

# PRÁCTICA 1: PREPARACIÓN DE LAS HERRAMIENTAS

David Armenteros Soto

## 1. Introducción

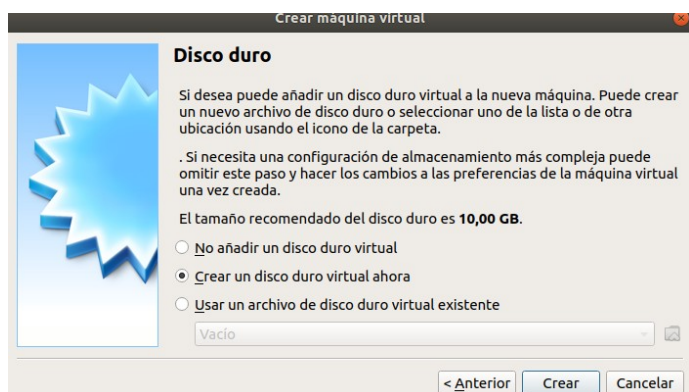
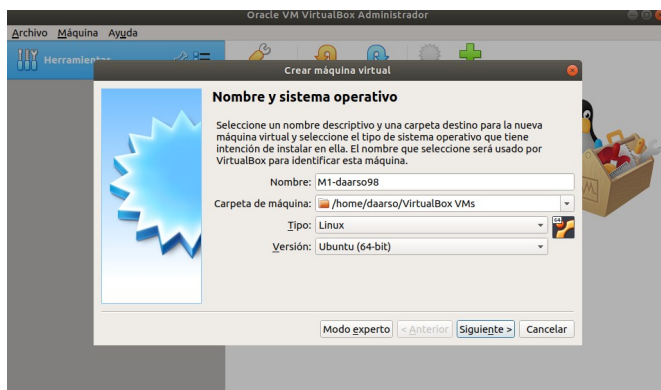
El objetivo principal de la práctica es preparar las herramientas para el correcto desarrollo de la asignatura. La realización de la práctica consistirá en la creación de dos máquinas virtuales con SO linux y servidor web con las herramientas mysql, apache, ssh y curl. El método de virtualización utilizado será virtualBox.

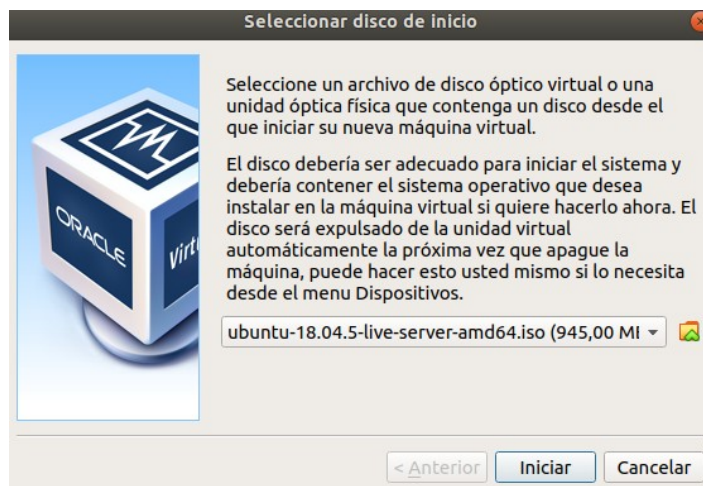
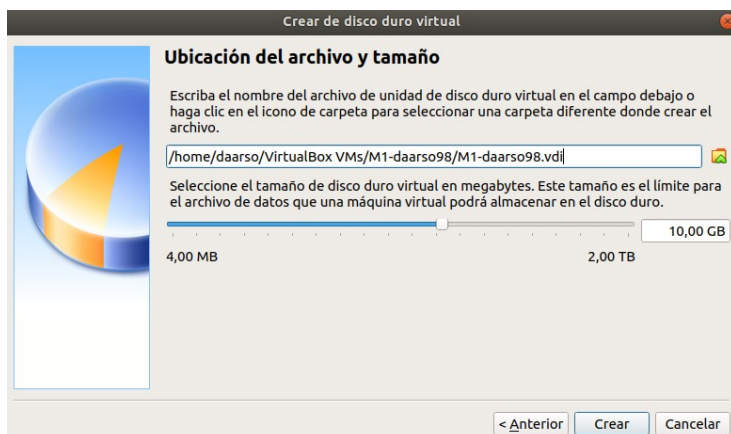
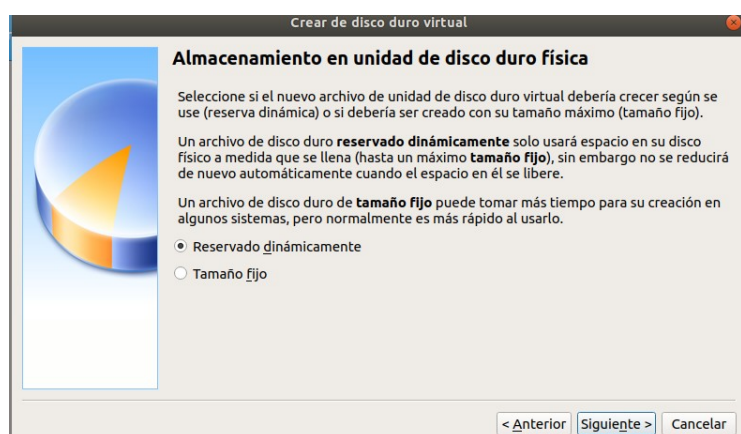
Los objetivos a conseguir son:

