

*Pino Fernández, Pablo*

## Servicios de red e Internet

### Práctica 2º trimestre- Servidor de Alojamiento

Para dar alojamiento a páginas web tanto estáticas como dinámicas necesitaremos de varios servicios, formando el grupo “LAMP” (Linux-Apache-MySQL-Php), por lo que primero preparamos dicha base. Comenzaremos actualizando paquetes e instalando **Apache**:

```
$ sudo apt update
```

```
$ sudo apt install apache2
```

```
pablo@pablo-Standard-PC-i440FX-PIIX-1996:~/Escritorio$ sudo apt update
Obj:1 http://es.archive.ubuntu.com/ubuntu focal InRelease
Obj:2 http://security.ubuntu.com/ubuntu focal-security InRelease
Obj:3 http://es.archive.ubuntu.com/ubuntu focal-updates InRelease
Obj:4 http://es.archive.ubuntu.com/ubuntu focal-backports InRelease
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se pueden actualizar 84 paquetes. Ejecute «apt list --upgradable» para verlos.
pablo@pablo-Standard-PC-i440FX-PIIX-1996:~/Escritorio$ sudo apt install apache2
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.2-0
Paquetes sugeridos:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom
Se instalarán los siguientes paquetes NUEVOS:
  apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1
```

Para permitir el paso a través del firewall por el puerto 80 aplicamos:

```
$ sudo ufw allow in "Apache"
```

Continuamos con MySQL. Este podrá establecer la base de datos del sitio, permitiendo páginas estáticas y dinámicas. Iniciamos la descarga e instalación con los siguientes comandos:

```
$ sudo apt install mysql-server
```

Nos pedirá confirmación para continuar como en la imagen.

```
pablo@pablo-Standard-PC-i440FX-PIIX-1996:~/Escritorio$ sudo apt install mysql-server
[sudo] contraseña para usuario:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  libaio1 libcgi-fast-perl libcgi-pm-perl libevent-core-2.1-7 libevent-pthreads-2.1-7 libfcgi-perl libhtml-template-perl libmecab2 mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client-8.0
  mysql-client-core-8.0 mysql-server-8.0 mysql-server-core-8.0
Paquetes sugeridos:
  libipc-sharedcache-perl mailx tinycd
Se instalarán los siguientes paquetes NUEVOS:
  libaio1 libcgi-fast-perl libcgi-pm-perl libevent-core-2.1-7 libevent-pthreads-2.1-7 libfcgi-perl libhtml-template-perl libmecab2 mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client-8.0
  mysql-client-core-8.0 mysql-server mysql-server-8.0 mysql-server-core-8.0
0 actualizados, 16 nuevos se instalarán, 0 para eliminar y 5 no actualizados.
Se necesita descargar 31,1 MB de archivos.
Se utilizarán 261 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [s/n]
Des:1 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 mysql-client-core-8.0 amd64 8.0.28-0ubuntu0.20.04.3 [4.429 kB]
Des:2 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 mysql-client-8.0 amd64 8.0.28-0ubuntu0.20.04.3 [22,0 kB]
Des:3 http://es.archive.ubuntu.com/ubuntu focal/main amd64 libaio1 amd64 8.3-112-5 [7,184 B]
```

```
$ sudo mysql_secure_installation
```

Este proceso pedirá permiso para usar un plugin, que en caso de responder afirmativamente nos pedirá contraseña y nivel de validación de contraseñas entre otros, de cara a la base de datos:

```
pablo@pablo-Standard-PC-i440FX-PIIX-1996:~/Escritorio$ sudo mysql_secure_installation

Securing the MySQL server deployment.

Connecting to MySQL using a blank password.

VALIDATE PASSWORD COMPONENT can be used to test passwords
and improve security. It checks the strength of password
and allows the users to set only those passwords which are
secure enough. Would you like to setup VALIDATE PASSWORD component?

Press y|Y for Yes, any other key for No: y

There are three levels of password validation policy:

LOW      Length >= 8
MEDIUM  Length >= 8, numeric, mixed case, and special characters
STRONG Length >= 8, numeric, mixed case, special characters and dictionary file

Please enter 0 = LOW, 1 = MEDIUM and 2 = STRONG: 0
Please set the password for root here.

New password:
Sorry, you can't use an empty password here.

New password:

Re-enter new password:

Estimated strength of the password: 25
Do you wish to continue with the password provided?(Press y|Y for Yes, any other key for No) :
```

Para instalar PHP solo necesitamos el comando:

```
$ sudo apt install php libapache2-mod-php php-mysql
```

Podemos comprobar su versión con:

```
$ php -v (para comprobar versión instalada)
```

Para añadir el dominio de por ejemplo un apágina html index iremos a modificamos el archivo “/etc/hosts”, donde lo añadimos junto a la ip correspondiente:

```
$ sudo nano /etc/hosts
```

Luego creamos un directorio en “/var/www/html/” donde alojar la página:

```
$ sudo mkdir /var/www/html/marisma
```

A continuación crearemos un archivos de configuración, que se alojarán en “/etc/apache2/sites-available”. Nosotros copiaremos el archivo 000-default.conf original, para tener una base de la que partir. Después lo modificaremos añadiendo el ServerName, ServerAlias y el DocumentRoot:

```
$ sudo cp 000-default.conf 000-marisma.conf
```

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

Podemos comprobar el estado de los archivos de configuración con:

```
$ sudo apachectl configtest
```

Solo necesitamos añadir la página index.html deseada en el directorio pertinente, y podremos comprobarlo con el browser. Antes, eso sí, debemos reiniciar el servicio con:

```
$ sudo systemctl restart apache2
```

## DNS

Para montar nuestro DNS primero debemos instalar bind junto a sus utilidades con el siguiente comando:

```
$ sudo apt-get install bind9 bind9utils bind9-doc
```

```
usuario@usuario-VirtualBox:~/Escritorio$ sudo apt-get install bind9 bind9utils bind9-doc
[sudo] contraseña para usuario:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
```

Luego deberemos modificar el archivo name.conf.options para añadir los DNS de google, indicar “forward only” y dejar los dnssec en “yes” como se indica en la imagen:

```
$ sudo nano /etc/bind/name.conf.options
```

```
pablo@ns1: ~/Escritorio
GNU nano 4.8 /etc/bind/named.conf.options

// If your ISP provided one or more IP addresses for stable
// nameservers, you probably want to use them as forwarders.
// Uncomment the following block, and insert the addresses replacing
// the all-0's placeholder.

forwarders {
    8.8.8.8;
    8.8.4.4;};
forward only;
//=====
// If BIND logs error messages about the root key being expired,
// you will need to update your keys.  See https://www.isc.org/bind-keys
//=====
dnssec-enable yes
dnssec-validation yes;

listen-on-v6 { any; };
};
```

[ 25 líneas escritas ]

^G Ver ayuda ^O Guardar ^W Buscar ^K Cortar Text ^J Justificar ^C Posición  
^X Salir ^R Leer fich. ^\ Reemplazar ^U Pegar ^T Ortografía ^\_ Ir a línea

Para comprobar que todo avanza bien, podemos utilizar un comando para hacer un check:

```
$ sudo named-checkconf
```

Es posible que nos aparezca la siguiente respuesta que sugiere eliminar una de las líneas “**dnssec-enable**” por estar obsoleta, pero no interfiere en el funcionamiento.

```
/etc/bind/named.conf.options:21: option 'dnssec-enable' is obsolete and should be removed
```

Y podemos comprobar que está activo mediante el listado de estados de servicios:

```
$ service --status-all
```

```
[ + ] acpid
[ - ] alsa-utils
[ - ] anacron
[ - ] apache-htcacheclean
[ + ] apache2
[ + ] apparmor
[ + ] appport
[ + ] avahi-daemon
[ - ] bluetooth
[ - ] console-setup.sh
[ + ] cron
[ + ] cups
[ + ] cups-browsed
[ + ] dbus
[ + ] gdm3
[ - ] grub-common
[ - ] hwclock.sh
[ - ] irqbalance
[ + ] kerneloops
[ - ] keyboard-setup.sh
[ + ] kmod
[ + ] mysql
[ + ] named
[ + ] network-manager
[ + ] openvpn
[ - ] plymouth
[ - ] plymouth-log
[ - ] pppd-dns
[ + ] procs
[ + ] proftpd
[ - ] pulseaudio-enable-autospawn
[ - ] rsync
[ + ] rsyslog
[ - ] saned
[ - ] speech-dispatcher
[ - ] spice-vdagent
[ + ] udev
[ + ] ufw
[ + ] unattended-upgrades
[ - ] uuid
[ - ] vsftpd
[ + ] whoopsie
```

Ahora vamos a meter un archivo de zona para marisma.local. Para ello debemos modificar el archivo /etc/bind/named.conf.local para dejarlo como en la imagen, indicando específicamente el tipo “maestro” y el archivo/ruta:

```
$ sudo /etc/bind/named.conf.local
```

```
pablo@ns1: ~/Escritorio
GNU nano 4.8 /etc/bind/named.conf.local Modificado
//
// Do any local configuration here
//
// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";

zone "marisma.local" {
    type master;
    file "/etc/bind/db.marisma.local";
};
```

Por facilidad de trabajo, podemos copiar el archivo que nos interesa para luego modificarlo, en lugar de crear uno nuevo a mano:

```
$ sudo cp /etc/bind/db.local etc/bind/db.marisma.local
```

```
pablo@ns1:~/Escritorio$ sudo cp /etc/bind/db.local /etc/bind/db.marisma.local
```

Ahora continuamos modificando el nuevo archivo:

```
$ sudo nano etc/bind/db.marisma.local
```

```
pablo@ns1: ~/Escritorio
GNU nano 4.8 /etc/bind/db.marisma.local Modificado
;
; BIND data file for local loopback interface
;
$TTL      604800
@         IN      SOA      ns.marisma.local. root.marisma.local. (
; Serial
                        2      ; Refresh
                        604800 ; Retry
                        86400  ; Expire
                        2419200 ; Negative Cache TTL
                        604800 )
;
@         IN      NS       ns.marisma.local.
ns        IN      A        10.6.1.126
@         IN      A        10.6.1.126
www       IN      A        10.6.1.126
```

Esto representa la zona. En la tabla que aparece en la parte inferior izquierda se añadirían nuevas líneas para futuras máquinas y/o servicios.

Como siempre que modificamos archivos de configuración, reiniciamos el servicio:

```
$ sudo service bind9 restart
```

Podemos comprobar si funciona con el comando “dig”:

```
$ dig @10.6.1.126 google.com
```

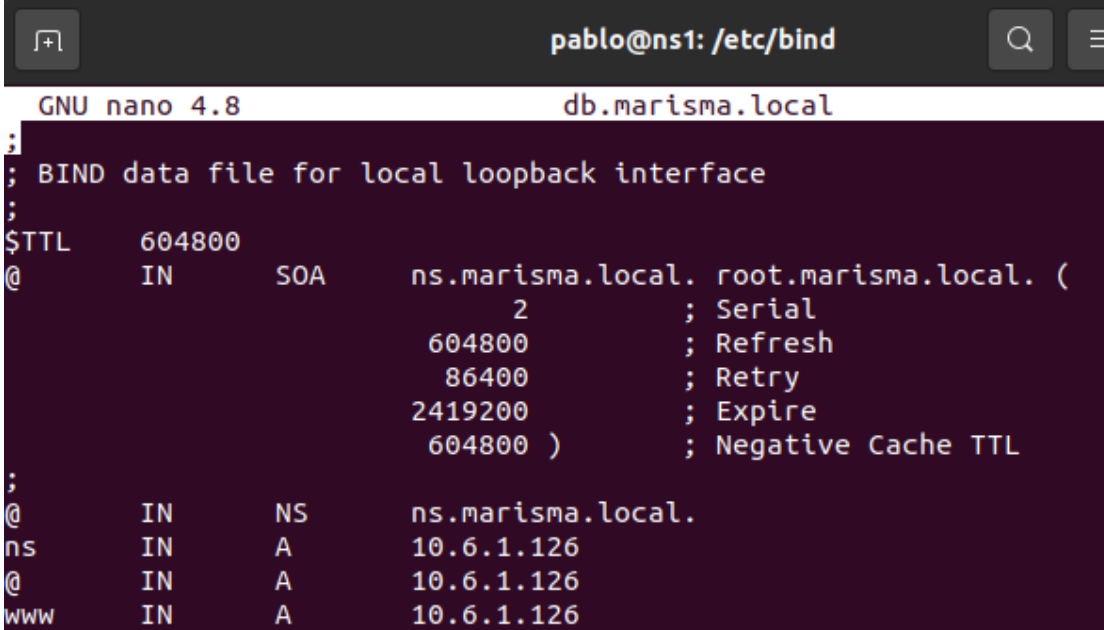
```
$ dig @ipdenuestroservidor google.com
```

Y luego desde Windows con el comando nslookup:

```
$ nslookup google.com 10.6.1.126
```

Después de lo anterior ya podemos intentar agregar un subdominio. En las siguientes imágenes se muestra el archivo db.marisma.local antes y después de añadirlo:

Pre



```
pablo@ns1: /etc/bind
GNU nano 4.8 db.marisma.local
;
; BIND data file for local loopback interface
;
$TTL      604800
@         IN      SOA      ns.marisma.local. root.marisma.local. (
                                2      ; Serial
                                604800 ; Refresh
                                86400  ; Retry
                                2419200 ; Expire
                                604800 ) ; Negative Cache TTL
;
@         IN      NS       ns.marisma.local.
ns        IN      A        10.6.1.126
@         IN      A        10.6.1.126
www       IN      A        10.6.1.126
```

Post

```
pablo@ns1: /etc/bind
GNU nano 4.8 db.marisma.local
;
; BIND data file for local loopback interface
;
$TTL      604800
@         IN      SOA      ns.marisma.local. root.marisma.local. (
                        2      ; Serial
                        604800  ; Refresh
                        86400   ; Retry
                        2419200 ; Expire
                        604800 ) ; Negative Cache TTL
;
@         IN      NS       ns.marisma.local.
ns        IN      A        10.6.1.126
@         IN      A        10.6.1.126
www       IN      A        10.6.1.126
; definición de subdominio
$ORIGIN pepe.marisma.local.
@         IN      A        10.6.1.126
www       IN      A        10.6.1.126
```

Y tras ello reiniciamos el servicio con “`sudo service bind9 restart`” para terminar.

## PROFTPD

En esta parte vamos a instalar Proftpd. Comenzamos con el comando de instalación y la modificación de su archivo de configuración `/etc/proftpd/proftpd.conf`:

```
$ sudo apt-get install proftpd
```

```
pablo@pablo-Standard-PC-i440FX-PIIX-1996:~/Escritorio$ sudo apt-get install proftpd
[sudo] contraseña para pablo:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Nota, seleccionando «proftpd-basic» en lugar de «proftpd»
Paquetes sugeridos:
  openbsd-inetd | inet-superserver proftpd-mod-ldap proftpd-mod-mysql
  proftpd-mod-odbc proftpd-mod-pgsql proftpd-mod-sqlite proftpd-mod-geoip
  proftpd-mod-snmp
Se instalarán los siguientes paquetes NUEVOS:
  proftpd-basic
0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 84 no actualizados.
Se necesita descargar 0 B/2.172 kB de archivos.
Se utilizarán 5.374 kB de espacio de disco adicional después de esta operación.
Seleccionando el paquete proftpd-basic previamente no seleccionado.
(Leyendo la base de datos ... 185942 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../proftpd-basic_1.3.6c-2_amd64.deb ...
Desempaquetando proftpd-basic (1.3.6c-2) ...
Configurando proftpd-basic (1.3.6c-2) ...
usermod: sin cambios
Replacing config file /etc/proftpd/proftpd.conf with new version
Cannot start proftpd, please check syntax of your configuration file /etc/proftpd/proftpd.c
Procesando disparadores para man-db (2.9.1-1) ...
Procesando disparadores para systemd (245.4-4ubuntu3.15) ...
```



```
$ sudo /etc/proftpd/proftpd.conf
```

Cambiaremos el **ServerName** a nuestro **host name**, y descomentaremos la línea (DefaultRoot) que limita a los usuarios a su directorio **home**. A su vez incluiremos/activaremos la directiva **“AllowOverwrite”** dejándola en **“on”**. En las siguientes imágenes se muestra el archivo entero:

```
# /etc/proftpd/proftpd.conf -- This is a basic ProFTPd configuration file.
# To really apply changes, reload proftpd after modifications, if
# it runs in daemon mode. It is not required in inetd/xinetd mode.
#

# Includes DSO modules
Include /etc/proftpd/modules.conf

# Set off to disable IPv6 support which is annoying on IPv4 only boxes.
UseIPv6                                     off
# If set on you can experience a longer connection delay in many cases.
IdentLookups                               off

# Set to inetd only if you would run proftpd by inetd/xinetd.
# Read README.Debian for more information on proper configuration.
ServerName                                 ftp.marisma.local
ServerType                                 standalone
DeferWelcome                               off

MultilineRFC2228                           on
DefaultServer                             on
ShowSymlinks                              on

TimeoutNoTransfer                          600
TimeoutStalled                             600
TimeoutIdle                               1200

DisplayLogin                               welcome.msg
DisplayChdir                               .message true
ListOptions                               "-l"

DenyFilter                                 \*.*/*

# Use this to jail all users in their homes
DefaultRoot                                ~
ServerIdent on "FTP Server ready."

# Users require a valid shell listed in /etc/shells to login.
# Use this directive to release that constrain.
RequireValidShell                          off

# Port 21 is the standard FTP port.
```

ServerIdent on "FTP Server ready."

# Users require a valid shell listed in /etc/shells to login.  
# Use this directive to release that constrain.

RequireValidShell off

# Port 21 is the standard FTP port.

Port 21

# In some cases you have to specify passive ports range to by-pass  
# firewall limitations. Ephemeral ports can be used for that, but  
# feel free to use a more narrow range.

# PassivePorts 49152 65534

# If your host was NATted, this option is useful in order to  
# allow passive tranfers to work. You have to use your public  
# address and opening the passive ports used on your firewall as well.

# MasqueradeAddress 1.2.3.4

# This is useful for masquerading address with dynamic IPs:  
# refresh any configured MasqueradeAddress directives every 8 hours

<IfModule mod\_dynmasq.c>

# DynMasqRefresh 28800

</IfModule>

# To prevent DoS attacks, set the maximum number of child processes  
# to 30. If you need to allow more than 30 concurrent connections  
# at once, simply increase this value. Note that this ONLY works  
# in standalone mode, in inetd mode you should use an inetd server  
# that allows you to limit maximum number of processes per service  
# (such as xinetd)

MaxInstances 30

# Set the user and group that the server normally runs at.

User nobody

Group nogroup

# Umask 022 is a good standard umask to prevent new files and dirs  
# (second parm) from being group and world writable.

Umask 022 022

# Normally, we want files to be overwriteable.

<Directory />

AllowOverwrite on

</Directory>

```
# Normally, we want files to be overwriteable.
<Directory />
    AllowOverwrite          on
</Directory>

# Uncomment this if you are using NIS or LDAP via NSS to retrieve passwords:
# PersistentPasswd          off

# This is required to use both PAM-based authentication and local passwords
# AuthOrder                  mod_auth_pam.c* mod_auth_unix.c

# Be warned: use of this directive impacts CPU average load!
# Uncomment this if you like to see progress and transfer rate with ftpwho
# in downloads. That is not needed for uploads rates.
#
# UseSendFile                off

TransferLog /var/log/proftpd/xferlog
SystemLog   /var/log/proftpd/proftpd.log

# Logging onto /var/log/lastlog is enabled but set to off by default
#UseLastlog on

# In order to keep log file dates consistent after chroot, use timezone info
# from /etc/localtime.  If this is not set, and proftpd is configured to
# chroot (e.g. DefaultRoot or <Anonymous>), it will use the non-daylight
# savings timezone regardless of whether DST is in effect.
#SetEnv TZ :/etc/localtime

<IfModule mod_quotatab.c>
QuotaEngine off
</IfModule>

<IfModule mod_ratio.c>
Ratios off
</IfModule>

# Delay engine reduces impact of the so-called Timing Attack described in
# http://www.securityfocus.com/bid/11430/discuss
# It is on by default.
<IfModule mod_delay.c>
DelayEngine on
</IfModule>
```

```
</IfModule>

<IfModule mod_ctrls.c>
ControlsEngine      off
ControlsMaxClients  2
ControlsLog         /var/log/proftpd/controls.log
ControlsInterval    5
ControlsSocket      /var/run/proftpd/proftpd.sock
</IfModule>

<IfModule mod_ctrls_admin.c>
AdminControlsEngine off
</IfModule>

#
# Alternative authentication frameworks
#
#Include /etc/proftpd/ldap.conf
#Include /etc/proftpd/sql.conf

#
# This is used for FTPS connections
#
Include /etc/proftpd/tls.conf

#
# Useful to keep VirtualHost/VirtualRoot directives separated
#
#Include /etc/proftpd/virtuals.conf

# A basic anonymous configuration, no upload directories.

<Anonymous ~ftp>
    User ftp
    Group ftp
#   # We want clients to be able to login with "anonymous" as well as "ftp"
    UserAlias anonymous ftp
#   # Cosmetic changes, all files belongs to ftp user
#   DirFakeUser on ftp
#   DirFakeGroup on ftp
#
#   RequireValidShell off
#
#   # Limit the maximum number of anonymous logins
```

```
#
# # We want 'welcome.msg' displayed at login, and '.message' displayed
# # in each newly chdired directory.
DisplayLogin          welcome.msg
DisplayChdir           .message
#
# # Limit WRITE everywhere in the anonymous chroot
# <Directory *>
#   <Limit WRITE>
#     DenyAll
#   </Limit>
# </Directory>
#
# # Uncomment this if you're brave.
# # <Directory incoming>
# #   # Umask 022 is a good standard umask to prevent new files and dirs
# #   # (second parm) from being group and world writable.
# #   Umask          022  022
# #   <Limit READ WRITE>
# #     DenyAll
# #   </Limit>
# #   <Limit STOR>
# #     AllowAll
# #   </Limit>
# # </Directory>
#
</Anonymous>

# Include other custom configuration files
# !! Please note, that this statement will read /all/ file from this subdir,
# i.e. backup files created by your editor, too !!!
# Eventually create file patterns like this: /etc/proftpd/conf.d/*.conf
#
Include /etc/proftpd/conf.d/
```

