POWERED BY LINUX

CONFIGURACIÓN DE IPTABLES
MAS ALLA DE UN SIMPLE FIREWALL

III Semana del pingüino (PiMI's)
Presentado por:
Ing. Alberto Grájeda Chacón

CONTENIDO

POWERED BY LINUX

1. Introducción

2. Configuración de iptables

3. Mas allá de la configuración tradicional

POWERED BY LINUX

1

INTRODUCCIÓN

Seguridad

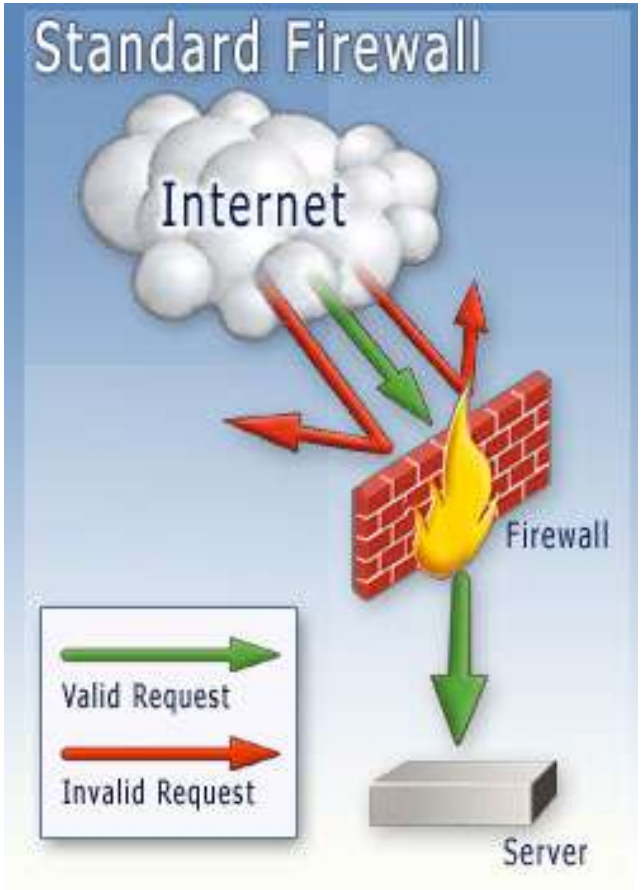
En la mayoría de las organizaciones la seguridad en redes es una parte mas de la red y del sistema.

Existen muchas amenazas en internet que deben se deben proteger:

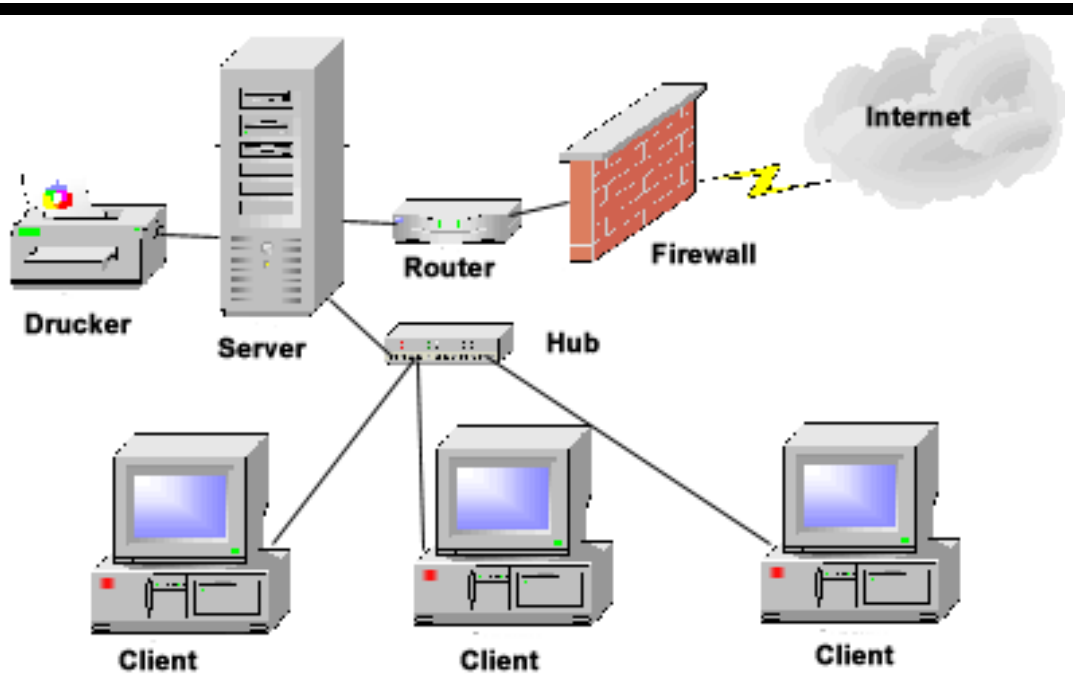
- Hackers maliciosos.
- Propagación de gusanos (desde, hacia)
- DoS (Denegación de Servicios)
- Muchos otros.

