TECNOLÓGICO DE MONTERREY

IPTables

Roberto Gómez Cárdenas
rogoca@gmail.com
http://www.cryptomex.org

TECNOLÓGICO DE MONTERREY

Netfilter/IPTables

- Las dos piezas principales de producto firewall disponibles gratuitamente para distribuciones Linux
- IPTables es usado para construir las reglas.
- Netfilter es puente entre núcleo linux y las IPTables
- IPTables es como se conoce al módulo Netfilter
  - herramienta estándar actual de firewall de Linux
- Administradores especifican que reglas que protocolos o tipos de tráficos se deben seguir.
  - cuando empieza conexión con protocolos IPTables añade una entrada de estado para la conexión en cuestión

TECNOLÓGICO DE MONTERREY

Las tablas de IPTables

- Tabla constituida de un numero arbitrario e ilimitado de cadenas
- Una cadena es una secuencia lineal de reglas
- Las reglas estas constituidas por
  - un patrón: reconocer paquetes de acuerdo a un número indeterminado de criterios
  - una acción (llamado target) a tomar en caso de reconocer el paquete

Lámina 3 Dr. Roberto Gómez Cárdenas

TECNOLÓGICO DE MONTERREY

Las tres tablas de IPTables y sus cadenas de base

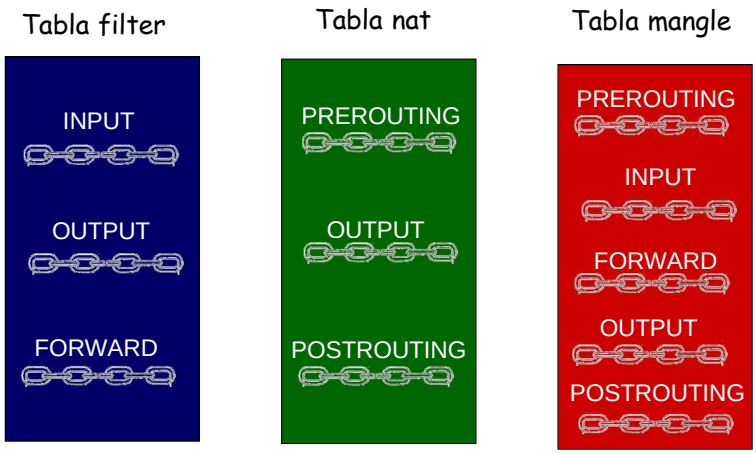

Lámina 4 Dr. Roberto Gómez Cárdenas

Seleccionando la tabla

- Cada regla a añadir a IPTables necesita insertarse o adjuntarse en una cadena de alguna de las tres tablas.
  - tablas disponibles depende configuración núcleo.
- Opción -t comando iptables permite elegir la tabla

```
iptables -t filter -A INPUT --source 192.168.0.1 -j DROP
```

  - si no se propociona la regla será insertada en la tabla "filter" que es la tabla por defecto.

Lámina 5 Dr. Roberto Gómez Cárdenas

Operaciones manejo cadenas usuario

iptables [–t tabla] comando

-N crear una nueva cadena usuario
-X borrar una cadena vacía
-P cambiar la política por default de una cadena base
-L listar las reglas de una cadena usuario
junto con opción **–v** despliega paquetes tratados por cada una de las reglas
-F vaciar las reglas de una cadena usuario
-Z poner a cero el contador de paquete y de byte en todas las reglas en una cadena
-E vieja nueva: renombra la cadena vieja a la cadena nueva

Lámina 6 Dr. Roberto Gómez Cárdenas

Desplegando las reglas de la cadena filter

```
# iptables -L
Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
#
```

