Shorewall: Un buen Cortafuego para sus Servidores GNU/Linux

¿Qué es un Firewall?

Un **Firewall o Cortafuego** es un dispositivo de tipo **Hardware o Software** que nos permite filtrar o gestionar todo el tráfico entrante y saliente que hay que entre dos o más redes. Todo tráfico saliente o entrante es controlado por una serie de **Reglas** para acceder conexión desde la red pública o acceder conexión desde la red privada. En caso de no cumplir con las reglas, el tráfico saliente o entrante es denegado.

Cuando se instala por primera vez alguna distribución de **GNU/Linux** derivados para **Servidores**, tales como: **CentOS, Red Hat, Debian, Ubuntu Server o Fedora**; por lo general en la mayoría de las distros vienen integrado en el kernel de estos sistemas un **firewall de tipo software** llamado **iptables de netfilter**.

Debido a que muchos administradores de sistemas que se adentran al mundo de **GNU/Linux** le dificultan administrar y configurar el firewall con **Iptables**, **Shorewall** es la herramienta ideal para crear y manipular de forma automatizada las reglas, cadenas y módulos de Iptables.

¿Qué es Shorewall (Shoreline Firewall)?

Shorewall o originalmente llamado como **Shoreline Firewall** es una robusta y extensible herramienta de alto nivel para la configuración de muros cortafuego.

SISTEMA DEL SERVIDOR FW

El sistema del **Servidor Firewall** que estamos utilizando para éste tutorial es una distribución llamado **CentOS** **Versión 6.8** de **64 bits** pre-instalado como **Máquina Virtual** con **VMware Workstation**.

- **Nombre del Ordenador:**

```
[root@fwlab1~]# hostname
fwlab1.tecnishn.local
```

- **Información General del Sistema del Servidor FW:**

```
[root@fwlab1~]# cat /etc/centos-release
CentOS release 6.8 (Final)
[root@fwlab1~]# uname -a
Linux fwlab1.tecnishn.local 2.6.32-642.15.1.el6.x86_64 #1 SMP Fri Feb 24
14:31:22 UTC 2017 x86_64 x86_64 x86_64 GNU/Linux
```

IDENTIFICAR LAS DIRECCIONES IP DEL SERVIDOR FW

Lo primero del todo antes de empezar en convertir el servidor en un **Servidor Firewall *(Cortafuego + Enrutador)***, es necesario que el mismo servidor tenga **por lo menos dos interfaces de red *(tarjetas de red pre-instaladas en la placa madre)***. Por lo general, en las interfaces de red de un **Servidor FW en Producción**, deben tener por lo menos **una dirección ip pública en la interfaz para la WAN y una dirección ip privada en la interfaz para la LAN**. Existen muchos casos que en los servidores de cortafuegos tienen por ejemplo: **4 interfaces de red**, es decir, **2 direcciones ip públicas con balanceo de carga para la WAN *(una dirección ip pública por cada interfaz de red)***, **1 dirección ip privada para la LAN en una tercera interfaz de red** y **1 dirección ip privada para la DMZ en una cuarta interfaz de red**.
Tomando en cuenta la **Identificación de las Direcciones Ip del Servidor FW**, nuestro **servidor virtualizado** está utilizando las siguientes direcciones ip:

- **WAN:** Dirección Ip **192.168.2.10** con una máscara de red de **24 bits *(255.255.255.0)*** en la interfaz de red **eth0**.
```
[root@fwlab1 ~]# ifconfig eth0
eth0 Link encap:Ethernet HWaddr 00:50:56:2B:ED:CD
inet addr:192.168.2.10 Bcast:192.168.2.255 Mask:255.255.255.0
```
- **LAN:** Dirección Ip **172.16.20.1** con una máscara de red de **28 bits *(255.255.255.240)*** en la interfaz de red **eth1**.
```
[root@fwlab1 ~]# ifconfig eth1
eth1 Link encap:Ethernet HWaddr 00:50:56:29:41:79
inet addr:172.16.20.1 Bcast:172.16.20.15 Mask:255.255.255.240
```
- **DMZ:** Dirección Ip **10.16.20.1** con una máscara de red de **29 bits *(255.255.255.248)*** en la interfaz de red **eth2**.
```
[root@fwlab1 ~]# ifconfig eth2
eth2 Link encap:Ethernet HWaddr 00:50:56:3A:0B:DA
inet addr:10.16.20.1 Bcast:10.16.20.7 Mask:255.255.255.248
```

Además de identificar las direcciones ip, también es bueno como referencia extra identificar y conocer las **Subredes** de cada interfaz de red. Para esto podemos utilizar el comando ***"netstat -r"*** o el comando ***"ip route"***.

