

TUTORIAL INSTALACIÓN DEBIAN SERVER Y CONFIGURACIÓN COMO SERVIDOR WEB

PARTE 1: Instalación de Debian 10 Server.

Configuración Máquina Virtual:

1024 MB RAM

25GB Disco Duro

3 Adaptadores Red:

-NAT (enp0s3)

-Anfitrión (enp0s8)

-Red Interna (enp0s9)

Imagen: https://drive.google.com/file/d/1YE_wipmvtIueJt-r4P-2Y_caGN0fII2a/view

Empezamos la instalación:

- Iniciamos la instalación y seleccionamos la opción de hacerlo mediante interfaz gráfica
- Seleccionamos idioma y localización
- Nos pide seleccionar interfaz de red primaria, elegimos la NAT
- Ponemos nombre a nuestro server
- Dejamos vacío el dominio y posteriormente lo mismo con el superusuario
- Añadimos nuestro nombre de usuario y contraseña
- Método de particionado: guiado utilizar todo el disco y opción de novatos
- Sí queremos escribir los cambios en los discos
- Esperamos mientras se instala el sistema base
- Sí, gestor de paquetes
- España y la réplica que nos marque por defecto
- La configuración de Proxy la dejamos en blanco
- NO queremos participar en la encuesta de paquetes

Como lo que vamos a instalar un servidor no queremos consumir muchos recursos, así pues en la siguiente pantalla de la instalación:

- Desmarcaremos la opción de Interfaz Gráfica y el Xfce (entorno de escritorio ligero para S.O. Tipo Unix). Adicionalmente podremos instalar el servidor SSH y el Servidor Web (Apache). Si no lo instalamos ahora, podremos hacerlo mas adelante una vez instalado el Debian.
- También instalaremos el servidor de impresión y utilidades estándar.
- Sí al arranque GRUB, y **MUY IMPORTANTE** seleccionamos nuestro disco
- Con esto finalizamos la instalación y pasaremos a la fase de configuración del Debian Server 10

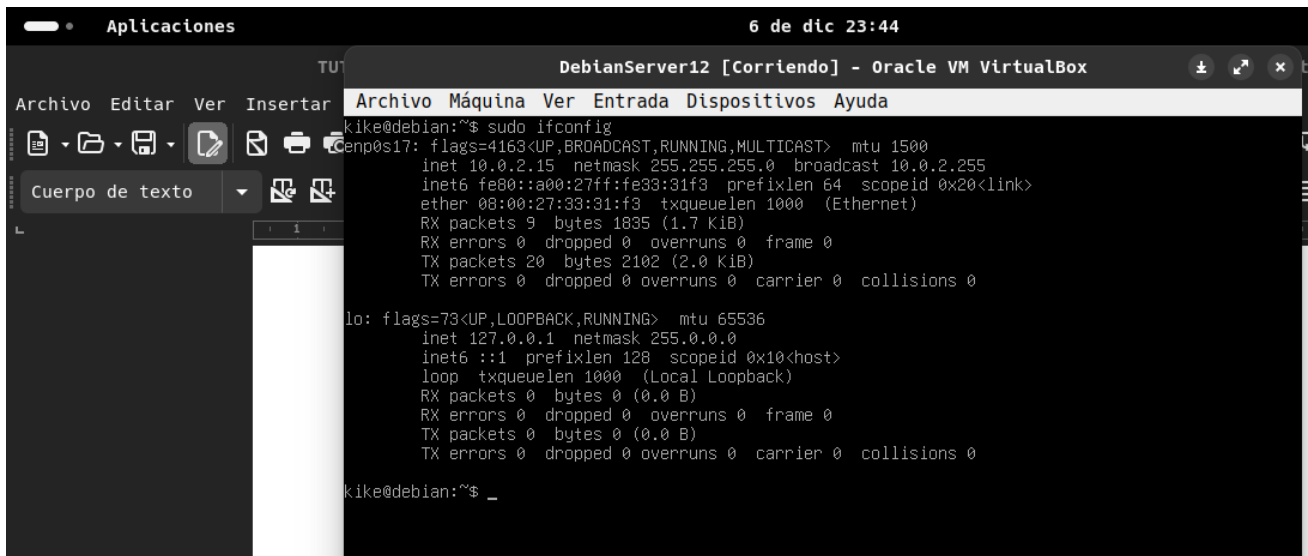
Empezamos la configuración de red:

Lo siguiente que debemos hacer será configurar las interfaces de red de nuestro Debian.

Para ello lo primero será la instalación de las net-tools

\$sudo apt install net-tools

Una vez instalado miraremos que configuración de red tenemos
\$sudo ifconfig



The screenshot shows a VirtualBox window titled "DebianServer12 [Corriendo] - Oracle VM VirtualBox". Inside the window is a terminal window with the following output:

```
kike@debian:~$ sudo ifconfig
enp0s17: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::a00:27ff:fe93:31f3 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:33:31:f3 txqueuelen 1000 (Ethernet)
    RX packets 9 bytes 1835 (1.7 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 20 bytes 2102 (2.0 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

kike@debian:~$ _
```

Podemos observar que solo tenemos configurada una interfaz de red y el loop, así pues tenemos dos

interfaces por configurar: enp0s8 y enp0s9 (en caso de que la NAT sea la enp0s3).