- Acceder por **ssh** de una máquina a otra
- Acceder mediante la herramienta **curl** desde una máquina a la otra
- Mostrar configuraciones de red y opciones de **netplan**

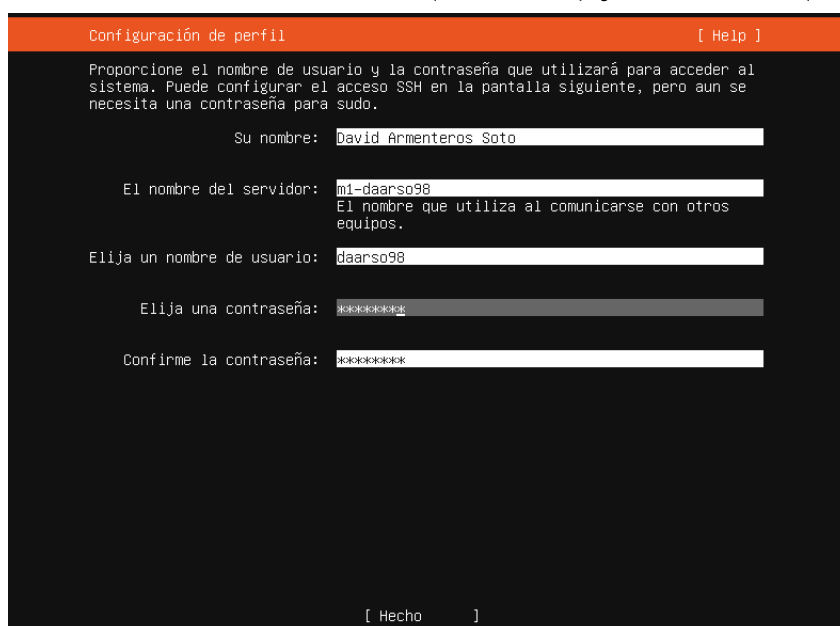
## 2. Instalación

En primer lugar creamos dos máquinas m1-daarso98 y m2-daarso98 con un 1GB de RAM y 10GB de disco duro de forma dinámica. Ambos tienen dos adaptadores: NAT y solo-anfitrión para la comunicación con el anfitrión y para la comunicación con ambas máquinas virtuales.