- Identificar y conocer la subredes de cada interfaz de red con el comando ***"netstat -r"***.
```
[root@fwlab1 ~]# netstat -r
Kernel IP routing table
Destination Gateway Genmask Flags MSS Window irtt Iface
10.16.20.0 * 255.255.255.248 U 0 0 0 eth2
172.16.20.0 * 255.255.255.240 U 0 0 0 eth1
192.168.2.0 * 255.255.255.0 U 0 0 0 eth0
```
- Identificar y conocer las subredes de cada interfaz de red con el comando ***"ip route"***.
```
[root@fwlab1 ~]# ip route
10.16.20.0/29 dev eth2 proto kernel scope link src 10.16.20.1
172.16.20.0/28 dev eth1 proto kernel scope link src 172.16.20.1
192.168.2.0/24 dev eth0 proto kernel scope link src 192.168.2.10
```

Como podrán observar, con ambos comandos lo que se muestra es una **Tabla de Enrutamiento** para cada interfaz de red ***("eth0". "eth1". "eth2")***, en pocas palabras, cada interfaz de red es un **Nodo de Red** diferente en donde los **paquetes de datos pueden ser encaminados en cada una de los nodos**. Por lo tanto, para que un **Servido FW** pueda

enrutar paquetes de datos entre diferentes nodos, **es necesario habilitar el enrutamiento de datos en el servidor**.

HABILITAR EL ENRUTAMIENTO (ROUTER) EN EL SERVIDOR FW

Para que un **Servidor Cortafuego** tenga la funcionalidad de un **Enrutador o Router**, es sumamente importante habilitar esta función para facilitar al **Shorewall** enrutar los paquetes de datos a través del servidor por medio de dos o más **interfaces de red *(tarjetas de red)***.
Para habilitar el enrutamiento de paquetes de datos para el **Protocolo IPV4** en el servidor, es necesario editar el archivo ***"ip_forward"*** de la siguiente manera:

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

Ahora, la única inconveniencia es que después de haberse realizado este paso, es que si por alguna razón el sistema del servidor debe reiniciarse, la habilidad del servidor como enrutador se perderá y para que el enrutamiento sea habilitado nuevamente, hay que volver a ejecutar el comando mostrado anteriormente. Entonces para que esto no suceda, hay que editar un archivo llamado **sysctl.conf** cambiando el valor ***"0"*** del parámetro ***"net.ipv4.ip_forward"*** al valor ***"1"***. Hay que utilizar un **editor de texto** ***(por ejemplo: vi, vim, nano, gedit, etc.)*** para realizar este paso de configuración.

```
[root@fwlab1 ~]# ls -l /etc/sysctl.conf
-rw-r--r--. 1 root root 1057 Feb 24 10:05 /etc/sysctl.conf
[root@fwlab1 ~]# cat /etc/sysctl.conf | grep ip_forward
net.ipv4.ip_forward = 0 # El valor por defecto siempre está en 0.
[root@fwlab1 ~]# vim /etc/sysctl.conf
```

Una vez abierto el archivo editable ***"sysctl.conf"***, simplemente cambiamos el valor ***"0"*** por el valor ***"1"***.

```
net.ipv4.ip_forward = 1 # Antes el valor era 0 y ahora su valor es 1.
```

Una vez que hemos hecho todos estos cambios, ya no habrá motivos para preocuparse en perder la capacidad de enrutar paquetes de datos en el servidor por si algún motivo el sistema del mismo se reinicia o por si hay que reiniciarlo. Por lo general en un servidor en producción con **GNU/Linux** es bien raro en que se debe reiniciarse cada vez que se hace un cambio de configuración.

