov  9 10:02:42 2006
```

12. Khắc phục sự cố trên iptables

- Với phần trình bày về iptables ở trên là khá đầy đủ. Với kiến thức trên, chúng ta có thể thực hiện những yêu cầu về lọc gói tin một cách khá tốt. Nhưng phần trên chỉ trình bày cách thực hiện với iptables mà không nêu ra cách khắc phục sự cố trên iptables. Trong phần này, chúng tôi sẽ trình bày cách khắc phục sự cố về iptables nói riêng, những phần mềm trên hệ điều hành mã nguồn mở nói chung.
- Cách vận hành và bảo trì những phần mềm trên Linux thường sẽ qua những bước sau đây: cài đặt, cấu hình, vận hành và khắc phục sự cố khi có lỗi. Trong những phần trên, chúng tôi đã trình bày cách cài đặt, cấu hình và vận hành. Còn phần khắc phục sự cố về những phần mềm trên Linux, thường thì người quản trị sẽ đọc file Log, cụ thể với iptables thì chúng ta cần kiểm tra Firewall Logs.
- Firewall logs được ghi nhận vào file */var/log/message*. Để cho phép iptables ghi vào */var/log/message*, chúng ta phải cấu hình như sau:

```
iptables -A  OUTPUT -j LOG
iptables -A  INPUT -j LOG
iptables -A  FORWARD -j LOG
iptables -A  OUTPUT -j DROP
iptables -A  INPUT -j DROP
iptables -A  FORWARD -j DROP
```

13. iptables không khởi động

- Khi ta khởi động iptables thì ta dùng lệnh /etc/init.d/iptables start. Lúc này, iptables gọi script trong file /etc/sysconfig/iptables. Do đó, nếu file này không tồn tại hoặc bị lỗi thì iptables sẽ không thực hiện được.
- Khi ta thay đổi cấu hình trên iptables thì ta phải dùng lệnh service iptables save để lưu lại các thông tin cấu hình. Sau đó, mới tiến hành restart lại iptables.

Ví dụ 13:

```
# service iptables start                ## Khởi động iptables
# touch /etc/sysconfig/iptables         ## Tạo file iptables trống
##Thiết lập quyền cho file này
# chmod 600 /etc/sysconfig/iptables
# service iptables start
Applying iptables firewall rules: [OK]
```

TÀI LIỆU THAM KHẢO

[1] Nguyễn Thị Điệp và Tiêu Đông Nhơn, *Giáo trình Dịch vụ mạng Linux,* Đại học Quốc Gia Thành phố Hồ Chí Minh 12/2005.
[2] How To Set Up A Debian Linux Proxy Server by Debian's Web.