



**Facultad de Ingeniería**

---

## **Trabajo Práctico**

**Materia: Computación Aplicada**

**Carrera: Lic. en Redes y Comunicaciones de Datos  
Turno: Noche**

**Docente: Ignacio Sánchez**

**Alumno: Facundo Romano**

**Legajo: 0111622**

## Contenido

Configuración Firewall.....	3
Instalación JAVA y TOMCAT .....	5
Configuración de servidor DHCP .....	6
Configuración LVM .....	6
Scripting.....	8

## Configuración Firewall

Para la configuración del firewall, se utilizó una VM con la distribución Linux Debian 8 sin entorno gráfico. La misma tiene 3 interfaces de red:

1. Conectada a Internet
2. Conectada a la red 192.168.10.0/24
3. Conectada a la red 192.168.20.0/24

Red

Adaptador 1 Adaptador 2 Adaptador 3 Adaptador 4

☒ Habilitar adaptador de red

Conectado a: Adaptador puente

Nombre: Intel(R) Ethernet Connection (4) I219-LM

Avanzadas

Red

Adaptador 1 Adaptador 2 Adaptador 3 Adaptador 4

☒ Habilitar adaptador de red

Conectado a: Red interna

Nombre: 192.168.10.0/24

Avanzadas

Red

Adaptador 1 Adaptador 2 Adaptador 3 Adaptador 4

☒ Habilitar adaptador de red

Conectado a: Red interna

Nombre: 192.168.20.0/24

Avanzadas

Estas interfaces fueron configuradas de forma estática, salvo la eth0 que toma automáticamente del dhcp interno del router hogareño. El archivo de configuración se encuentra en: **/etc/network/interfaces**

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# interfaz hacia INTERNET
auto eth0
iface eth0 inet dhcp

#interfaz hacia RED 192.168.10.0/24
auto eth1
iface eth1 inet static
    address 192.168.10.1
    netmask 255.255.255.0
    broadcast 192.168.10.255
    network 192.168.10.0

#interfaz hacia RED 192.168.20.0/24
auto eth2
iface eth2 inet static
    address 192.168.20.1
    netmask 255.255.255.0
    broadcast 192.168.20.255
    network 192.168.20.0
```

Para aplicar las restricciones que pedía el trabajo practico en el firewall, se utilizaron las siguientes reglas de iptables:

```
root@firewall:/etc/iptables# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source                destination
ACCEPT     all  --  anywhere              anywhere
ACCEPT     tcp  --  192.168.20.2          anywhere            tcp dpt:ssh

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination
DROP       all  --  192.168.20.3          192.168.10.3
ACCEPT     tcp  --  192.168.10.3          192.168.20.4        tcp dpts:1024:65535
ACCEPT     tcp  --  192.168.20.4          anywhere            tcp dpt:http
ACCEPT     tcp  --  192.168.20.4          anywhere            tcp dpt:http-alt
ACCEPT     all  --  192.168.20.3          anywhere
ACCEPT     all  --  anywhere              192.168.20.3

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
ACCEPT     all  --  anywhere              anywhere
ACCEPT     all  --  192.168.20.1          192.168.20.2
root@firewall:/etc/iptables# _
```

También se configuraron las reglas NAT que se pedían:

```
root@firewall:/etc/iptables# iptables -t nat -L
Chain PREROUTING (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

Chain POSTROUTING (policy ACCEPT)
target     prot opt source                destination
MASQUERADE all  --  192.168.10.0/24        anywhere
MASQUERADE all  --  192.168.20.0/24        anywhere
```

Para que estas configuraciones de iptables queden definitivamente en la VM Firewall, se guardó en el archivo: **/etc/iptables/rules.ipv4.rules**

Y para hacer que las reglas se carguen automáticamente al encender la VM, se creó el siguiente script, ubicado en el path: **/etc/network/if-up.d/iptables**