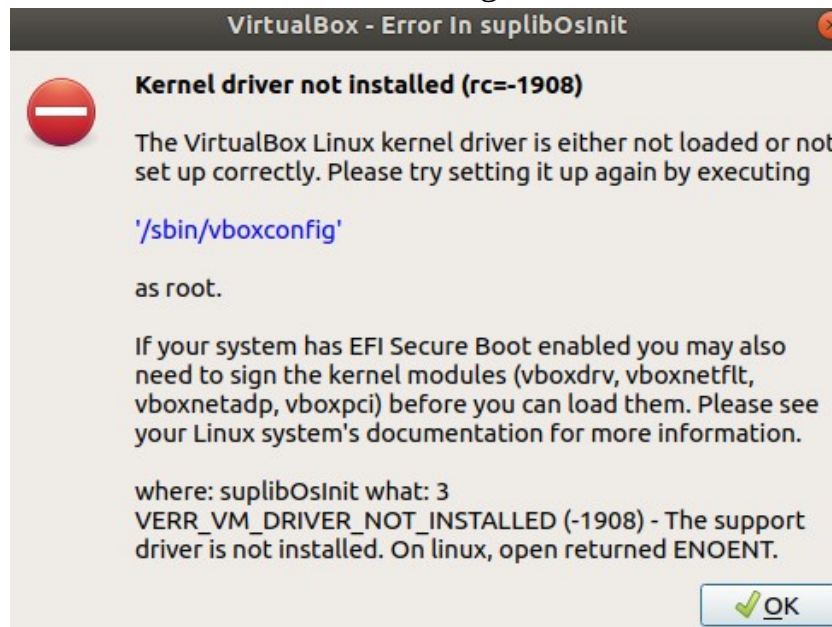




Finalmente, rellenamos los datos de usuario(*daarso98*) y contraseña (*Swap1234*)



En todo este proceso me he encontrado con el siguiente error:



Para solucionarlo, he entrado en la BIOS del anfitrión y he deshabilitado el **secureBoot** (disabled).