Lámina 7 Dr. Roberto Gómez Cárdenas

Ejercicio: desplegando las reglas cadenas

- Desplegar reglas de la tabla nat

  # iptables –t nat –L

- Desplegar reglas de la tabla mangle

  # iptables –t mangle –L

- Desplegar reglas de la tabla filter

  # iptables –t filter –L

Lámina 8 Dr. Roberto Gómez Cárdenas

TECNOLÓGICO DE MONTERREY

Acciones por default

- Cadenas de base cuentan con una política por default
- Cuando paquete llega al final de una cadena es posible aplicarle una acción por default
- Acciones posibles: DROP o ACCEPT
- Ejemplo

```
iptables -P INPUT DROP
iptables -P FORWARD DROP
iptables -P OUTPUT ACCEPT
```

- Cadenas usuario no cuentan con política por default

Lámina 9 Dr. Roberto Gómez Cárdenas

TECNOLÓGICO DE MONTERREY

Probando

- Desde Windows hacer un telnet a la máquina linux

```
C:\> telnet w.x.y.z          /* w.x.y.z = direccón IP máquina linux */
```

- Una vez adentro de la máquina salgase

```
$ exit
```

- Cambie la acción de la tabla INPUT por DROP

```
# iptables –P INPUT DROP
```

- Desde Windows hacer un telnet

```
C:\> telnet w.x.y.z          /* w.x.y.z = direccón IP máquina linux */
```

- Volver a cambiar la acción por default de INPUT

```
# iptables –P INPUT ACCEPT
```

Lámina 10 Dr. Roberto Gómez Cárdenas

TECNOLÓGICO DE MONTERREY

Manipulación reglas dentro cadenas

iptables comando [match] [objetivos / saltos]

- **-A** añadir una nueva regla a una cadena
- **-I** Insertar una nueva regla en alguna posición en una cadena
- **-R** Reemplazar una regla en alguna posición en una cadena
- **-D** Anular una regla en alguna posición en una cadena
- **-D** Anular la primera regla que se cumple en una cadena

Lámina 11 Dr. Roberto Gómez Cárdenas

TECNOLÓGICO DE MONTERREY

Ejemplo alta de regla

- Un ping normal y un ping bloqueado

```
# ping -c 1 127.0.0.1
    PING 127.0.0.1 (127.0.0.1): 56 data bytes
    64 bytes from 127.0.0.1: icmp_seq=0 ttl=64 time=0.2 ms

    --- 127.0.0.1 ping statistics ---
    1 packets transmitted, 1 packets received, 0% packet loss
    round-trip min/avg/max = 0.2/0.2/0.2 ms
# iptables -A INPUT -s 127.0.0.1 -p icmp -j DROP
# ping -c 1 127.0.0.1
    PING 127.0.0.1 (127.0.0.1): 56 data bytes

    --- 127.0.0.1 ping statistics ---
    1 packets transmitted, 0 packets received, 100% packet loss
#
```

Lámina 12 Dr. Roberto Gómez Cárdenas

TECNOLÓGICO DE MONTERREY

Ejemplo borrando regla

- Podemos borrar la regla de dos maneras.
- Primero, como es la única regla en la cadena, podemos usar un borrado por número:

```
# iptables -D INPUT 1
#
```

- La segunda manera es repetir la orden –A
  - pero cambiando -A por -D.
  - útil cuando se tiene una compleja cadena de reglas y no se quiere invertir tiempo en saber que número regla es

```
# iptables -D INPUT -s 127.0.0.1 -p icmp -j DROP
#
```

Lámina 13 Dr. Roberto Gómez Cárdenas

TECNOLÓGICO DE MONTERREY

Construcción de una regla

- Una regla se divide en cuatro partes
  - una tabla de aplicación de la cadena
    - filter si no se precisa ninguna
  - una cadena de aplicación
  - un patrón de reconocimiento
  - una acción representando la decisión a tomar