Para ello existen opciones de Caja (PIX, netscreen, zywall) y de software (netfilter, programas varios).

Hoy en día también existen firewalls personales.



Seguridad (2)



Se debe tomar en cuenta:

- El tráfico entrante desde internet hacia nuestra red.
- El tráfico saliente desde nuestra red hacia internet.

NO EXISTE UNA SOLUCIÓN QUE IMPLEMENTE 100% DE SEGURIDAD EN NUESTRA RED.

POWERED BY LINUX

2

CONFIGURACIÓN DE IPTABLES

NetFilter / iptables

Proporciona una solución robusta y flexible para implementar firewalls.

NetFilter es el nombre del proyecto e iptables el nombre del software en linux que viene integrado en el kernel 2.4.x

Es el sucesor de ipchains y ipfwadmin.

Sitio web: http://www.netfilter.org

Bajando el código fuente (1.2.11) y compilando funciona en la mayoría de los kernels, caso contrario, recompilar el kernel.

Fundamentos para implementar un firewall:

- Instalar un servidor con 2 tarjetas de red (Red Outside, Red Inside). O instalar un servidor con 3 tarjetas (Red DMZ)
- *Filosofía:* Abro el trafico que necesito y cierro todos los demás.
- Tráfico de entrada y salida mediante puertos TCP, UPD, ICMP.

Guía de Sintaxis de iptables

Para ver la ayuda en línea ejecutar el comando desde terminal:
man iptables

chain= {INPUT,FORWARD,OUTPUT}

Para ver cuales son las reglas activas del iptables:
iptables -L [chain] – Lista la configuración actual de iptables.

Para hacer modificaciones
iptables -A [chain] – Adiciona una regla en el “chain” deseado en la configuración actual.

iptables -D [chain] – Borra una regla existente en el “chain” deseado en la configuración actual.

iptables -R [chain] – Reemplaza una regla existente en el “chain” deseado en la configuración actual.

Guía de Sintaxis de iptables (2)

iptables - I [chain] - Inserta una nueva regla en el “chain” deseado en la configuración actual.

iptables -N [chain] – Crea un nuevo “chain”

iptables -X [chain] – Borra un “chain”

Borrando las reglas
iptable -X – Borra todas las cadenas

iptables -F – Borra el contenido de todas las tablas

Importando y exportando
iptables-save > filename - exporta la configuración actual en un archivo plano.

iptables-restore < filename - importa la configuración actual en un archivo plano.

CONFIGURANDO IPTABLES

- Borrar toda las tablas (reglas) que están creadas para empezar con una configuración desde cero.

- Crear un archivo donde esten las reglas listadas, no olvidar adicionar las reglas:
  - Probar que los módulos están arriba.
  - echo "1" > /proc/sys/net/ipv4/ip_forward
  - Listar las reglas del firewall tradicional.

Para hacer que los accesos se vayan a un log:

- Adicionar la siguiente linea en: /etc/syslog.conf

```
#IPTables logging
# kernel messages.
kern.debug;kern.info /var/log/firewall
```

- Reiniciar el demonio de syslog: service syslog restart

POWERED BY LINUX

3

MAS ALLA DE LA CONFIGURACIÓN TRACIONAL

CONFIGURANDO IPTABLES

La configuración típica de un firewall básico son filtradores de paquetes. Ven los paquetes que van pasando en la red, y hacen elecciones de como manejar los paquetes. La configuración dirá cuales pasan y cuales son rechazados.

Las reglas se básicas se hacen mediante sobre la dirección IP y el número de puerto.

Iptables también puede hacer un seguimiento a la conexión que puede ir adicionado a las reglas.

Iptables tiene mucho mas que ofrecer que un simple criterio de filtro de paquetes. Existen muchos criterios avanzados de iptables que llevarían mucho tiempo de explicar, en la presentación solo describo la existencia de algunos y los invito al camino de la exploración.

POM (Patch-o-matic)

Netfilter tiene 2 grupos de componentes:

- Kernel
- Modo usuario (iptables y sus utilitarios, librerias, manuales, etc)

Importante: Tener cuidado aplicar parches al kernel.