Tras instalar las dos máquinas Linux con Ubuntu Server, en ambas vamos a realizar una instalación completa de servidor web: **Apache + PHP + MySQL**, así como servidor SSH para disponer de acceso remoto y del servidor web.

```
daarso98@m1-daarso98:~$ sudo passwd root
[sudo] password for daarso98:
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
```

```
daarso98@m1-daarso98:~$ sudo apt-get install apache2 mysql-server mysql-client_
```

```
daarso98@m1-daarso98:~$ apache2 -v
Server version: Apache/2.4.29 (Ubuntu)
Server built: 2020-08-12T21:33:25
```

```
daarso98@m1-daarso98:~$ ps aux | grep apache
root      2853   0.0  0.4  73960  4480 ?        Ss   16:35   0:00 /usr/sbin/apache2 -k start
www-data  2856   0.0  0.4  826252  4468 ?        S1   16:35   0:00 /usr/sbin/apache2 -k start
www-data  2857   0.0  0.4  826252  4468 ?        S1   16:35   0:00 /usr/sbin/apache2 -k start
daarso98  3999   0.0  0.1  13212  1028 tty1    S+   16:42   0:00 grep --color=auto apache
daarso98@m1-daarso98:~$ sudo service apache2 status
* apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Drop-In: /lib/systemd/system/apache2.service.d
            └─apache2-systemd.conf
   Active: active (running) since Wed 2021-03-03 16:35:37 UTC; 7min ago
     Main PID: 2853 (apache2)
       Tasks: 55 (limit: 1107)
    CGroup: /system.slice/apache2.service
            └─2853 /usr/sbin/apache2 -k start
              └─2856 /usr/sbin/apache2 -k start
                └─2857 /usr/sbin/apache2 -k start

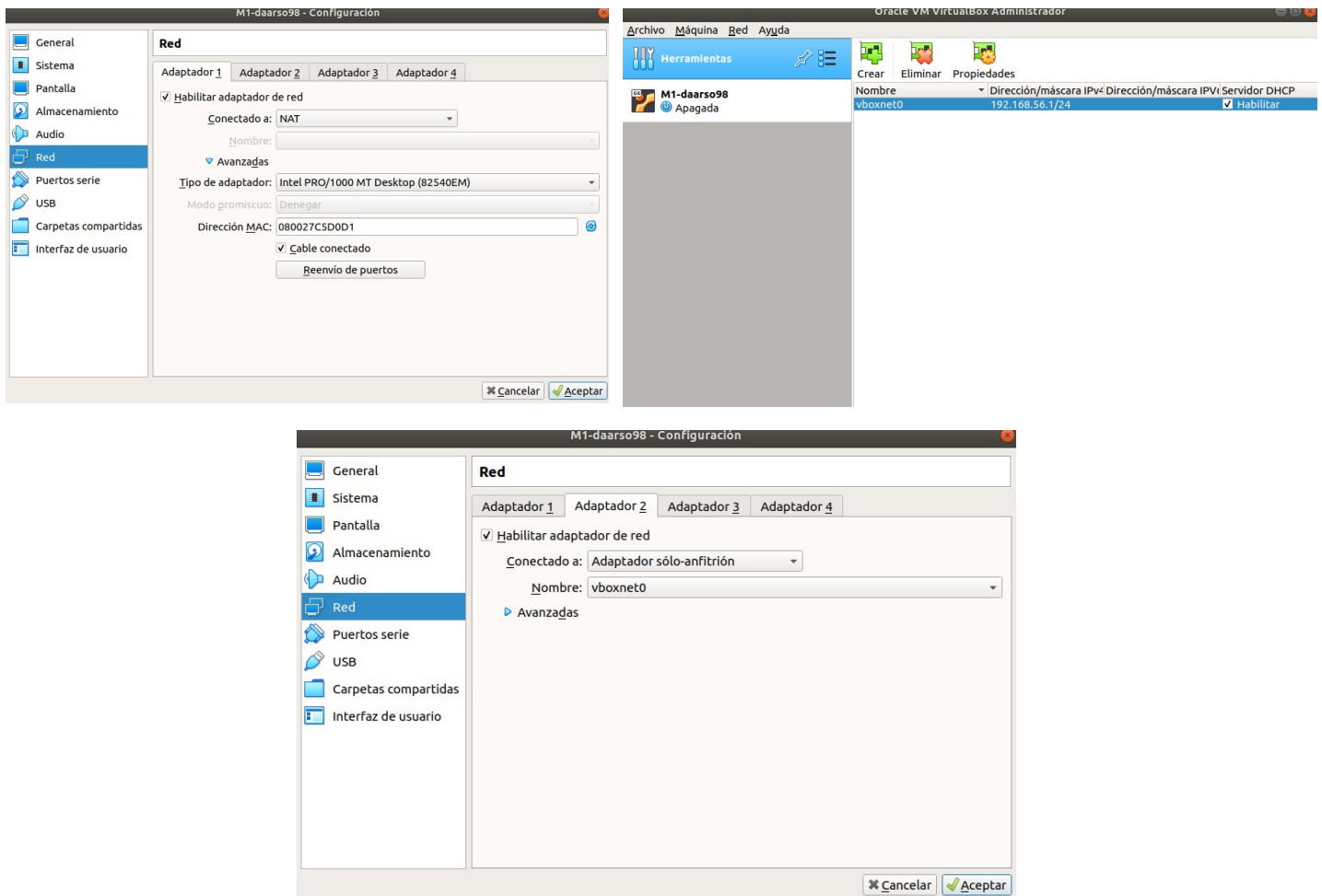
Mar 03 16:35:37 m1-daarso98 systemd[1]: Starting The Apache HTTP Server...
Mar 03 16:35:37 m1-daarso98 apachectl[2828]: AH00558: apache2: Could not reliably determine the s
Mar 03 16:35:37 m1-daarso98 systemd[1]: Started The Apache HTTP Server.
```

## Instalación de la herramienta **curl**:

```
daarso98@m2-daarso98:~$ sudo apt-get install curl
Reading package lists... Done
Building dependency tree
Reading state information... Done
curl is already the newest version (7.58.0-2ubuntu3.12).
curl set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 53 not upgraded.
```

### 3. Añadir y configurar adaptadores de red

A la hora de añadir y configurar los adaptadores de red tenemos la opción de añadirlos manualmente desde la configuración de red de virtualBox



Mediante el comando **ifconfig** podemos visualizar las direcciones ip de cada una de las redes:

```
daarso98@m1-daarso98:~$ ifconfig
enp0s3: 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:fe9a:58e8 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:9a:58:e8 txqueuelen 1000 (Ethernet)
    RX packets 15436 bytes 23168066 (23.1 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1044 bytes 74557 (74.5 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.56.104 netmask 255.255.255.0 broadcast 192.168.56.255
    inet6 fe80::a00:27ff:fe45:7b36 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:45:7b:36 txqueuelen 1000 (Ethernet)
    RX packets 13 bytes 5052 (5.0 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 22 bytes 2943 (2.9 KB)
    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 106 bytes 8586 (8.5 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 106 bytes 8586 (8.5 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