Para configurar las interfaces de red deberemos modificar el archivo de nombre interfaces pero antes de hacerlo se recomienda realizar una copia de seguridad del mismo por si cometemos algún error poder restaurar el mismo.

```
$sudo cp /etc/network/interfaces /etc/network/interfaces_copy
```

Ahora procedemos a editar interfaces, veremos que ya tenemos la NAT configurada.

```
$sudo nano /etc/network/interfaces
```

Cuando lo abramos veremos que tenemos la enp0s3 en dhcp. Nosotros deberemos poner la enp0s8 y la enp0s9 en static. Para editar el archivo copiamos el mismo formato que tiene enp0s3, es decir:

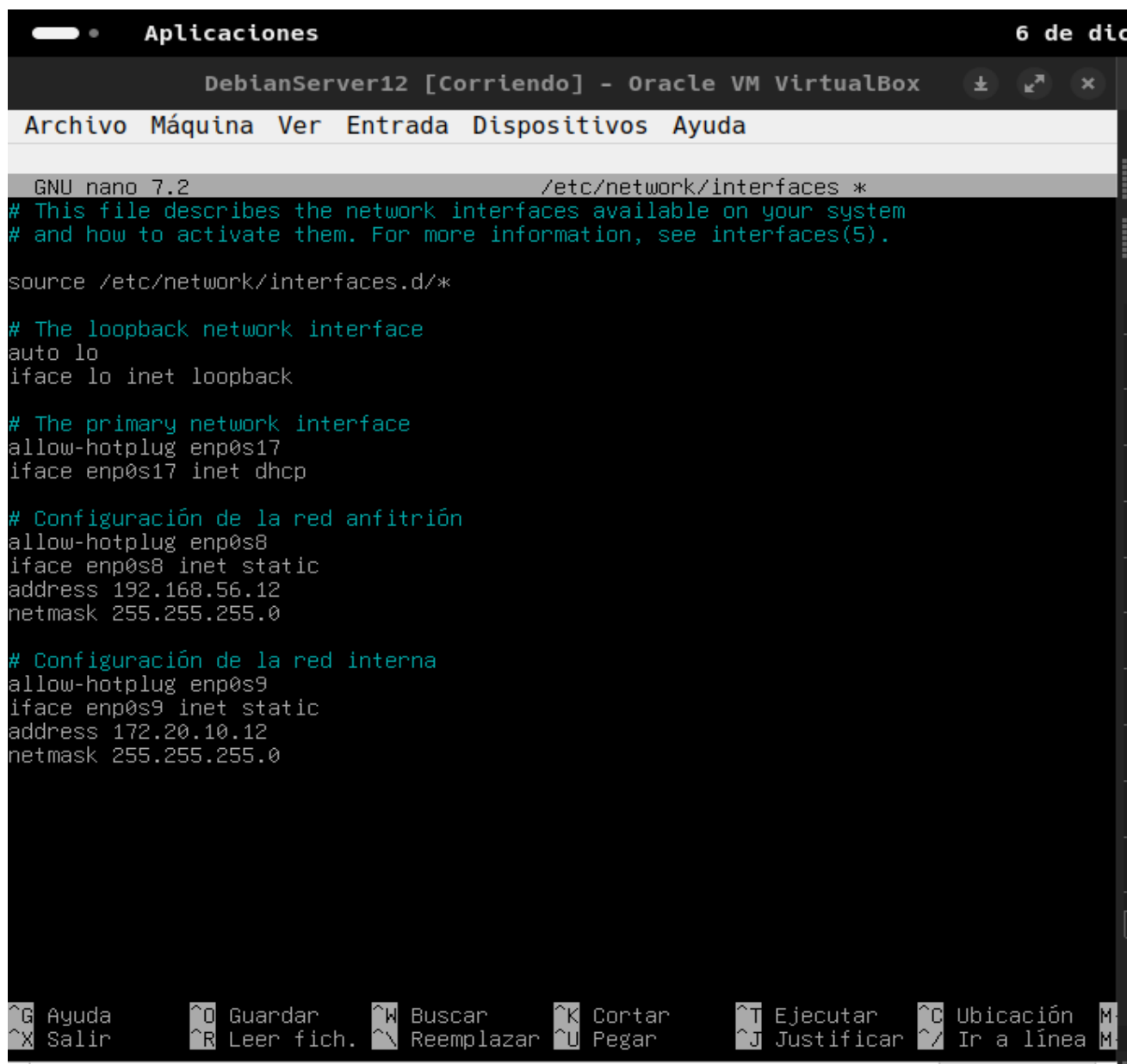
```
allow-hotplug enp0s8
```

```
iface enp0s8 inet static
```

```
address (ponemos una IP del rango de nuestro anfitrión, las mas comunes: 192.168.56.X o 172.20.10.X)
```

```
netmask 255.255.255.0
```

Para la enp0s9 lo mismo, que para la enp0s8 y el rango IP que queremos para nuestra red interna.



```
Aplicaciones 6 de dic
DebianServer12 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
GNU nano 7.2 /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

# The primary network interface
allow-hotplug enp0s17
iface enp0s17 inet dhcp

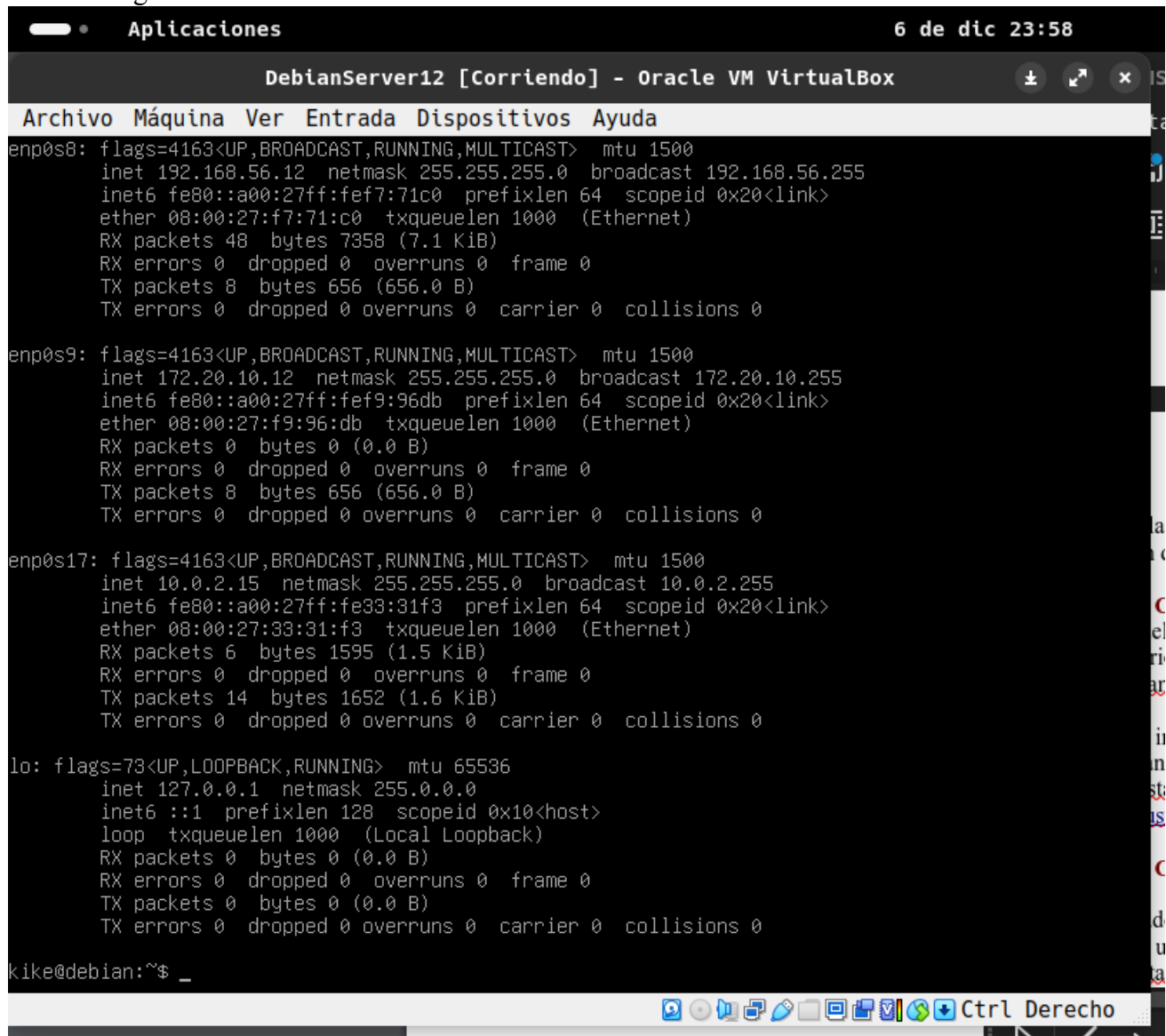
# Configuración de la red anfitrión
allow-hotplug enp0s8
iface enp0s8 inet static
address 192.168.56.12
netmask 255.255.255.0

# Configuración de la red interna
allow-hotplug enp0s9
iface enp0s9 inet static
address 172.20.10.12
netmask 255.255.255.0

^G Ayuda  ^O Guardar  ^W Buscar  ^K Cortar  ^T Ejecutar  ^C Ubicación  M
^X Salir  ^R Leer fich.  ^\ Reemplazar  ^U Pegar  ^J Justificar  ^_ Ir a línea  M
```

Reiniciamos la máquina (sudo reboot) y comprobamos con ifconfig que efectivamente tenemos la configuración de red que hemos configurado.

del ifconfig del debian



```
Aplicaciones 6 de dic 23:58
DebianServer12 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.56.12 netmask 255.255.255.0 broadcast 192.168.56.255
    inet6 fe80::a00:27ff:fef7:71c0 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:f7:71:c0 txqueuelen 1000 (Ethernet)
    RX packets 48 bytes 7358 (7.1 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 8 bytes 656 (656.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

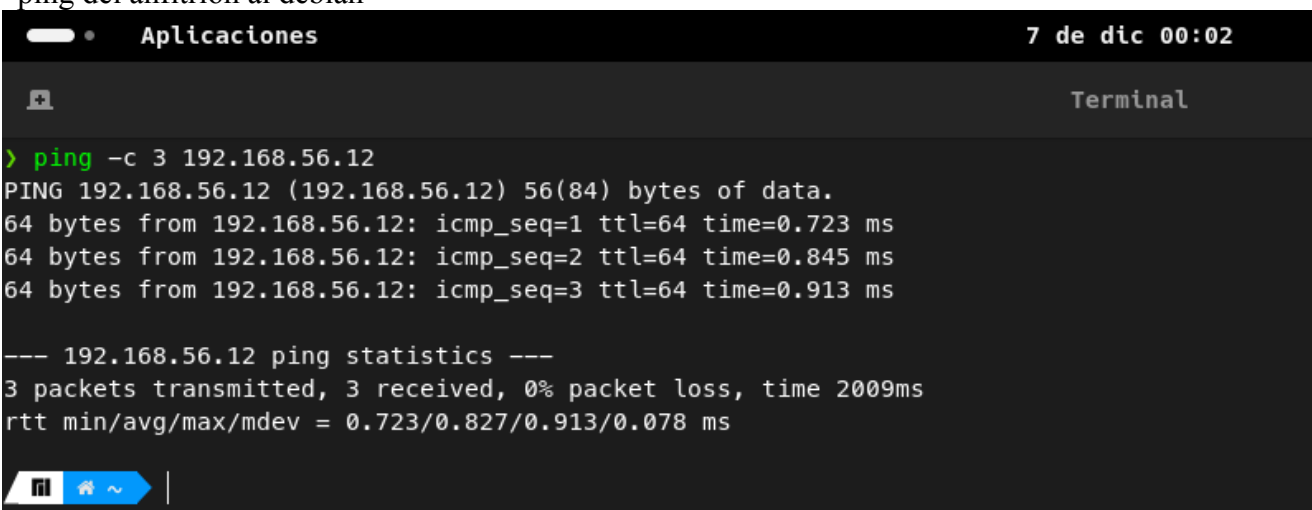
enp0s9: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 172.20.10.12 netmask 255.255.255.0 broadcast 172.20.10.255
    inet6 fe80::a00:27ff:fef9:96db prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:f9:96:db txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 8 bytes 656 (656.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp0s17: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::a00:27ff:fe33:31f3 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:33:31:f3 txqueuelen 1000 (Ethernet)
    RX packets 6 bytes 1595 (1.5 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 14 bytes 1652 (1.6 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

kike@debian:~$ _
```