```
#!/bin/sh
iptables-restore < /etc/iptables/rules.v4.conf
```

## Instalación JAVA y TOMCAT

El trabajo práctico solicitaba crear una VM que cumpla la función de Web-Server. Para esto se utilizó una VM con distribución Debian 8 sin entorno gráfico.

Se instaló JAVA y TOMCAT en la VM. Se transfirieron los archivos .tar.gz a través de WinSCP en el directorio **/opt**

```
root@web-server:/opt# ls -la
total 199588
drwxr-xr-x  4 root root    4096 jun  4 17:31 .
drwxr-xr-x 22 root root    4096 may 27 19:09 ..
drwxr-xr-x  9 root root    4096 jun  4 17:19 apache-tomcat-8.5.54
-rw-r--r--  1 root root 10312541 jun  4 16:42 apache-tomcat-8.5.54.tar.gz
drwxr-xr-x  7 uucp 143     4096 dic 15 2018jdk1.8.0_202
-rw-r--r--  1 root root 194042837 jun  4 16:40jdk-8u202-linux-x64.tar.gz
root@web-server:/opt#
```

Se descomprimieron e instalaron los archivos con el comando **tar xzvf archivo.tar.gz** en el mismo directorio como se ve en la imagen. Las versiones que se utilizaron son:

1. Java 1.8.0\_202
2. Tomcat 8.5.54

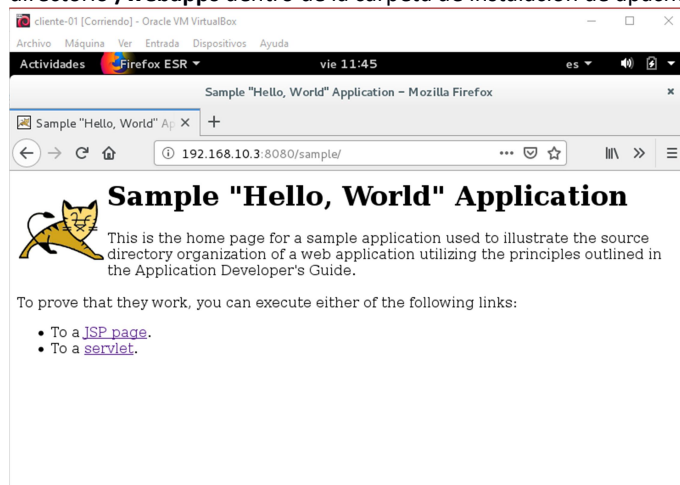
Para que la aplicación JAVA inicie automáticamente al encender la VM, se agregó al archivo **/root/.bashrc** las siguientes líneas.

```
#JDK
JAVA_HOME=/opt/jdk1.8.0_202
export PATH=$JAVA_HOME/bin:$PATH
```

Para iniciar el TOMCAT se ejecuta el script manualmente startup.sh en el directorio **/opt/apache-tomcat-8.5.54/bin**.

```
root@web-server:/opt/apache-tomcat-8.5.54/bin# ./startup.sh
Using CATALINA_BASE:   /opt/apache-tomcat-8.5.54
Using CATALINA_HOME:   /opt/apache-tomcat-8.5.54
Using CATALINA_TMPDIR: /opt/apache-tomcat-8.5.54/temp
Using JRE_HOME:        /opt/jdk1.8.0_202
Using CLASSPATH:       /opt/apache-tomcat-8.5.54/bin/bootstrap.jar:/opt/apache-tomcat-8.5.54/bin/tomcat-juli.jar
Tomcat started.
root@web-server:/opt/apache-tomcat-8.5.54/bin#
```

Una vez levantados los servicios de Java y Tomcat se copió la aplicación de ejemplo "simple.war" al directorio **/webapps** dentro de la carpeta de instalación de apache y se verifico el funcionamiento:



## Configuración de servidor DHCP

Para la configuración del servidor DHCP, se instaló una VM con la distribución Linux Debian 8 sin interfaz gráfica.

Como primera medida, se instaló el ISC DHCP server con el comando **apt-get instal isc-dhcp-server** y se modificó el archivo **/etc/default/isc-dhcp-server** para habilitar la interfaz eth0 entregue IPs.

