Contenido

[Configuración de firewall 1](#_Toc43132489)

[Configuración del servidor web 2](#_Toc43132490)

[Configuración del servidor DHCP 3](#_Toc43132491)

[Configuración RAID y LVM 4](#_Toc43132492)

## Configuración de firewall

Esta máquina virtual consta de 3 placas de red las cuales conectan las redes LAN 192.168.10.0/24 y 192.168.20.0/24, y la conexión WAN.

Se habilita el reenvió de paquetes IPv4 editando el archivo /etc/sysctl.conf y se descomenta la siguiente línea.



Para que el firewall cargue la configuración al inicio se debe crear un directorio donde guardar el archivo que contendrá la configuración y posteriormente mediante un script se cargara dicha configuración cuando levanten los servicios de networking. Para ello realizaremos los siguientes pasos.

1. Creamos un directorio:

Mkdir /home/iptables

1. Guardamos el archivo de configuración de iptables:

Iptables-save > /home/iptables/rules.ipv4.conf

1. Creamos el script en el siguiente directorio y lo editamos:

Touch /etc/network/if-up.d/iptables

Nano /etc/network/if-up.d/iptables

1. Agregamos las siguientes líneas en el script:

#!/bin/sh

iptables-restore < /home/iptables/iptables.v4

1. Le damos permisos de ejecución al script creado:

Chmod +x iptables.v4

Hay que tener en cuenta que para cada cambio que realicemos en el iptables y que queramos que se mantenga con cada reinicio de la máquina virtual es necesario ejecutar el comando del apartado 2° : Iptables-save > /home/iptables/rules.ipv4.conf

* Se setean las políticas de las 3 cadenas para que por defecto dropeen el trafico:

Iptables –P INPUT DROP

Iptables –P OUTPUT DROP

Iptables –P FORWARD DROP

* Se habilita el tráfico desde/hacia la interfaz loopback:

Iptables -A INPUT -i lo -j ACCEPT

Iptables -A OUTPUT -o lo -j ACCEPT

* Para permitir que la única VM que pueda administrar el firewall vía ssh (puerto tcp 22) sea cliente-02:

Iptables -A INPUT -s 192.168.20.2/32 -d 192.168.20.1/32 -i eth2 -p tcp -m tcp --dport 22 -j ACCEPT

Iptables -A OUTPUT -s 192.168.20.1/32 -d 192.168.20.2/32 -o eth2 -p tcp -m tcp --sport 22 -j ACCEPT

* La única VM que pueda navegar por internet sea cliente-03:

Iptables -A FORWARD -s 192.168.20.3/32 -o eth0 -j ACCEPT

Iptables -A FORWARD -d 192.168.20.3/32 -o eth2 -j ACCEPT

Iptables -A POSTROUTING -s 192.168.20.0/24 -o eth0 -j MASQUERADE

* La única VM de la red 192.168.20.0/24 que pueda ingresar al web server de la red 10.0 sea cliente-04:

Iptables -A FORWARD -s 192.168.20.4/32 -d 192.168.10.3/32 -o eth1 -p tcp -m tcp --dport 8080 -j ACCEPT

Iptables -A FORWARD -s 192.168.10.3/32 -d 192.168.20.4/32 -o eth2 -p tcp -m tcp --sport 8080 -j ACCEPT

## Configuración del servidor web

Para correr el servicio de tomcat deberemos instalar la Java Runtime Machine, para ello deberemos obtener el archivo tar (en este caso se instaló la versión jdk-8u202-linux-x64.tar.gz ) y descomprimirlo en el directorio /opt con el siguiente comando:

* tar xvf (directorio donde guardamos el tar)/jdk-8u202-linux-x64.tar.gz -C /opt/

Luego debemos definir las variables de entorno para que el servicio tomcat se pueda ejecutar, para ello editamos el archivo .bashrc y agregamos las siguientes líneas al final del archivo:

JAVA\_HOME=/opt/jdk1.8.0\_202/

export PATH=$PATH:$JAVA\_HOME/bin

Posteriormente descomprimiremos el tomcat el cual previamente deberemos de haberlo descargado o copiado mediante ftp o scp desde algún otro pc. Para descomprimirlo usaremos nuevamente el comando tar como se hizo con java:

* tar xfv (directorio donde guardamos el tar)/apache-tomcat-8.5.54.tar.gz -C /opt/