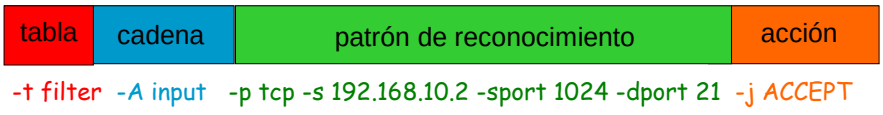

Lámina 14 Dr. Roberto Gómez Cárdenas

El patron de reconocimiento

- Paquete es comparado con un patron de reconocimiento
- Patrón compuesto de un número variable de critérios, de acuerdo la finesa deseada
- Algunas opciones están implementadas directamente en el código de iptables, mientras que otras opciones más complejas necesitan cargar los módulos necesarios en el núcleo para estar disponibles.
- La disponibilidad de estas opciones extendidas depende de la configuración del núcleo

Lámina 15 Dr. Roberto Gómez Cárdenas

Las acciones

- Define que hacer si paquete cumple con regla.
- El objetivo se define con la opción:
  - j <target> [target-options]
    - al final de la regla.
- Hay dos tipos de reglas, terminales y no terminales.
- Si un objetivo es terminal, el paquete no pasará a otras reglas al cumplir con esta.
  - por ejemplo: DROP
- Un ejemplo de regla no terminal es LOG
  - le dice al núcleo de escribir en el registro del sistema (syslog) los paquetes que verifican esta regla.

Lámina 16 Dr. Roberto Gómez Cárdenas

TECNOLÓGICO DE MONTERREY

Acciones más comunes

- ACCEPT
  - dejar el paquete pasar
- DROP
  - tirar el paquete
- QUEUE
  - pasar el paquete a espacio usuario (si esta soportado por el nucleo)
- RETURN
  - regreso a la cadena que activo la llamada
- LOG, ULOG
  - almacenamiento en bitacora

Lámina 17 Dr. Roberto Gómez Cárdenas

TECNOLÓGICO DE MONTERREY

Otras acciones

- DNAT
  - Destination Network Address Translation
- MARK
  - marcar valores que estan asociadaos con paquetes en específico
- MASQUERADE
  - direcciones DHCP, igual que SNAT
- MIRROR
  - invertir campos fuente y destino en el encabezado IP
- REDIRECT
  - redireccionar paquetes y streams a la misma máquina}
- SNAT
  - Source Network Address Translation
- TOS
  - asignar valor del campo Type of Service
- TTL
  - modificar valor el Time To Live en el encabezado IP

Lámina 18 Dr. Roberto Gómez Cárdenas

Criterios patrón reconocimiento

- Las principales criterios de patrón de reconocimiento para formar una regla son
  - **s:** indica un dominio o IP (rango de Ips) de origen sobre el que se evalúa la condición de la regla
  - **d:** igual anterior, solo que sobre dirección destino
  - **i:** interfaz entrada para aplicar regla
  - **o:** interfaz salida sobre la que se aplica la regla
  - **p:** especifica el protocolo del datagrama a analizar valores válidos: tcp, udp o icmp, o un número
  - **f** para tratar con paquetes fragmentados
  - **!** invierte el valor lógico de la condición de regla

Lámina 19 Dr. Roberto Gómez Cárdenas

La IP origen y destino

- La dirección puede ser el nombre de una red, el de un servidor, una dirección de red IP (con la máscara adjunta), o una simple dirección IP.
  - nota: especificar un nombre que tenga que resolverse con una consulta remota a un DNS es una mala idea
- La máscara puede ser una máscara de red o bien un simple número.
  - p.e. la máscara 24 equivale a 255.255.255.
- El comando ! antes de la dirección sirve para invertir el sentido de la dirección.
- El argumento --src es un alias para esta opción.