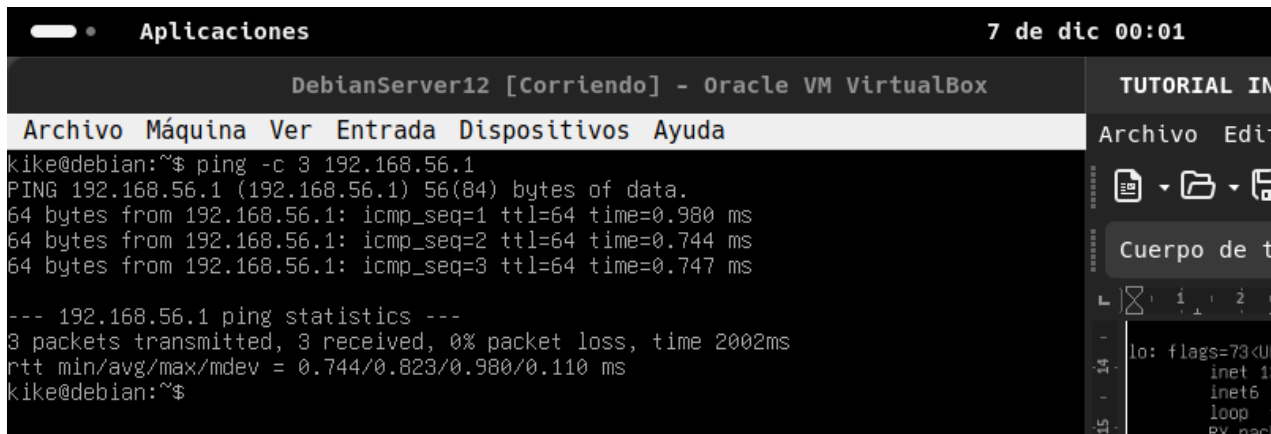
ping del anfitrión al debian



```
Aplicaciones 7 de dic 00:02
Terminal
> ping -c 3 192.168.56.12
PING 192.168.56.12 (192.168.56.12) 56(84) bytes of data.
64 bytes from 192.168.56.12: icmp_seq=1 ttl=64 time=0.723 ms
64 bytes from 192.168.56.12: icmp_seq=2 ttl=64 time=0.845 ms
64 bytes from 192.168.56.12: icmp_seq=3 ttl=64 time=0.913 ms

--- 192.168.56.12 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2009ms
rtt min/avg/max/mdev = 0.723/0.827/0.913/0.078 ms
```

ping del debian al anfitrión



The screenshot shows a terminal window titled "Aplicaciones" with a subtitle "DebianServer12 [Corriendo] - Oracle VM VirtualBox". The terminal output shows a successful ping from the Debian VM to the host IP 192.168.56.1. The output includes the command, the number of bytes, the sequence number, the TTL, and the time taken for three packets. It also shows the ping statistics: 3 packets transmitted, 3 received, 0% packet loss, and the round-trip time (rtt) statistics.

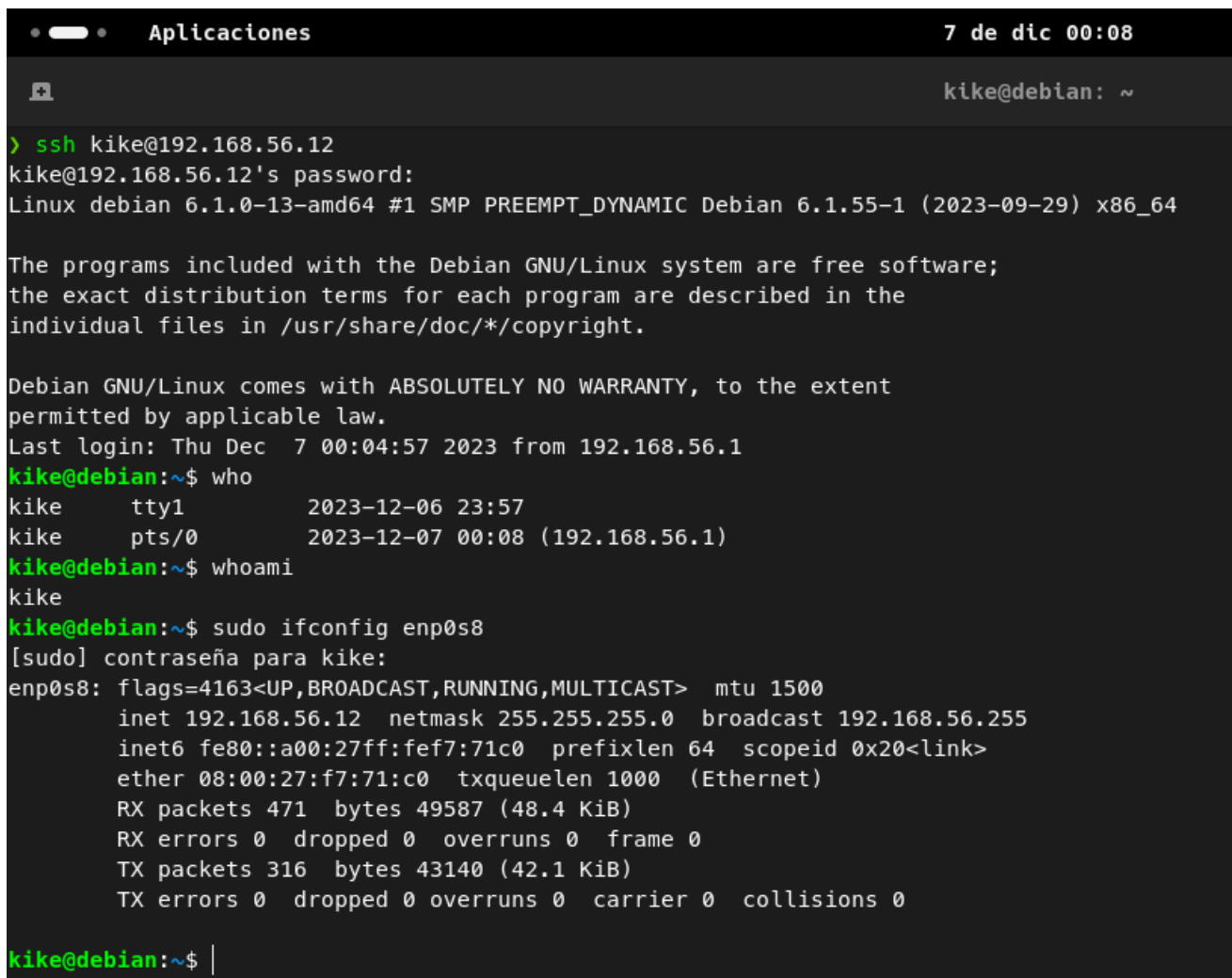
```
kike@debian:~$ ping -c 3 192.168.56.1
PING 192.168.56.1 (192.168.56.1) 56(84) bytes of data.
64 bytes from 192.168.56.1: icmp_seq=1 ttl=64 time=0.980 ms
64 bytes from 192.168.56.1: icmp_seq=2 ttl=64 time=0.744 ms
64 bytes from 192.168.56.1: icmp_seq=3 ttl=64 time=0.747 ms

--- 192.168.56.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 0.744/0.823/0.980/0.110 ms
kike@debian:~$
```