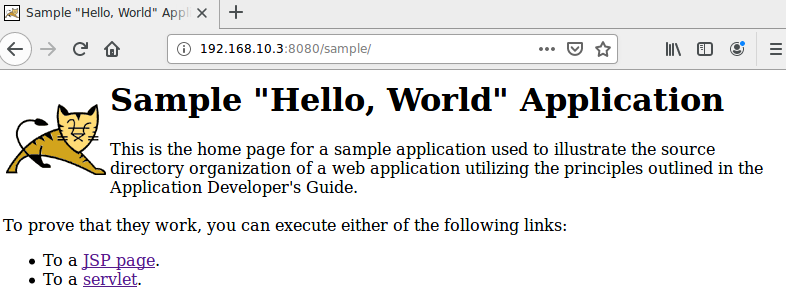
Procedemos a descargar la aplicación de prueba y la copiamos al directorio del tomcat:

* (Directorio donde descargamos el archivo)/sample.war/opt/apache-tomcat-8.5.54/webapps/

Por ultimo procedemos a ejecutar el servicio de tomcat:

* /opt/apache-tomcat-8.5.54/bin/startup.sh

Y probamos desde un navegador que podamos acceder a la url <http://192.168.10.3:8080/sample> donde deberemos ver la página de prueba.



## Configuración del servidor DHCP

Instalamos el servicio DHCP realizando un update de los repositorios y luego la descarga e instalación de dicho servicio.

* apt-get update
* apt-get install isc-dhcp-server

una vez instalado deberemos habilitar la interfaz por la cual recibirá y responderá a las solicitudes dhcp, para ello debemos editar el siguiente archivo:

* /etc/default/isc-dhcp-server

Y modificaremos la última línea donde especificaremos la interfaz por la cual escuchara las solicitudes dhcp.

INTERFACES="eth0"

Configuramos las opciones del scope del servidor dhcp:

subnet 192.168.20.0 netmask 255.255.255.0 {

range 192.168.20.101 192.168.20.110;

option domain-name "TEST.LOCAL";

option domain-name-servers 8.8.8.8;

option routers 192.168.20.1;

default-lease-time 600;

max-lease-time 7200;

}

Por ultimo reiniciamos el servicio dhcp.

* /etc/init.d/isc-dhcp-server restart

## Configuración RAID y LVM

Para este caso se configuro el host Cliente-02 con 6 discos, durante la instalación se utilizaron 2 discos para armar un RAID 1 por software.

Los 4 discos restantes se utilizaron en RAID 10 los cuales se utilizaron para implementar LVM. A continuación, se detallan los pasos para armar el raid 10 para luego utilizarlos para la implementación de LVM.

*Implementación del RAID 10*

1. Se particionan los discos que formaran parte del raid con la herramienta fdisk en formato linux raid autodetect(fd).

* fdisk /dev/sdc1 ; fdisk /dev/sdd1 ; fdisk /dev/sde1 ; fdisk /dev/sdf1

1. Se crea el raid en este caso 10:

* mdadm --create /dev/md10 --level=10 --raid-devices=**4** /dev/sd**c**1 /dev/sd**d**1 /dev/sd**e**1 /dev/sd**f**1

1. Se particiona el nuevo disco md10 con fdisk.

* fdisk /dev/md10

1. Se formatea el disco con el formato de fileSystem que se requiera con mkfs en este caso se formatea en ext4.

* mkfs.ext4 /dev/md10p1

1. Se crea un directorio para que se monte la partición nueva.

* Mkdir /dir/raid

1. Se agrega el nuevo disco al archivo de configuración de mdadm para que este lo detecte al inicio

* Mdadm --detail --scan >> /etc/mdadm/mdadm.conf

1. Se obtiene el uuid del disco mediante la utilidad “blkid” y se lo agrega para que se automonte al arranque del sistema operativo en el fstab.

*Implementación de LVM*

Para la implementación de LVM lo primero que debemos hacer es crear los physical volumes, para ello usaremos el siguiente comando:

* pvcreate /dev/md10

Creamos los volume groups:

* vgcreate lvg01 /dev/md10

Creamos el logical volumen, en este caso con un tamaño de 5Gb:

* lvcreate -L 5g -n archives lvg01

Formateamos el nuevo logical volumen:

* mkfs.ext4 /dev/lvg01/archives

Luego crearemos un directorio donde montar el nuevo logical volumen:

* mkdir /home/archives

Por ultimo obtenemos el UUID mediante el blkid y lo agregamos al archivo fstab para que se monte automáticamente al arranque del SO:

Output del comando blkid:



Agregamos el UUID al fstab:

