IPTABLES el firewall incluido a nivel del kernel en las distribuciones Linux

Donde colocar las reglas del firewall

Se pueden escribir en el modo comando y si ponemos `/etc/init.d/iptables save`
Se salvan en **`/etc/sysconfig/iptables`**

para que estén activas siempre colocarlas en el fichero el cual podemos editar y escribirlas directamente. **`/etc/sysconfig/iptables`**

Para borrar todas las reglas de iptables

```
iptables -F
```

Sistema asistido para configurar IPTABLES

system-config-firewall

Funcionamiento básico de iptables es el siguiente:

Existen cadenas de : INPUT, OUTPUT y FORWARD.

Dentro de las cadena hay reglas que se evaluan por orden

Cuando una regla se evalua de forma positiva, es dirigida a un TARGET. Puede ser aceptada, denegada, eliminada, logada u otras muchas consas más

Sintaxis IPTABLES Básica

```
iptables [-t tabletype] COMMAND  [-m MATCH_EXTENSION] -j TARGET
```

Para listar las reglas actuales

```
/sbin/iptables -L
```

La relación de puertos y servicios se puede ver en

```
/etc/services
```

Iniciar/Parar/Reiniciar Iptables

- sudo service iptables start
- sudo service iptables stop
- sudo service iptables restart

Argumentos de una orden:

- -A –append → agrega una regla a una cadena.
- -D –delete → borra una regla de una cadena especificada.
- -R –replace → reemplaza una regla.
- -I –insert → inserta una regla en lugar de una cadena.
- -L –list → muestra las reglas que le pasamos como argumento.
- -F –flush → borra todas las reglas de una cadena.
- -Z –zero → pone a cero todos los contadores de una cadena.
- -N –new-chain → permite al usuario crear su propia cadena.
- -X –delete-chain → borra la cadena especificada.
- -P –policy → explica al kernel qué hacer con los paquetes que no coincidan con ninguna regla.
- -E –rename-chain → cambia el orden de una cadena.

Condiciones para Iptables:

- -p –protocol → la regla se aplica a un protocolo.
- -s –src –source → la regla se aplica a una IP de origen.
- -d –dst –destination → la regla se aplica a una Ip de destino.
- -i –in-interface → la regla de aplica a una interfaz de origen, como eth0.
- -o –out-interface → la regla se aplica a una interfaz de destino.

Condiciones TCP/UDP

- -sport –source-port → selecciona o excluye puertos de un determinado puerto de origen.
- -dport –destination-port → selecciona o excluye puertos de un determinado puerto de destino.

Reglas que bloquean

- iptables -P INPUT DROP
- iptables -P FORWARD DROP
- iptables -P OUTPUT DROP

Regla que filtre un determinado puerto

iptables -A INPUT -p tcp –sport 22 22 → crea una regla para el puerto de origen tcp 2222

bloquear el tráfico procedente de una IP determinada

iptables -A INPUT -p tcp -m iprange –src-range 192.168.1.13-192.168.2.19

bloquear una MAC determinada

iptables -A INPUT -m mac –mac-source 00:00:00:00:00:01

Una vez escritas las reglas que necesitemos y queremos aplicar , las guardarmos tecleando **sudo service iptables save**

Ver el estado del firewall

- iptables -L -n -v

L muestra las líneas abiertas.

V permite recibir más información sobre las conexiones y

N nos devuelve las direcciones IP y sus correspondientes puertos sin pasar por un servidor DNS.

Eliminar las reglas existentes

Borra toda la configuración del firewall

iptables –F

Permitir conexiones entrantes

Teclearemos los siguientes parámetros:

- iptables -A INPUT -i [interface] -p [protocolo] –dport [puerto] -m state –state NEW,ESTABLISHED -j ACCEPT

-i: debemos configurar la interfaz, por ejemplo, eth0. Esto es útil en caso de tener varias tarjetas de red, si tenemos sólo una, no tenemos por qué especificar este parámetro.

-p: protocolo. Debemos especificar si el protocolo será TCP o UDP.

–dport: el puerto que queremos permitir, por ejemplo, en caso de HTTP sería el 80.

Un ejemplo para permitir las conexiones entrantes desde páginas web:

- iptables -A INPUT -i eth0 -p tcp –dport 80 -m state –state NEW,ESTABLISHED -j ACCEPT

Permitir las conexiones salientes

- iptables -A OUTPUT -o [interfaz] -p [protocolo] –sport [puerto] -m state –state ESTABLISHED -j ACCEPT

-o: debemos configurar la interfaz, por ejemplo, eth0, al igual que en el caso anterior.

-p: protocolo. Debemos especificar si el protocolo será TCP o UDP.

–sport: el puerto que queremos permitir, por ejemplo, en caso de HTTPS sería el 443.

Un ejemplo para permitir el tráfico saliente hacia páginas web:

- iptables -A OUTPUT -o eth0 -p tcp –sport 80 -m state –state ESTABLISHED -j ACCEPT

Permitir los paquetes ICMP

Por defecto, el ping está deshabilitado y hay que habilitarlo manualmente

Para poder hacer ping a otros servidores:

- iptables -A OUTPUT -p icmp –icmp-type echo-request -j ACCEPT

Para permitir recibir solicitudes de ping de otros equipos:

- iptables -A INPUT -p icmp –icmp-type echo-reply -j ACCEPT

Permitir que el tráfico interno salga a internet

En el caso de tener 2 tarjetas de red (eth0 en local y eth1 conectada a internet) podemos configurar el firewall para que reenvíe el tráfico de la red local a través de internet. Para ello escribiremos:

- iptables -A FORWARD -i eth0 -o eth1 -j ACCEPT

Bloquear y prevenir ataques DDoS

- iptables -A INPUT -p tcp –dport 80 -m limit –limit 25/minute –limit-burst 100 -j ACCEPT

Consultar los paquetes rechazados por iptables

- iptables -N LOGGING