Como hemos instalado el SSH server vamos a conectarnos mediante SSH al Debian desde el terminal del anfitrión. Si no lo hubiésemos instalado:

(\$sudo apt install openssh-server)

\$ssh [nombreusuarioDebian@IPdelDebian](#)



The screenshot shows a terminal window titled "Aplicaciones" with a subtitle "7 de dic 00:08". The terminal output shows an SSH connection from the host to the Debian VM. The user 'kike' is prompted for a password and then logs in. The output shows the Debian version, the system's warranty disclaimer, the last login time, and the results of the 'who', 'whoami', and 'sudo ifconfig enp0s8' commands.

```
> ssh kike@192.168.56.12
kike@192.168.56.12's password:
Linux debian 6.1.0-13-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.55-1 (2023-09-29) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Dec  7 00:04:57 2023 from 192.168.56.1
kike@debian:~$ who
kike      tty1          2023-12-06 23:57
kike      pts/0          2023-12-07 00:08 (192.168.56.1)
kike@debian:~$ whoami
kike
kike@debian:~$ sudo ifconfig enp0s8
[sudo] contraseña para kike:
enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 192.168.56.12  netmask 255.255.255.0  broadcast 192.168.56.255
    inet6 fe80::a00:27ff:fef7:71c0  prefixlen 64  scopeid 0x20<link>
    ether 08:00:27:f7:71:c0  txqueuelen 1000  (Ethernet)
    RX packets 471  bytes 49587 (48.4 KiB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 316  bytes 43140 (42.1 KiB)
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

kike@debian:~$ |
```

En un navegador del anfitrión pondremos la IP del Debian. ¿Nos sale una página apache? En caso contrario. No uséis https.
\$sudo apt install apache2



Configuración del servidor web Apache y de su Firewall

En la siguiente parte de la práctica vamos a realizar las siguientes acciones:

Instalaremos el firewall para Debian ufw, para abrir solo determinados puertos y acceso a aplicaciones.

Abriremos los puertos de openssh y http para que los clientes puedan utilizar estos servicios.

Cambiaremos el puerto por el que escucha http 80 por el 8080 modificando el fichero de configuración de puertos del apache.

Modificaremos el DocumentRoot, que es el sitio desde donde se sirven las páginas.

Accederemos desde la página anfitrión por el puerto 8080 a nuestro servidor de páginas web.

Crearemos un servidor de archivos y accederemos asimismo desde nuestro sistema anfitrión al servidor de archivos.

Lo primero que debemos comprobar que es tenemos instalado, el OpenSSH y el apache2. (Si hemos realizado bien la instalación lo tendremos instalado). En caso contrario ejecutamos:

```
$sudo apt update
```

```
$sudo apt install apache2
```

```
$sudo apt install openssh-server
```

Acto seguido procederemos a la instalación del Firewall, para lo cual ejecutaremos las siguientes órdenes:

```
$sudo apt update (si lo hemos hecho antes, no sería necesario)
```

```
$sudo apt install ufw
```

Para ver el estado del firewall:

```
$sudo ufw status verbose
```

¿Está activado o desactivado?

```
Aplicaciones 7 de dic
DebianServer12 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda

kike@debian:~$ sudo ufw status verbose
Status: inactive
kike@debian:~$
```

Ahora vamos a añadir 3 reglas, la primera para abrir el puerto del ssh (22), para poder acceder al servidor remotamente por ssh, la segunda para abrir el puerto 80 para las conexiones por http y la tercera para abrir el puerto 8080, que utilizaremos posteriormente.

```
$sudo ufw allow OpenSSH
$sudo ufw allow http
$sudo allow 8080/tcp
```