```
# On what interfaces should the DHCP server (dhcpd) serve DHCP requests?
#       Separate multiple interfaces with spaces, e.g. "eth0 eth1".
INTERFACES="eth0"
```

Luego se configuro en el archivo **/etc/dhcp/dhcp.conf** el pool de IPs que debe entregar.

```
#Subnet RED20.LAN
subnet 192.168.20.0 netmask 255.255.255.0 {
    range 192.168.20.101 192.168.20.110;
    option routers 192.168.20.1;
    option broadcast-address 192.168.20.255;
}
```

Se corroboró que entregue IPs correctamente.

```
root@cliente-06:/home/fromano# ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:e6:a7:21
          inet addr:192.168.20.101  Bcast:192.168.20.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fee6:a721/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:6 errors:0 dropped:0 overruns:0 frame:0
          TX packets:106 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:926 (926.0 B)  TX bytes:13395 (13.0 KiB)
```

## Configuración LVM

Para aplicar la configuración de LVM, se utilizó una VM con distribución Linux Debian 8 sin entorno gráfico. El mismo cuenta con 1 solo disco de 16GB y configurado cuando se instaló el Linux.

Se creó 1 solo volumen físico con 3 volúmenes lógicos, separando los directorios **/home**, **/tmp**, **/var-log**.

Esto compone el Volume-Group LVM-UP.

```
--- Volume group ---
VG Name                LVM-UP
System ID
Format                 lvm2
Metadata Areas         1
Metadata Sequence No   6
VG Access               read/write
VG Status               resizable
MAX LV                 0
Cur LV                 3
Open LV                3
Max PV                 0
Cur PV                1
Act PV                 1
VG Size                13,02 GiB
PE Size                4,00 MiB
Total PE               3332
Alloc PE / Size        3332 / 13,02 GiB
Free PE / Size         0 / 0
VG UUID                HoY4u5-IQvD-wtGm-jVjL-CwQf-99rU-FW3u1a

--- Logical volume ---
LV Path                /dev/LVM-UP/tmp
LV Name                tmp
VG Name                LVM-UP
LV UUID                Fz2ypI-Q0EH-oN20-dXe6-V8Qr-wBBt-HJZhLO
LV Write Access         read/write
LV Creation host, time cliente-02, 2020-06-03 16:24:40 -0300
LV Status                available
# open                  1
LV Size                 1,86 GiB
Current LE              476
Segments                1
Allocation              inherit
Read ahead sectors      auto
- currently set to     256

--- Logical volume ---
LV Path                /dev/LVM-UP/var-log
LV Name                var-log
VG Name                LVM-UP
LV UUID                QnTWQa-w54d-NCrd-5QdK-mtuF-PL7q-xtmvOB
LV Write Access         read/write
LV Creation host, time cliente-02, 2020-06-03 16:25:05 -0300
LV Status                available
# open                  1
LV Size                 2,79 GiB
Current LE              715
Segments                1
Allocation              inherit
Read ahead sectors      auto
- currently set to     256
Block device            254:1

--- Logical volume ---
LV Path                /dev/LVM-UP/home
LV Name                home
VG Name                LVM-UP
LV UUID                evurtr-1Vy4-XKUN-mEUz-WXRy-0HW3-ffQrec
LV Write Access         read/write
LV Creation host, time cliente-02, 2020-06-03 16:25:46 -0300
LV Status                available
# open                  1
LV Size                 8,36 GiB
Current LE              2141
Segments                1
Allocation              inherit
Read ahead sectors      auto
- currently set to     256
Block device            254:2