Netfilter provee un POM, que es una colección de parches y un script para aplicarlos sin mayores problemas.

Bajar POM desde: ftp.netfilter.org/pub/patch-o-matic/

Para la instalación: KERNEL_DIR=/usr/src/linux-2.4 ./runme extra

Manejo de cadenas

El módulo de cadenas es probablemente el módulo mas usado. Permite que los paquetes sean emparejados contra cadenas.

El módulo tiene muchas características pero hay que aplicarlo con cuidado.

Ejemplo: Bloquear el bajar ELF ejecutables de la web.

```
iptables -A FORWARD -i eth0 -p tcp –sport 80 -m string –string '|7F|ELF' -j DROP
```

Múltiples puertos

La extensión mport permite especificar en una simple regla un número de puertos, rangos y usar una sintaxis extendida.

Ejemplo: Permitir el uso de terminales X-Window, web y email

```
iptables -A INPUT -p tcp -m mport --dports 80,110,25,6000:6003 -j ACCEPT
```

Sin el uso de mport tendríamos que usar reglas separadas como:

```
iptables -A INPUT -p tcp --dports 80 -j ACCEPT
iptables -A INPUT -p tcp --dports 110 -j ACCEPT
iptables -A INPUT -p tcp --dports 21 -j ACCEPT
iptables -A INPUT -p tcp --dports 6000:6003 -j ACCEPT
```

Ganamos PERFORMANCE!!!

Reglas basadas en tiempo

El módulo de tiempo permite introducir reglas de tiempo en iptables.

Ejemplo: No permitir el uso del servicio web entre las 4 a las 6:30 am, los viernes por razones de mantenimiento del servidor.

```
iptables -A INPUT -p tcp -m time –timestart 04:00 –timestop 06:30 –days Fri \
--syn -j REJECT
```

Muchas ventajas!!!

Usando Tar pits

El concepto detrás de un tarpit es bastante simple. Las conexiones llegan, pero no vuelven. IPtables permite esto con un puerto tarpit que acepta cualquier conexión entrante de TCP. Cuando la transferencia de datos comienza a ocurrir, el tamaño de la ventana del TCP se fija a cero, así que ningun dato se puede transferir en la sesión. La conexión entonces se queda abierta, y cualquier petición para cerrar la sesión es ignorada. Esto significa que el atacante debe esperar por un “timeout” de la conexión. Esta es negativo para el comportamiento en las herramientas de exploración automatizadas (como gusanos) porque confían en una vuelta rápida de sus víctimas potenciales.

Ejemplo: Para confundir a los atacantes para hacer paracer linux a una máquina windows, podemos hacer que el netbios responda a pedidos, luego pasamos la conexión al tar pit.

```
iptables -A INPUT -p tcp -m tcp -m mport --dports 135,139,1025 -j TARPIT
```

Ejemplo2: Todos los puertos con tar pit y abrir solo los necesarios

```
iptables -A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
iptables -A INPUT -p tcp -m tcp --dport 25 -j ACCEPT
iptables -A INPUT -p tcp -m tcp -j TARPIT
```

Randomizando

El módulo de random, empareja los paquetes basados en nada mas que una elección randómica.

Ejemplo: Distribuir el tráfico entre 3 servidores que son mirrors. La primera regla envía 33% de las conexiones al servidor 192.168.1.100, los siguientes 33% son enviados a 192.168.1.101 y el resto al servidor 192.168.1.102

```
iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 80 --syn -m random \
--average 33 -j DNAT --to-destination 192.168.0.100:80

iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 80 --syn -m random \
--average 50 -j DNAT --to-destination 192.168.0.101:80

iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 80 --syn -j DNAT \
   --to-destination 192.168.0.102:80
```

Muchos más módulos

Cuando corre el script runme, describe todas las descripciones de los paches que va instalando. Como por ejemplo:

- Seguimiento a la conexión de RSH, MMS (sistemas multimedia), PPTP, el RPC, etc.
- Ayuda extendida para el acceso de la configuración e información a través del sistema de archivos /proc
- Ayuda extendida a las características IPv6.
- Manipulación de opciones, de la TTL y más en paquetes IP.
- Control más específico sobre conexiones NAT.
- Control de límites en ancho de banda.
- Etc.

Para conseguir ayuda en línea de los módulos:
iptables -m random -help

Conclusión

La parte divertida del software libre es que no oculta nada de verdad.

Todo lo que existe está esperando a un buscador que lo encuentre

POWERED BY LINUX

MUCHAS GRACIAS !!!