```
daarso98@m2-daarso98:~$ ifconfig
enp0s3: 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:fedf:3158 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:fd:31:58 txqueuelen 1000 (Ethernet)
    RX packets 15501 bytes 23171121 (23.1 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1113 bytes 78900 (78.9 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.56.105 netmask 255.255.255.0 broadcast 192.168.56.255
    inet6 fe80::a00:27ff:fedf:29bd prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:df:29:bd txqueuelen 1000 (Ethernet)
    RX packets 9 bytes 3190 (3.1 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 21 bytes 2873 (2.8 KB)
    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 106 bytes 8586 (8.5 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 106 bytes 8586 (8.5 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

El fin de la práctica es gestionar las configuraciones de red con netplan. Para ello nos dirigimos al archivo de configuración con terminación **.yaml** que encontraremos en el directorio **etc/netplan**. Aplicaremos **sudo netplan apply** para confirmar los cambios.

```
daarso98@m1-daarso98:~$ cd /etc/netplan
daarso98@m1-daarso98:/etc/netplan$ sudo nano 00-installer-config.yaml _
```

```
GNU nano 2.9.3 /etc/netplan/00-installer-config.yaml
# This is the network config written by 'subiquity'
network:
  ethernets:
    enp0s3:
      dhcp4: true
    enp0s8:
      dhcp4: no
      dhcp6: no
      addresses: [192.168.56.104/24, ]
      #gateway4: 192.168.56.1
      nameservers:
        addresses: [8.8.8.8, 8.8.4.4]
  version: 2
```

```
GNU nano 2.9.3 /etc/netplan/00-installer-config.yaml
# This is the network config written by 'subiquity'
network:
  ethernets:
    enp0s3:
      dhcp4: true
    enp0s8:
      dhcp4: no
      dhcp6: no
      addresses: [192.168.56.105/24, ]
      #gateway4: 192.168.56.1
      nameservers:
        addresses: [8.8.8.8, 8.8.4.4]
  version: 2
```

```
daarso98@m1-daarso98:/etc/netplan$ sudo netplan apply
```

## 4. Comprobación del funcionamiento

Para comprobar que ambas máquinas se ven entre sí podemos hacerlo con **ping**

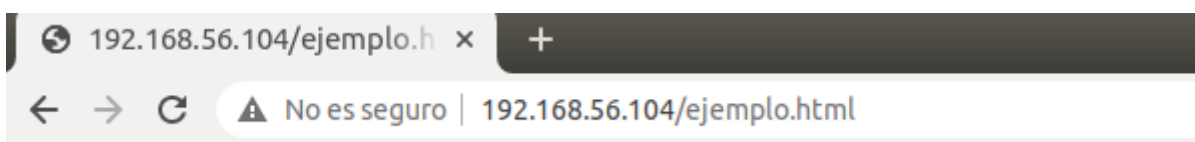
```
daarso98@m1-daarso98:~$ ping 192.168.56.105
PING 192.168.56.105 (192.168.56.105) 56(84) bytes of data.
64 bytes from 192.168.56.105: icmp_seq=1 ttl=64 time=0.454 ms
64 bytes from 192.168.56.105: icmp_seq=2 ttl=64 time=1.11 ms
64 bytes from 192.168.56.105: icmp_seq=3 ttl=64 time=0.991 ms
^C
```

```
daarso98@m2-daarso98:~$ ping 192.168.56.104
PING 192.168.56.104 (192.168.56.104) 56(84) bytes of data.
64 bytes from 192.168.56.104: icmp_seq=1 ttl=64 time=0.881 ms
64 bytes from 192.168.56.104: icmp_seq=2 ttl=64 time=1.07 ms
64 bytes from 192.168.56.104: icmp_seq=3 ttl=64 time=0.937 ms
64 bytes from 192.168.56.104: icmp_seq=4 ttl=64 time=0.889 ms
64 bytes from 192.168.56.104: icmp_seq=5 ttl=64 time=1.03 ms
64 bytes from 192.168.56.104: icmp_seq=6 ttl=64 time=0.525 ms
```