Para activar el firewall, deberíamos activar el mismo:

```
$sudo ufw enable
```

Volvemos a comprobar el estado:

```
$sudo ufw status verbose
```

```
Aplicaciones 7 de dic
DebianServer12 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda

kike@debian:~$ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), disabled (routed)
New profiles: skip

To Action From
--
22/tcp (OpenSSH) ALLOW IN Anywhere
80/tcp ALLOW IN Anywhere
8080/tcp ALLOW IN Anywhere
22/tcp (OpenSSH (v6)) ALLOW IN Anywhere (v6)
80/tcp (v6) ALLOW IN Anywhere (v6)
8080/tcp (v6) ALLOW IN Anywhere (v6)

kike@debian:~$ _
```

Una vez configuradas las reglas del firewall, ahora vamos a crear el directorio desde donde se servirán nuestras páginas webs: Para ello crearemos la carpeta MisPaginas dentro de /var/www/html.

```
$cd /var/www/html
$sudo mkdir MisPaginas
$ls -l
```

Como lo que queremos es que el index.html se sirva desde esa carpeta, lo moveremos a MisPaginas

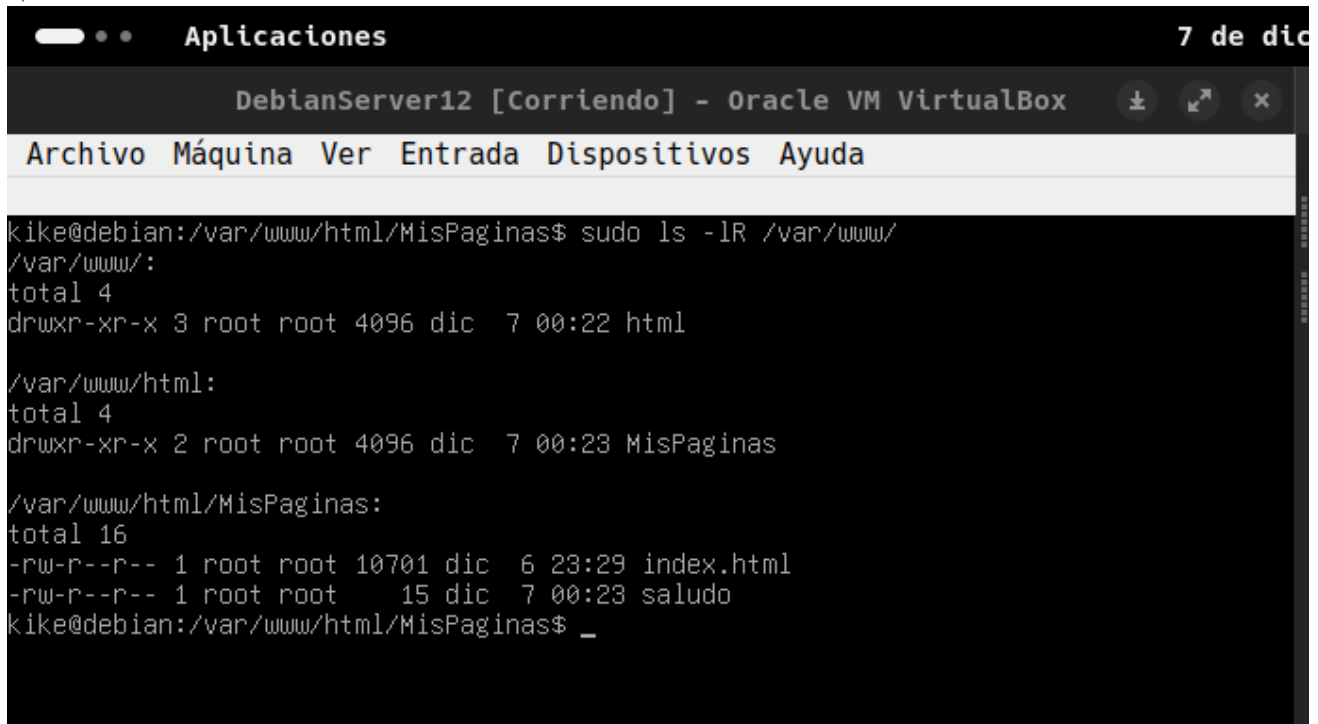
y crearemos un archivo en MisPaginas que se llame saludo y contenga una linea con el texto “Buenos días”.

```
$sudo mv index.html MisPaginas/
```

```
$cd MisPaginas
```

```
$sudo nano saludo
```

```
$sudo ls -lR /var/www/
```



```
Aplicaciones 7 de dic
DebianServer12 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda

kike@debian:/var/www/html/MisPaginas$ sudo ls -lR /var/www/
/var/www/:
total 4
drwxr-xr-x 3 root root 4096 dic  7 00:22 html

/var/www/html:
total 4
drwxr-xr-x 2 root root 4096 dic  7 00:23 MisPaginas

/var/www/html/MisPaginas:
total 16
-rw-r--r-- 1 root root 10701 dic  6 23:29 index.html
-rw-r--r-- 1 root root    15 dic  7 00:23 saludo
kike@debian:/var/www/html/MisPaginas$ _
```

Todos los archivos de configuración de Apache están en /etc/apache2

```
$cd /etc/apache2
```