## TLS

Ahora procedemos con TLS, pero antes necesitamos instalar openssl, así que usamos el comando:

```
$ sudo apt-get install openssl
```

```
usuario@usuario-VirtualBox:/etc/proftpd/ssl$ sudo apt-get install proftpd openssl
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Nota, seleccionando «proftpd-basic» en lugar de «proftpd»
proftpd-basic ya está en su versión más reciente (1.3.6c-2).
openssl ya está en su versión más reciente (1.1.1f-1ubuntu2.10).
fijado openssl como instalado manualmente.
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 5 no actualizados.
```

Por seguridad, modificaremos ligeramente el archivo **proftpd.conf** para evitar que un posible atacante pueda ver información sobre el software del servidor FTP o la versión del sistema operativo. También indicaremos que nuestros usuarios de FTP inicien en su directorio **home**.

```

ServerName                ftp.marisma.local
ServerType                 standalone
DeferWelcome               off

MultilineRFC2228           on
DefaultServer              on
ShowSymlinks               on

TimeoutNoTransfer          600
TimeoutStalled             600
TimeoutIdle                1200

DisplayLogin               welcome.msg
DisplayChdir                .message true
ListOptions                 "-l"

DenyFilter                 \ *.* /

# Use this to jail all users in their homes
DefaultRoot                 ~
ServerIdent on "FTP Server ready."

# Users require a valid shell listed in /etc/shells to login.
# Use this directive to release that constrain.
RequireValidShell           off

# Port 21 is the standard FTP port.
Port                        21

# In some cases you have to specify passive ports range to by-pass
# firewall limitations. Ephemeral ports can be used for that, but
# feel free to use a more narrow range.
# PassivePorts              49152 65534

# If your host was NATted, this option is useful in order to
# allow passive tranfers to work. You have to use your public
# address and opening the passive ports used on your firewall as well.
# MasqueradeAddress         1.2.3.4

# This is useful for masquerading address with dynamic IPs:
# refresh any configured MasqueradeAddress directives every 8 hours
<IfModule mod_dynmasq.c>
# DynMasqRefresh 28800
</IfModule>

```

A continuación necesitaremos crear un directorio para que albergue el certificado SSL, y generaremos el mismo con los siguientes comandos:

```
$ sudo mkdir /etc/proftpd/ssl
```

```
usuario@usuario-VirtualBox:~/Escritorio$ sudo mkdir /etc/proftpd/ssl
```

```
$ sudo openssl req -new -x509 -days 365 -nodes -out /etc/proftpd/ssl/proftpd.cert.pem -
keyout /etc/proftpd/ssl/proftpd.key.pem
```

Nos pedirá numerosos datos como el nombre del país en código de dos letras o la localidad:



```

usuario@usuario-VirtualBox:/etc/proftpd/ssl$ sudo openssl req -new -x509 -days 365 -nodes -out /etc/proftpd/ssl/proftpd.cert.pem -keyout /etc/proftpd/ssl/proftpd.key.pem
Generating a RSA private key
.....+++++
.....+++++
Writing new private key to '/etc/proftpd/ssl/proftpd.key.pem'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:ES
State or Province Name (full name) [Some-State]:ESPAÑAITA
Locality Name (eg, city) []:HUELVA
Organization Name (eg, company) [Internet Widgits Pty Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:
Email Address []:meloeestoyinventando@yahoo.com
-----

```

Acto seguido modificamos los permisos del certificado:

```
$ sudo chmod 600 /etc/proftpd/ssl/proftpd.*
```

```

usuario@usuario-VirtualBox:/etc/proftpd/ssl$ sudo chmod 600 /etc/proftpd/ssl/proftpd.*

```

Y volvemos a abrir el archivo proftpd.conf añadiendo la siguiente línea:

```
$ sudo nano /etc/proftpd/proftpd.conf
```

*Include /etc/proftpd/tls.conf*

```

<IfModule mod_ctrls.c>
ControlsEngine      off
ControlsMaxClients  2
ControlsLog         /var/log/proftpd/controls.log
ControlsInterval    5
ControlsSocket      /var/run/proftpd/proftpd.sock
</IfModule>

<IfModule mod_ctrls_admin.c>
AdminControlsEngine off
</IfModule>

#
# Alternative authentication frameworks
#
#Include /etc/proftpd/ldap.conf
#Include /etc/proftpd/sql.conf

#
# This is used for FTPS connections
#
Include /etc/proftpd/tls.conf

#
# Useful to keep VirtualHost/VirtualRoot directives separate
#
#Include /etc/proftpd/virtuals.conf

# A basic anonymous configuration, no upload directories.