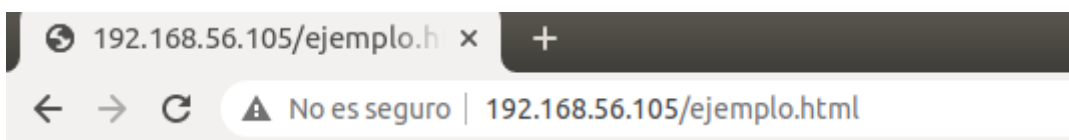
## 5. Configurar servicios de apache

Apache tiene una configuración básica por defecto. El puerto de escucha es el **80** y el directorio virtual es **var/www/html**. Vamos a comprobar su funcionamiento, para ello creamos un fichero ejemplo.html contenido en el directorio anteriormente mencionado:

```
daarso98@m2-daarso98:~$ cat /var/www/html/ejemplo.html
<HTML>
<BODY>
Web de ejemplo de daarso98@correo.ugr.es
Email: daarso98@correo.ugr.es
</BODY>
</HTML>
```



Web de ejemplo de daarso98 para SWAP Email: daarso98@correo.ugr.es



Web de ejemplo de daarso98@correo.ugr.es Email: daarso98@correo.ugr.es



El directorio `/etc/apache2` contiene el archivo de configuración `ports.conf` ,  
procedemos a cambiar el puerto por defecto a escuchar en el puerto 8080

```
daarso98@m1-daarso98:/etc/apache2$ ls -l
total 80
-rw-r--r-- 1 root root 7224 Aug 12 2020 apache2.conf
drwxr-xr-x 2 root root 4096 Mar 3 20:58 conf-available
drwxr-xr-x 2 root root 4096 Mar 3 20:59 conf-enabled
-rw-r--r-- 1 root root 1782 Jul 16 2019 envvars
-rw-r--r-- 1 root root 31063 Jul 16 2019 magic
drwxr-xr-x 2 root root 12288 Mar 3 20:59 mods-available
drwxr-xr-x 2 root root 4096 Mar 3 20:59 mods-enabled
-rw-r--r-- 1 root root 322 Mar 9 13:57 ports.conf
drwxr-xr-x 2 root root 4096 Mar 3 20:59 sites-available
drwxr-xr-x 2 root root 4096 Mar 3 20:59 sites-enabled
```

```
daarso98@m1-daarso98:/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>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
daarso98@m1-daarso98:/etc/apache2$
```

```
daarso98@m1-daarso98:/etc/apache2$ sudo netstat -anp | grep apache2
tcp6      0      0 :::8080          :::*              LISTEN     5036/apache2
```



Web de ejemplo de daarso98 para SWAP Email: daarso98@correo.ugr.es



El directorio **/etc/apache2/conf-enabled** contiene el archivo de configuración **00-default.conf** , en el vamos a configurar un nuevo directorio virtual.

```
daarso98@m1-daarso98:/etc/apache2/sites-enabled$ sudo nano 000-default.conf
```

Para realizar una prueba, he creado un fichero **index.html** en el nuevo directorio virtual **/webprueba** con alias **/web**

```
daarso98@m1-daarso98:/webprueba$ cat index.html  
HELLO WORLD! DAARSO98
```

```
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>  
  
Alias /web /webprueba  
  
<Directory /webprueba>  
AllowOverride None  
Require all granted  
</Directory>
```



← → ↻ No es seguro | 192.168.56.104:8080/web/  
Aplicaciones Google Fotos - Google... GMAIL YouTube Webmail UGR Acceso identif  
HELLO WORLD! DAARSO98

## 6. Configurar servicio ssh

Para acceder mediante ssh tenemos que poner el siguiente comando:

`ssh <IP_destino> -l <nombre_usuario>` o `ssh <nombre_usuario>@<IP_destino>`

Como se puede apreciar en la configuración por defecto me pide la contraseña para acceder de forma remota.

```
daarso98@m2-daarso98:~$ ssh 192.168.56.104 -l daarso98
The authenticity of host '192.168.56.104 (192.168.56.104)' can't be established.
ECDSA key fingerprint is SHA256:5S4GF0Bi030fI27NmaXUtPJWdq4Fz6B+U3kVgikoHmE.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.56.104' (ECDSA) to the list of known hosts.
daarso98@192.168.56.104's password:
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 4.15.0-136-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Wed Mar  3 21:33:04 UTC 2021

System load:  0.04               Processes:           94
Usage of /:   43.4% of 8.79GB    Users logged in:    1
Memory usage: 33%               IP address for enp0s3: 10.0.2.15
Swap usage:   0%                IP address for enp0s8: 192.168.56.104

54 packages can be updated.
0 updates are security updates.

New release '20.04.2 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Wed Mar  3 20:53:12 2021
daarso98@m1-daarso98:~$ _
```