Lámina 20 Dr. Roberto Gómez Cárdenas

TECNOLÓGICO DE MONTERREY

Ejemplos direcciones

```
iptables -A INPUT --source 12.168.120.15 -j ACCEPT
iptables -A INPUT --src 12.168.120.15 -j ACCEPT
```

La especificación de la direccion destino es similar en uso a la de origen.

```
iptables -A INPUT --destination ! 32.112.0.31/24 -j ACCEPT
iptables -A INPUT --dst ! 32.112.0.31/24 -j ACCEPT
```

Lámina 21 Dr. Roberto Gómez Cárdenas

TECNOLÓGICO DE MONTERREY

El protocolo

- El protocolo puede ser tcp, udp, icmp, o todos
  - puede ser un valor numérico, que represente estos o algún otro protocolo.
- También es válido cualquier nombre de protocolo incluido en /etc/protocol.
- Argumento ! anterior al protocolo invierte la prueba.
- Número cero es equivalente a todos los protocolos.
- El protocolo ¨all¨(todos) es el que se usa por defecto cuando esta opción no aparece.
- Ejemplos:

```
iptables -A INPUT --protocol ! udp -j LOG
iptables -A INPUT --protocol tcp -j ACCEPT
```

Lámina 22 Dr. Roberto Gómez Cárdenas

Verificando campos encabezado paquetes TCP

- sport
  - puerto origen del datagrama, posible especificar rango indicando limites superior e inferior (“:”)
- dport
  - igual que anterior solo que puerto destino
- tcp-flags [!] mascara comp
  - especifica si las bandareas de estan activas
- syn
  - regla verifica que bit SYN valga 1 y los bits ACK y FIN valgan ambos 0, abreviatura de tcp-flags SYN,RST,ACK SYN
- tcp-option [!] <numero>
  - verica si la opcion TCP esta activa
- mss <valor>[:valor]
  - paquetes TCP SYN o SYN/ACK con el valor MSS especificado (o en el rango)

Lámina 23 Dr. Roberto Gómez Cárdenas

Verificando encabezados protocolos UDP e ICMP

- Extensiones UDP
  - **--sport** puerto origen del datagrama
  - **-- dport** puerto destino del datagrama
- Extensiones ICMP
  - **-- icmp-type** especificar tipo mensaje ICMP tanto por su número como por los siguientes indicadores: echo-request, echo-reply, source-quench, time-exceeded, destination-unreachable, network-unreachable, host-unreachable, protocol-unreachable y port-unreachable

Lámina 24 Dr. Roberto Gómez Cárdenas

Verificando puerto origen/destino

```
iptables -A INPUT --protocol tcp --source-port 22:12 -j LOG
iptables -A POSTROUTING --protocol tcp --destination-port ! 21232 -j DROP
```

- El puerto origen-destino o un rango de puertos.
- Se puede poner un nombre de servicio o un número de un puerto.
- También se puede poner un intervalo, poniendo puerto:puerto.
- Si se omite el primer puerto, se asume el valor "0" .
- Si se omite el segundo, se supone el valor 65535.

Lámina 25 Dr. Roberto Gómez Cárdenas

Ejemplo especificación puerto

- Solo se desea que la dirección 10.10.23.17 puede acceder a la página de la compañía
- Dos formas de hacerlo
  - Depende del resto de las reglas en el servidor web
- Negamos todo y solo permitimos el paso a la IP deseada

```
# iptables -P INPUT DROP
# iptables -A INPUT -s 10.10.10.23.17 -p tcp --dport 80 -j ACCEPT
#
```

- Aceptamos todo y no permitimos el paso a la IP deseas

```
# iptables -P INPUT ACCEPT
# iptables -A INPUT -s ! 10.10.10.23.17 -p tcp --dport 80 -j DROP
#
```

Lámina 26 Dr. Roberto Gómez Cárdenas

TECNOLÓGICO DE MONTERREY

¿Qué pasa si no especificamos el protocolo al especificar un puerto?

- Ejemplo anterior:

```
# iptables –A INPUT –s 10.10.10.23.17 --dport 80 –j ACCEPT
iptables v1.3.0: Unknown arg '--dport'
Try 'iptables  -h' or 'iptables --help' for more information
#
```


TECNOLÓGICO DE MONTERREY

IPTables

Roberto Gómez Cárdenas
rogoca@gmail.com
http://www.cryptomex.org