DETENER Y DESHABILITAR IPTABLES EN EL SERVIDOR FW

Tal como hemos mencionado en la introducción de este tutorial, cuando se instala desde cero alguna distribución de **GNU/Linux** para servidores, el cortafuego llamado **Iptables**, **siempre vendrá integrado y especialmente habilitado por defecto** en el kernel de estos sistemas. Es por esta razón que **forzosamente hay que detener y deshabilitar iptables** solamente si se requiere instalar y utilizar **Shorewall** como medida de seguridad informática en el servidor.
Hay que tomar en cuenta 2 puntos sumamente importantes con respecto a **Iptables** cuando el sistema de cualquier distribución de **GNU/Linux** está recién instalado en el servidor:

1. Iptables por defecto vienen disponibles para los **Protocolos de Internet IPv4** y **IPv6**.
2. Al detener y deshabilitar Iptables para ambos **Protocolos de Internet**, el servidor estará **100% vulnerable ante cualquier ataque cibernético o conexión no**

autorizado con el servidor. Por ende, al instalar Shorewall, **hay que configurarlo sin cometer errores y también lo más rápido posible**.

Por consiguiente, vamos a detener y deshabilitar Iptables para luego proceder con la instalación y configuración del Shorewall.

DETENER IPTABLES

Es necesario detener Iptables para ambos Protocos de Internet explicados anteriormente, entonces para cumplir tal objetivo, hay que seguir las siguientes instrucciones:

- Para detener Iptables con el **Protocolo de Internet IPv4**, hay que ejecutar cualquiera de los siguientes comandos: ***"/etc/rc.d/init.d/iptables stop"***, ***"/etc/init.d/iptables stop"*** o ***"service iptables stop"***.

```
[root@fwlab1 ~]# /etc/rc.d/init.d/iptables stop
iptables: Setting chains to policy ACCEPT: filter [ OK ]
iptables: Flushing firewall rules: [ OK ]
iptables: Unloading modules: [ OK ]
```

- Para detener Iptables con el **Protocolo de Internet IPv6**, hay que ejecutar cualquiera de los siguientes comandos: ***"/etc/rc.d/init.d/ip6tables stop"***, ***"/etc/init.d/ip6tables stop"*** o ***"service ip6tables stop"***.

```
[root@fwlab1 ~]# /etc/rc.d/init.d/ip6tables stop
ip6tables: Setting chains to policy ACCEPT: filter [ OK ]
ip6tables: Flushing firewall rules: [ OK ]
ip6tables: Unloading modules: [ OK ]
```

DESHABILITAR IPTABLES

Ahora que Iptables está detenido para ambos Protocos de Internet, el siguiente paso es deshabilitarlo y para cumplir tal objetivo, hay que seguir las siguientes instrucciones:

- Para deshabilitar Iptables con el **Protocolo de Internet IPv4**, hay que ejecutar cualquiera el siguiente comando: ***"chkconfig iptables off"***.

```
[root@fwlab1 ~]# chkconfig iptables off
[root@fwlab1 ~]# chkconfig --list | grep iptables
iptables 0:off 1:off 2:off 3:off 4:off 5:off 6:off
```

- Para deshabilitar Iptables con el **Protocolo de Internet IPv6**, hay que ejecutar cualquiera el siguiente comando: ***"chkconfig ip6tables off"***.

```
[root@fwlab1 ~]# chkconfig ip6tables off
[root@fwlab1 ~]# chkconfig --list | grep ip6tables
ip6tables 0:off 1:off 2:off 3:off 4:off 5:off 6:off
```

Un punto importante que debemos tomar en cuenta cuando **deshabilitamos Iptables o cualquier otro servicio ejecutable** en el servidor con el comando ***"chkconfig"*** es una vez que esté deshabilitado, significa que al reiniciar el sistema del servidor o al encenderlo, el **servicio ejecutable *(en este caso: Iptables)*** no volverá ser inicializado por el kernel del sistema.