En la nueva configuración que voy a realizar voy a cambiar el puerto al 2222, para ello nos dirigimos al directorio `/etc/ssh` al archivo de configuración `sshd_config`. A la hora de realizar la conexión tenemos que añadir la siguiente opción `-p <PUERTO>`

```
GNU nano 2.9.3          sshd_config          Modified
# $OpenBSD: sshd_config,v 1.101 2017/03/14 07:19:07 djm Exp $

# This is the sshd server system-wide configuration file. See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/bin:/bin:/usr/sbin:/sbin

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented. Uncommented options override the
# default value.

Port 2222
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:
```

```

daarso98@m2-daarso98:~$ ssh daarso98@192.168.56.104 -p 2222
daarso98@192.168.56.104's password:
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 4.15.0-136-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Tue Mar  9 17:47:40 UTC 2021

System load:  0.0                       Processes:            94
Usage of /:   43.5% of 8.79GB           Users logged in:     1
Memory usage: 33%                       IP address for enp0s3: 10.0.2.15
Swap usage:   0%                       IP address for enp0s8: 192.168.56.104

54 packages can be updated.
0 updates are security updates.

Last login: Wed Mar  3 21:39:32 2021 from 192.168.56.105
daarso98@m1-daarso98:~$ _

```

A continuación, voy a realizar un acceso remoto seguro sin contraseña, usando claves públicas y privadas.

En el archivo de configuración **/etc/ssh/sshd\_config** deshabilitamos la opción **PasswordAuthentication**

```

GNU nano 2.9.3          sshd_config          Modified
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

# Expect .ssh/authorized_keys2 to be disregarded by default in future.
#AuthorizedKeysFile         .ssh/authorized_keys .ssh/authorized_keys2

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
#HostbasedAuthentication no
# Change to yes if you don't trust ~/.ssh/known_hosts for
# HostbasedAuthentication
#IgnoreUserKnownHosts no
# Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes

# To disable tunneled clear text passwords, change to no here!
PasswordAuthentication no
#PermitEmptyPasswords no

# Change to yes to enable challenge-response passwords (beware issues with
# some PAM modules and threads)
ChallengeResponseAuthentication no

# Kerberos options
#KerberosAuthentication no

```

Con el comando **ssh-keygen** creamos el par de claves públicas/privadas que están contenidas en los archivos **id\_rsa** e **id\_rsa.pub** respectivamente.

Con el comando **ssh-copy-id** mandamos la clave al servidor que nos queremos conectar. De esta forma ya no nos pedirá la contraseña.

```
daarso98@m2-daarso98:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/daarso98/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/daarso98/.ssh/id_rsa.
Your public key has been saved in /home/daarso98/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:pg976I8H505jQZ0FJPda3kGflxDGnwyxCtP/IpB7i0A daarso98@m2-daarso98
The key's randomart image is:
----[RSA 2048]-----+
  ..+.0=+ |
  +.+.+0.0 |
  .00.0.=0+ |
  . ++0. =. |
  . So..... |
  . 000.+ . |
  E0*+0 0 . . |
  .B+.. . . |
  .+=+ |
-----[SHA256]-----+
```

```
daarso98@m1-daarso98:~/ssh$ ssh-copy-id -p 2222 daarso98@192.168.56.105
```

```
daarso98@m1-daarso98:~/ssh$ ssh -p 2222 192.168.56.105
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 4.15.0-136-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Wed Mar 10 12:43:42 UTC 2021

System load: 0.0          Processes:              94
Usage of /:  43.5% of 8.79GB Users logged in:             1
Memory usage: 33%        IP address for enp0s3: 10.0.2.15
Swap usage:  0%          IP address for enp0s8: 192.168.56.105

54 packages can be updated.
0 updates are security updates.

New release '20.04.2 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Wed Mar 10 11:13:25 2021 from 192.168.56.104
```

