Instituto Politécnico Nacional

Unidad Profesional Interdisciplinaria en Ingeniería y Tecnologías Avanzadas

Seguridad de Redes

Profesor. Polanco Montelongo Francisco Antonio

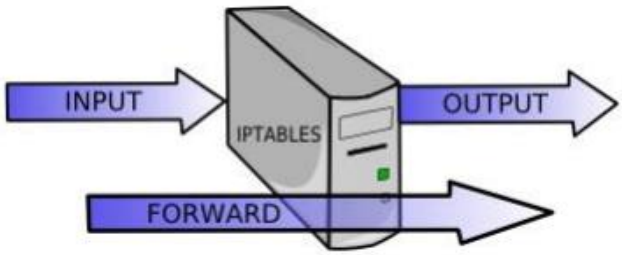
Alumno: Hernández Guzmán Bryan Alexis

Grupo: 3TV4

Práctica: IPtables

Introducción:

Iptables es un módulo del núcleo de Linux que se encarga de filtrar los paquetes de red, es decir, es la parte que se encarga de determinar qué paquetes de datos queremos que lleguen hasta el servidor y cuáles no. Al igual que ocurre con otros sistemas de cortafuegos, iptables funciona a través de reglas. Es decir, el usuario mediante sencillas instrucciones indica al firewall el tipo de paquetes que debe permitir entrar, los puertos por donde se pueden recibir esos paquetes, el protocolo utilizado para el envío de datos y cualquier otra información relacionada con el intercambio de datos entre redes. Cuando en el sistema se recibe o se envía un paquete, se recorren en orden las distintas reglas hasta dar con una que cumpla las condiciones. Una vez localizada, esa regla se activa realizando sobre el paquete la acción indicada. Gracias a su robustez, iptables se ha convertido hoy por hoy en una de las herramientas más utilizadas para el filtrado de tráfico en sistemas Linux.



Desarrollo:

Después de realizar algunos cambios dentro de la configuración de la maquina física, como lo es crear una nueva regla que permite el trafico de datos, a demás de permitir la conexión de impresora dominio al igual que la privada.

Verificamos nuestra dirección ip de nuestras 3 máquinas, la física y 2 virtuales.

Tenemos que meter nuestras reglas dentro de la máquina virtual 1:

```
Chain POSTROUTING (policy ACCEPT)
target     prot opt source               destination

┌──(bryan㉿kali)-[~]
└─$ sudo iptables -F

┌──(bryan㉿kali)-[~]
└─$ sudo iptables -A INPUT -i lo -j ACCEPT

┌──(bryan㉿kali)-[~]
└─$ sudo iptables -A FORWARD -j ACCEPT

┌──(bryan㉿kali)-[~]
└─$ sudo bash -c 'echo "1"> /proc/sys/net/ipv4/ip_forward'

┌──(bryan㉿kali)-[~]
└─$ sudo iptables -t nat -A PREROUTING -p icmp --icmp-type echo-request -d 192.168.1.65 -j DNAT --to-destination 192.1      67

┌──(bryan㉿kali)-[~]
└─$ sudo iptables -t nat -A POSTROUTING -p icmp --icmp-type echo-request -s 192.168.1.68 -j SNAT --to-source 192.168.1

┌──(bryan㉿kali)-[~]
└─$ sudo sysctl -w net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1

┌──(bryan㉿kali)-[~]
└─$ sudo iptables -t nat -L
```

```
Archivo  Acciones  Editar  Vista  Ayuda
└─$ sudo bash -c 'echo "1"> /proc/sys/net/ipv4/ip_forward'

┌──(bryan㉿kali)-[~]
└─$ sudo iptables -t nat -A PREROUTING -p icmp --icmp-type echo-request -d 192.168.1.65 -j DNAT --to-destination 192.16      67

┌──(bryan㉿kali)-[~]
└─$ sudo iptables -t nat -A POSTROUTING -p icmp --icmp-type echo-request -s 192.168.1.68 -j SNAT --to-source 192.168.1.

┌──(bryan㉿kali)-[~]
└─$ sudo sysctl -w net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1

┌──(bryan㉿kali)-[~]
└─$ sudo iptables -t nat -L
Chain PREROUTING (policy ACCEPT)
target     prot opt source               destination
DNAT       icmp --  anywhere             192.168.1.65         icmp echo-request to:192.168.1.67

Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain POSTROUTING (policy ACCEPT)
target     prot opt source               destination
SNAT       icmp --  192.168.1.68         anywhere             icmp echo-request to:192.168.1.65
```