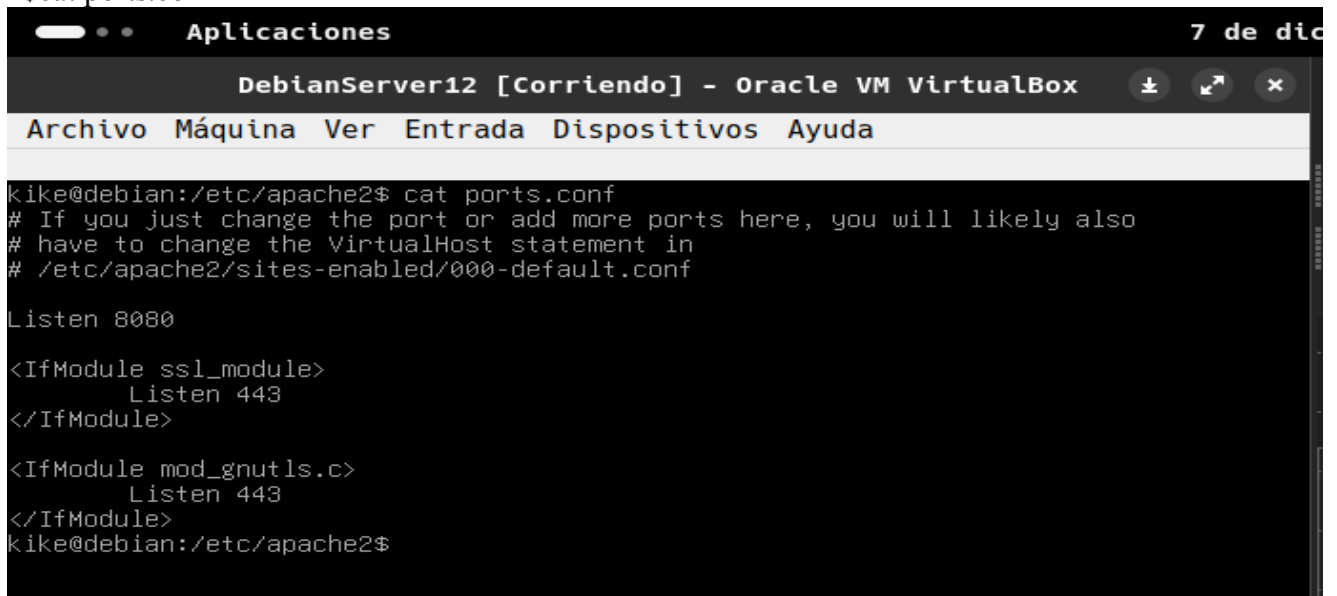
```
$sudo ls -l
```

Como lo siguiente que queremos realizar es cambiar el puerto de escucha. Editaremos dicho archivo. Para editarlo:

```
$sudo nano ports.conf
```

Y dentro cambiaremos la linea donde pone Listen 80 por la línea Listen 8080

```
$cat ports.conf
```



```
Aplicaciones 7 de dic
DebianServer12 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda

kike@debian:/etc/apache2$ cat ports.conf
# If you just change the port or add more ports here, you will likely also
# have to change the VirtualHost statement in
# /etc/apache2/sites-enabled/000-default.conf

Listen 8080

<IfModule ssl_module>
    Listen 443
</IfModule>

<IfModule mod_gnutls.c>
    Listen 443
</IfModule>
kike@debian:/etc/apache2$
```


Ahora lo que queremos editar es la orden que marca el sitio por defecto. Para ellos editaremos el archivo de configuración /etc/apache2/sites-available/000-default.conf.

```
$sudo nano 000-default.conf
```

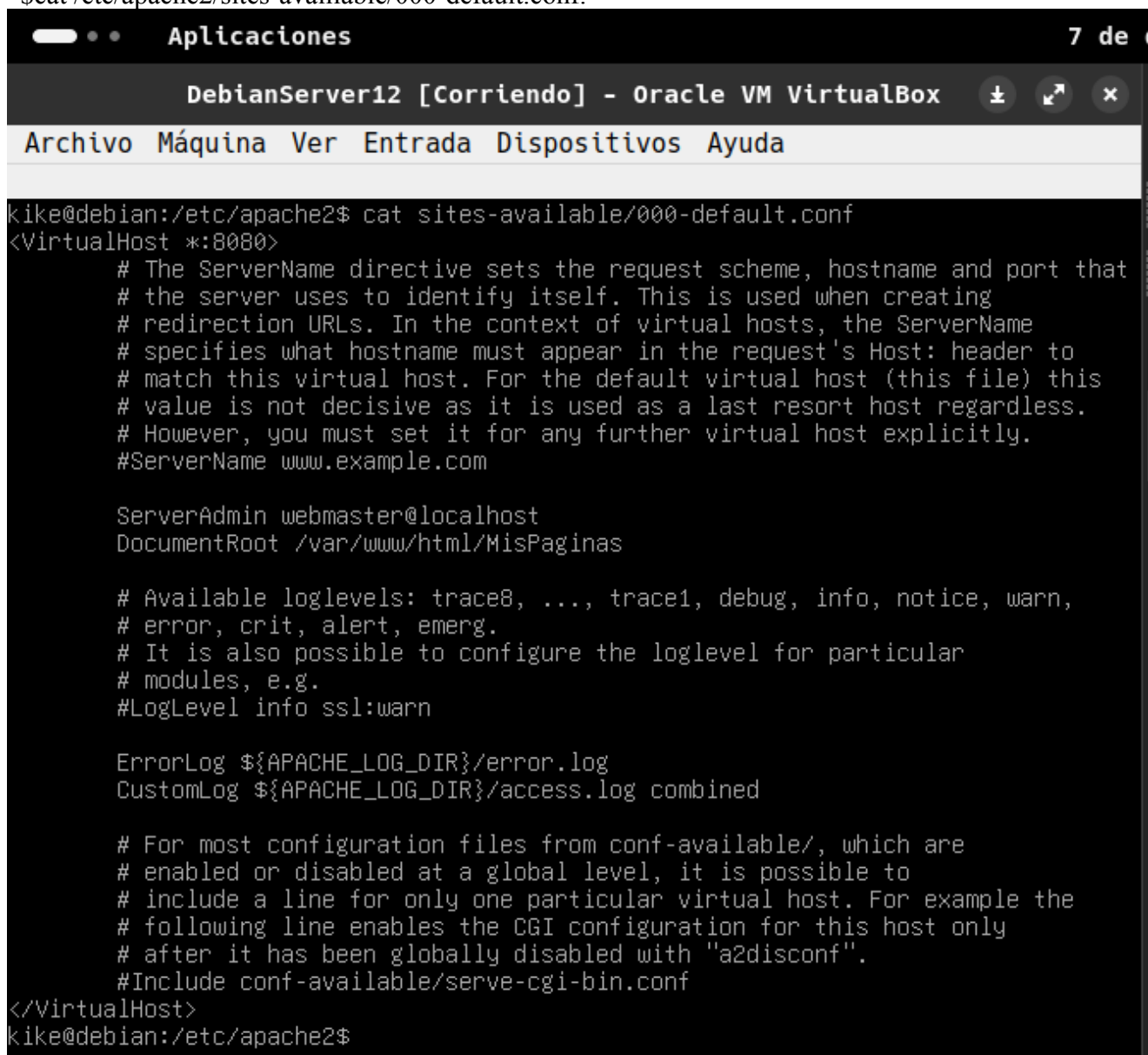
-Cambiamos la primera linea que contiene <VirtualHost *:80> por <VirtualHost *:8080>