## 7. Configurar servicio curl

Creamos un archivo html en el directorio **var/www/html** llamado **ejemplo.html** y accedemos a él desde el anfitrión y desde la otra máquina

```
daarso98@m2-daarso98:~$ cat /var/www/html/ejemplo.html
<HTML>
<BODY>
Web de ejemplo de daarso98@correo.ugr.es
Email: daarso98@correo.ugr.es
</BODY>
</HTML>
```

```
daarso98@m1-daarso98:~$ curl 192.168.56.105/ejemplo.html
<HTML>
<BODY>
Web de ejemplo de daarso98@correo.ugr.es
Email: daarso98@correo.ugr.es
</BODY>
</HTML>
```

```
daarso98@m2-daarso98:/etc/netplan$ curl 192.168.56.104:8080/ejemplo.html
<HTML>
<BODY>
Web de ejemplo de daarso98 para SWAP
Email: daarso98@correo.ugr.es
</BODY>
</HTML>
```

Con la opción **-o** se puede descargar una imagen de una url y renombrarla como imagen.png

```
daarso98@m1-daarso98:~$ curl -o imagen.png https://www.google.es/images/srpr/logo3w.png
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total     Spent    Left     Speed
100 6748  100 6748    0     0  30672      0 --:--:-- --:--:-- --:--:-- 30533
daarso98@m1-daarso98:~$ ls -l
total 12
-rw-rw-r-- 1 daarso98 daarso98 6748 Mar 10 11:23 imagen.png
```

Con la opción **-O** se puede descargar una imagen de una url con el mismo nombre que tiene en el servidor

```
daarso98@m1-daarso98:~$ curl -O https://www.google.es/images/srpr/logo3w.png
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total     Spent    Left     Speed
100 6748  100 6748    0     0  49617      0 --:--:-- --:--:-- --:--:-- 49617
daarso98@m1-daarso98:~$ ls -l
total 20
-rw-rw-r-- 1 daarso98 daarso98 6748 Mar 10 11:23 imagen.png
-rw-rw-r-- 1 daarso98 daarso98 6748 Mar 10 11:27 logo3w.png
```

Con la opción **-b** se pueden guardar los cookies de una url en un fichero determinado

```
daarso98@m1-daarso98:~$ curl -b cookie.txt https://www.google.es_
```

```
daarso98@m1-daarso98:~$ cat cookie.txt
# Netscape HTTP Cookie File
# https://curl.haxx.se/docs/http-cookies.html
# This file was generated by libcurl! Edit at your own risk.

#HttpOnly_.google.es    TRUE      /          FALSE    1631187558      NID      211=k5hYp7KrnQVRIKoptw
tX71muWkiyipcgvW6r3x7ynAoQwrZeaV3rzdvlxdrBEbAg0rqooPg2Vd65spMiVw1BitL-9QXJ1n2Q-FWOLHmw7XNev4YrMO
mCLC8SyuLftvnmStuyoHifrsebk10KQ9bNPcML_mQYC2VFU
.google.es              TRUE      /          FALSE    2145916800      CONSENT PENDING+582
```

Con la opción **--cookie-jar** se pueden lanzar cookies guardados en un fichero a una url

```
daarso98@m1-daarso98:~$ curl --cookie-jar cookie.txt https://www.google.es
```

Con la opción **-X** se puede hacer una petición POST (en este caso de una imagen) a una URL

```
daarso98@m1-daarso98:~$ curl -X POST -F 'image=@/home/daarso98/imagen.png' http://www.google.es
```

En caso de una petición get, sin añadir ninguna opción nos devuelve el código html de una url

```
daarso98@m1-daarso98:~$ curl https://www.ugr.es
<!DOCTYPE html>
<html lang="es" dir="ltr" prefix="content: http://purl.org/rss/1.0/modules/content/ dc: http://pu
rl.org/dc/terms/ foaf: http://xmlns.com/foaf/0.1/ og: http://ogp.me/ns# rdfs: http://www.w3.org
/2000/01/rdf-schema# schema: http://schema.org/ sioc: http://rdfs.org/sioc/ns# sioc: http://rd
```