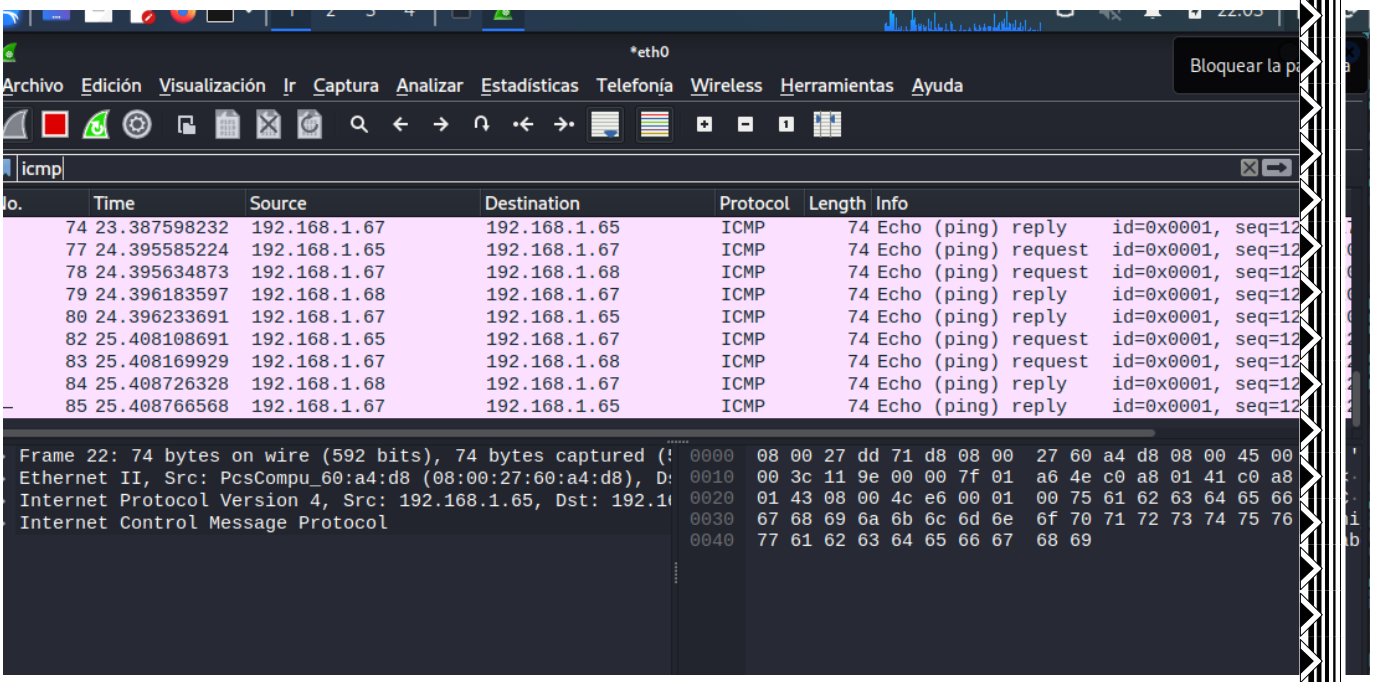
Procedemos a hacer lo mismo dentro de nuestra segunda máquina virtual:

```
┌──(bryan1㉿kali2)-[~]
└─$ sudo iptables -F

┌──(bryan1㉿kali2)-[~]
└─$ sudo sysctl -w net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1

┌──(bryan1㉿kali2)-[~]
└─$ sudo iptables -A INPUT -i lo -j ACCEPT

┌──(bryan1㉿kali2)-[~]
└─$ sudo iptables -A FORWARD -j ACCEPT

┌──(bryan1㉿kali2)-[~]
└─$ sudo bash -c 'echo "1" > /proc/sys/net/ipv4/ip_forward'

┌──(bryan1㉿kali2)-[~]
└─$ sudo iptables -t nat -A PREROUTING -p icmp --icmp-type echo-request -d 192.168.1.67 -j DNAT --to-destination 192.168.1.68

┌──(bryan1㉿kali2)-[~]
└─$ sudo iptables -t nat -A POSTROUTING -p icmp --icmp-type echo-request -s 192.168.1.65 -j SNAT --to-source 192.168.1.67
```

```
┌──(bryan1㉿kali2)-[~]
└─$ sudo iptables -t nat -L
Chain PREROUTING (policy ACCEPT)
target     prot opt source               destination
DNAT       icmp --  anywhere             192.168.1.67         icmp echo-request to:192.168.1.68

Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain POSTROUTING (policy ACCEPT)
target     prot opt source               destination
SNAT       icmp --  192.168.1.65         anywhere             icmp echo-request to:192.168.1.67
```

Procederemos a hacer un ping de nuestra maquina física a virtual y visualizaremos en wireshark, aplicaremos un filtro para icmp:}

```
C:\Users\rockr>ping 192.168.1.65 -n 10

Haciendo ping a 192.168.1.65 con 32 bytes de datos:
Respuesta desde 192.168.1.65: bytes=32 tiempo=1ms TTL=126
Respuesta desde 192.168.1.65: bytes=32 tiempo=2ms TTL=126
Respuesta desde 192.168.1.65: bytes=32 tiempo=2ms TTL=126
Respuesta desde 192.168.1.65: bytes=32 tiempo=2ms TTL=126
Respuesta desde 192.168.1.65: bytes=32 tiempo=1ms TTL=126
Respuesta desde 192.168.1.65: bytes=32 tiempo=2ms TTL=126
Respuesta desde 192.168.1.65: bytes=32 tiempo=1ms TTL=126
Respuesta desde 192.168.1.65: bytes=32 tiempo=2ms TTL=126
Respuesta desde 192.168.1.65: bytes=32 tiempo=1ms TTL=126
Respuesta desde 192.168.1.65: bytes=32 tiempo=1ms TTL=126

Estadísticas de ping para 192.168.1.65:
    Paquetes: enviados = 10, recibidos = 10, perdidos = 0
    (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 1ms, Máximo = 2ms, Media = 1ms
```



Conclusión:

Dentro de esta práctica pudimos observar cómo hacer una conexión entre maquinas físicas y virtuales, la importancia que conlleva eso, en un plano más real que nos motiva a crear cosas nuevas, a demás de la utilización de diversas herramientas para llevar a acabo el cometido final que es entender lo que estamos haciendo, en general en la práctica tuve algunos problemas con las direcciones ip y a al hora de puentear ambas maquinas virtuales, por otro lado investigar los

comandos y para qué sirven fue un plus para dejar atrás errores que fueron cometidos dentro de la práctica.

Referencias:

https://www.acens.com/wp-content/images/2014/07/wp-acens-iptables.pdf