-Cambiamos la linea DocumentRoot /var/www/html por DocumentRoot

/var/www/html/MisPaginas.

Guardamos el documento.

```
$cat /etc/apache2/sites-available/000-default.conf.
```



```
Aplicaciones 7 de d
DebianServer12 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
kike@debian:/etc/apache2$ cat sites-available/000-default.conf
<VirtualHost *:8080>
    # The ServerName directive sets the request scheme, hostname and port that
    # the server uses to identify itself. This is used when creating
    # redirection URLs. In the context of virtualhosts, the ServerName
    # specifies what hostname must appear in the request's Host: header to
    # match this virtual host. For the default virtual host (this file) this
    # value is not decisive as it is used as a last resort host regardless.
    # However, you must set it for any further virtual host explicitly.
    #ServerName www.example.com

    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html/MisPaginas

    # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
    # error, crit, alert, emerg.
    # It is also possible to configure the loglevel for particular
    # modules, e.g.
    #LogLevel info ssl:warn

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

    # For most configuration files from conf-available/, which are
    # enabled or disabled at a global level, it is possible to
    # include a line for only one particular virtual host. For example the
    # following line enables the CGI configuration for this host only
    # after it has been globally disabled with "a2disconf".
    #Include conf-available/serve-cgi-bin.conf
</VirtualHost>
kike@debian:/etc/apache2$
```

Como hemos cambiado la configuración del Apache, paramos el servicio apache y volvedlo a arrancar utilizando estos comandos:

```
$sudo service apache2 stop
```

```
$sudo service apache2 start
```

```
$sudo service apache2 status
```

```
Aplicaciones 7 de dic
```

```
DebianServer12 [Corriendo] - Oracle VM VirtualBox
```

```
Archivo Máquina Ver Entrada Dispositivos Ayuda
```

```
kike@debian:/etc/apache2$ sudo service apache2 stop
kike@debian:/etc/apache2$ sudo service apache2 start
kike@debian:/etc/apache2$ sudo service apache2 status
• apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; preset: enabled)
   Active: active (running) since Thu 2023-12-07 00:31:19 CET; 16s ago
     Docs: https://httpd.apache.org/docs/2.4/
  Process: 2102 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
 Main PID: 2106 (apache2)
    Tasks: 55 (limit: 2307)
   Memory: 8.7M
      CPU: 57ms
   CGroup: /system.slice/apache2.service
           └─2106 /usr/sbin/apache2 -k start
             └─2107 /usr/sbin/apache2 -k start
               └─2108 /usr/sbin/apache2 -k start

dic 07 00:31:19 debian systemd[1]: Starting apache2.service - The Apache HTTP Server
dic 07 00:31:19 debian apachectl[2105]: AH00558: apache2: Could not reliably determine
dic 07 00:31:19 debian systemd[1]: Started apache2.service - The Apache HTTP Server
lines 1-17/17 (END)
```

Ahora tenemos Apache activo y configurado y el Firewall lo mismo. Así pues, recordamos la IP del anfitrión en Debian server y en un navegador web del equipo local y escribimos la siguiente url: <http://172.20.10.10:8080/saludo> (la IP, cada uno la que tenga). Como veis le indicamos el puerto por el que tiene que entrar (sino http intentaría sin éxito el 80) y el archivo que nos tiene que mostrar.



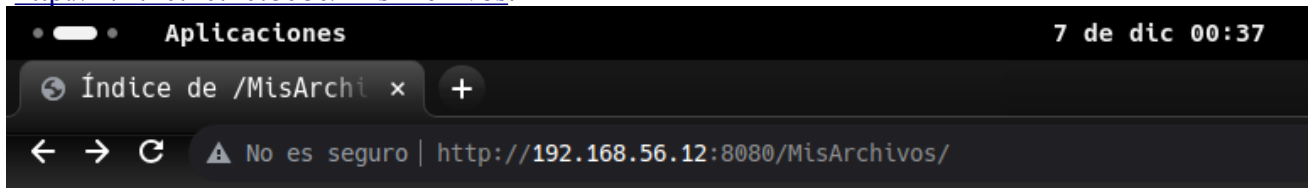
Ahora configuraremos un servidor de archivos. Para ello crearemos la carpeta `/var/www/html/MisPaginas/MisArchivos`.

Copiaremos algunos archivos a la nueva carpeta.



\$sudo mv saludo MisArchivos/

Abrid el navegador de vuestra máquina anfitrión y escribid la siguiente url:

<http://172.20.10.10:8080/MisArchivos>.



Índice de /MisArchivos

Nombre	Última modificación	Tamaño	Descripción
 directorio de padres		-	
 saludos	2023-12-07 00:23	15	

Servidor Apache/2.4.57 (Debian) en el puerto 192.168.56.12 8080