--- Logical volume ---
LV Path                /dev/LVM-UP/home
LV Name                home
VG Name                LVM-UP
LV UUID                evurtr-1Vy4-XKUN-mEUz-WXRy-0HW3-ffQrec
LV Write Access         read/write
LV Creation host, time cliente-02, 2020-06-03 16:25:46 -0300
LV Status                available
# open                  1
LV Size                 8,36 GiB
Current LE              2141
Segments                1
Allocation              inherit
Read ahead sectors      auto
- currently set to     256
Block device            254:2

--- Physical volumes ---
PV Name                /dev/sda7
PV UUID                QVeWva-s7cY-HVxI-Yt9Y-I3JN-decp-ZimU3M
PV Status               allocatable
Total PE / Free PE     3332 / 0
```

```
root@cliente-02:~# pvdisplay -v
DEGRADED MODE. Incomplete RAID LVs will be processed.
Scanning for physical volume names
--- Physical volume ---
PV Name                /dev/sda7
VG Name                LVM-UP
PV Size                13,02 GiB / not usable 2,00 MiB
Allocatable            yes (but full)
PE Size                4,00 MiB
Total PE               3332
Free PE                0
Allocated PE           3332
PV UUID                QVeWva-s7cY-HVxI-Yt9Y-I3JN-decp-ZimU3M
```

```
root@cliente-02:~# fdisk -l

Disco /dev/sda: 16 GiB, 17179869184 bytes, 33554432 sectores
Unidades: sectores de 1 * 512 = 512 bytes
Tamaño de sector (lógico/físico): 512 bytes / 512 bytes
Tamaño de E/S (mínimo/óptimo): 512 bytes / 512 bytes
Tipo de etiqueta de disco: dos
Identificador del disco: 0x268d3293

Device Boot      Start         End      Sectors  Size Id Type
/dev/sda1 *        2048       391167       389120    190M 83 Linux
/dev/sda2          393214   33552383   33159170   15,8G  5 Extended
/dev/sda5          393216   2344959    1951744    953M 82 Linux swap / Solaris
/dev/sda6         2347008   6250495    3903488    1,9G 83 Linux
/dev/sda7          6252544   33552383   27299840    13G 8e Linux LVM

Disco /dev/mapper/LVM--UP-tmp: 1,9 GiB, 1996488704 bytes, 3899392 sectores
Unidades: sectores de 1 * 512 = 512 bytes
Tamaño de sector (lógico/físico): 512 bytes / 512 bytes
Tamaño de E/S (mínimo/óptimo): 512 bytes / 512 bytes
Disco /dev/mapper/LVM--UP-var--log: 2,8 GiB, 2998927360 bytes, 5857280 sectores
Unidades: sectores de 1 * 512 = 512 bytes
Tamaño de sector (lógico/físico): 512 bytes / 512 bytes
Tamaño de E/S (mínimo/óptimo): 512 bytes / 512 bytes
Disco /dev/mapper/LVM--UP-home: 8,4 GiB, 8980004864 bytes, 17539072 sectores
Unidades: sectores de 1 * 512 = 512 bytes
Tamaño de sector (lógico/físico): 512 bytes / 512 bytes
Tamaño de E/S (mínimo/óptimo): 512 bytes / 512 bytes
root@cliente-02:~# _
```

## Scripting

Para el ejercicio de scripting, se creó el archivo **scripting-tp-integrador.sh** donde contiene toda la programación del ejercicio solicitado, aplicando el concepto de funciones y un menú para elegir que función ejecutar. El mismo se creó y ejecuto en la VM del firewall.

```
MENU
1. Pedir un numero entero y mostrar esa cantidad de elementos de la sucesion de Fibonacci
2. Pedir un numero y mostrar por pantalla ese numero en forma invertida
3. Pedir una cadena de caracteres y evaluar si es palindromo o no
4. Pedir el path a un archivo de texto y mostrar por pantalla la cantidad de lineas que tiene
5. Pedir el ingreso de 5 numeros enteros y mostrarlos por pantalla en forma ordenada
6. Pedir el path a un directorio y mostrar por pantalla cuantos archivos de cada tipo contiene
7. SALIR
Seleccione una opcion:
```