```

Ahora nos dirigimos al archivo que le hemos indicado, **tls.conf**:

```
$ sudo nano /etc/proftpd/tls.conf
```

```
#
# Proftpd sample configuration for FTPS connections.
#
# Note that FTPS impose some limitations in NAT traversing.
# See http://www.castaglia.org/proftpd/doc/contrib/ProFTPD-mini-HOWTO-TLS.html
# for more information.
#
<IfModule mod_tls.c>
  TLSEngine on
  TLSLog /var/log/proftpd/tls.log
  TLSProtocol SSLv23
  TLSCipherSuite AES128+EECDH:AES128+EDH
  TLSOptions NoCertRequest AllowClientRenegotiations

#TLSProtocol SSLv23
#
# Server SSL certificate. You can generate a self-signed certificate using
# a command like:
#
# openssl req -x509 -newkey rsa:1024 \
# -keyout /etc/ssl/private/proftpd.key -out /etc/ssl/certs/proftpd.crt \
# -nodes -days 365
#
# The proftpd.key file must be readable by root only. The other file can be
# readable by anyone.
#
# chmod 0600 /etc/ssl/private/proftpd.key
# chmod 0640 /etc/ssl/private/proftpd.key
#
  TLSRSACertificateFile /etc/proftpd/ssl/proftpd.cert.pem
  TLSRSACertificateKeyFile /etc/proftpd/ssl/proftpd.key.pem
#
# CA the server trusts...
#TLSCACertificateFile /etc/ssl/certs/CA.pem
# ...or avoid CA cert and be verbose
#TLSOptions NoCertRequest EnableDiags
# ... or the same with relaxed session use for some clients (e.g. FireFtp)
#TLSOptions NoCertRequest EnableDiags NoSessionReuseRequired
#
#
# Per default drop connection if client tries to start a renegotiate
# This is a fix for CVE-2009-3555 but could break some clients.
```



```
# -keyout /etc/ssl/private/proftpd.key -out /etc/ssl/certs/proftpd.crt \
# -nodes -days 365
#
# The proftpd.key file must be readable by root only. The other file can be
# readable by anyone.
#
# chmod 0600 /etc/ssl/private/proftpd.key
# chmod 0640 /etc/ssl/private/proftpd.key
#
TLRSACertificateFile /etc/proftpd/ssl/proftpd.cert.pem
TLRSACertificateKeyFile /etc/proftpd/ssl/proftpd.key.pem
#
# CA the server trusts...
#TLSCACertificateFile /etc/ssl/certs/CA.pem
# ...or avoid CA cert and be verbose
#TLSOptions NoCertRequest EnableDiags
# ... or the same with relaxed session use for some clients (e.g. FireFtp)
#TLSOptions NoCertRequest EnableDiags NoSessionReuseRequired
#
#
# Per default drop connection if client tries to start a renegotiate
# This is a fix for CVE-2009-3555 but could break some clients.
#
#TLSOptions AllowClientRenegotiations
#
# Authenticate clients that want to use FTP over TLS?
#
TLSVerifyClient off
#
# Are clients required to use FTP over TLS when talking to this server?
#
TLSRequired on
#
# Allow SSL/TLS renegotiations when the client requests them, but
# do not force the renegotiations. Some clients do not support
# SSL/TLS renegotiations; when mod_tls forces a renegotiation, these
# clients will close the data connection, or there will be a timeout
# on an idle data connection.

RequireValidShell no
#TLSRenegotiate required off
</IfModule>
```

Al tener “TLSRequired on” solo se permiten conexiones TLS. Finalmente, podemos reiniciar el servicio, y ya estaría listo.

```
$ systemctl restart proftpd.service
```

## SSH

Para añadir la seguridad SSH a nuestras conexiones, vamos a comenzar instalándolo, mediante el comando:

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

Durante la instalación nos pedirá confirmación para usar algo de espacio del disco duro para el proceso, a lo cual como siempre accedemos:

```
pablo@ns1:~/Escritorio$ sudo apt-get install openssh-server
[sudo] contraseña para pablo:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  ncurses-term openssh-client openssh-sftp-server ssh-import-id
Paquetes sugeridos:
  keychain libpam-ssh monkeysphere ssh-askpass molly-guard
Se instalarán los siguientes paquetes NUEVOS:
  ncurses-term openssh-server openssh-sftp-server ssh-import-id
Se actualizarán los siguientes paquetes:
  openssh-client
1 actualizados, 4 nuevos se instalarán, 0 para eliminar y 126 no actualizados.
Se necesita descargar 688 kB/1.359 kB de archivos.
Se utilizarán 6.010 kB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s
Des:1 http://es.archive.ubuntu.com/ubuntu focal/main amd64 ncurses-term all 6.2-0ubuntu2 [249 kB]
Des:2 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 openssh-sftp-server amd64 1:8.2p1-4ubuntu0.4 [51,5 kB]
Des:3 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 openssh-server amd64 1:8.2p1-4ubuntu0.4 [377 kB]
Des:4 http://es.archive.ubuntu.com/ubuntu focal/main amd64 ssh-import-id all 5.1
```

Esto sería suficiente para poder establecer conexiones mediante SSH, pero lo ideal es continuar configurandolo para mejorar la seguridad de las mismas (razón original de SSH). Para ello vamos a modificar el archivo `/etc/ssh/sshd_config`:

```
$ sudo nano /etc/ssh/sshd_config
```

```
pablo@ns1:~/Escritorio$ sudo nano /etc/ssh/sshd_config
```

Dentro del archivo de configuración vamos a modificar por un lado el puerto usado por defecto del 22 a otro cualquier otro comprendido en el rango 1025-65536.

Por otro, podemos limitar los usuarios que pueden conectarse a los indicados con “**AllowUser**”, añadiendo a continuación el nombre del usuario@IP del ordenador. Como es lógico, esto tiene sentido si la IP de cada Pc que queremos que se conecte es estática, y la conocemos. En este ejemplo vamos a omitir esta opción para permitir conexiones sin prefijar las IP/usuarios.

También podemos impedir que accedan remotamente utilizando el superusuario cambiando el parámetro **PermitRootLogin** a “**no**”.

Nuestro archivo quedaría así (aunque comentaremos la línea **AllowUsers** como se indicó previamente):

```

GNU nano 4.8 /etc/ssh/sshd_config
# 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.

Include /etc/ssh/sshd_config.d/*.conf

Port 10000
AllowUsers Usuario3@10.6.0.126

#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:
#LoginGraceTime 2m
PermitRootLogin no
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

```

Guardamos los cambios, cerramos el editor y necesitamos reiniciar el servicio para que se apliquen los cambios de configuración:

```
$ sudo service ssh restart
```

```
pablo@ns1:~/Escritorio$ sudo service ssh restart
```

Intento de conexión desde cmd Windows a la máquina ubuntu (el problema es la contraseña?)

```
$ ssh -p 10000 usuario@10.6.1.126
```

```
C:\Users\USER>ssh -p 10000 usuario@10.6.1.126
The authenticity of host '[10.6.1.126]:10000 ([10.6.1.126]:10000)' can't be established.
ECDSA key fingerprint is SHA256:pZx6M55kYjzEu7gGeQHVPxSJudZx7ssIlTsh64VcXVc.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[10.6.1.126]:10000' (ECDSA) to the list of known hosts.
usuario@10.6.1.126's password:
Permission denied, please try again.
usuario@10.6.1.126's password:
Permission denied, please try again.
usuario@10.6.1.126's password:
usuario@10.6.1.126: Permission denied (publickey,password).
```

```
$ ssh -p 10000 usuario2@10.6.1.126
```

```
C:\Users\USER>ssh -p 10000 usuario2@10.6.1.126
usuario2@10.6.1.126's password:
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.13.0-27-generic x86_64)

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

Se pueden aplicar 131 actualizaciones de forma inmediata.
41 de estas son actualizaciones de seguridad estándares.
Para ver estas actualizaciones adicionales ejecute: apt list --upgradable

Your Hardware Enablement Stack (HWE) is supported until April 2025.

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

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

No mail.
usuario2@ns1:~$
```

## X11 forwarding

Con esto estaría la primera parte completa, por lo que empezamos la siguiente, que es ssh forwarding . Para ello debemos instalar **xauth** si no lo tenemos aún en el sistema. Usamos el comando:

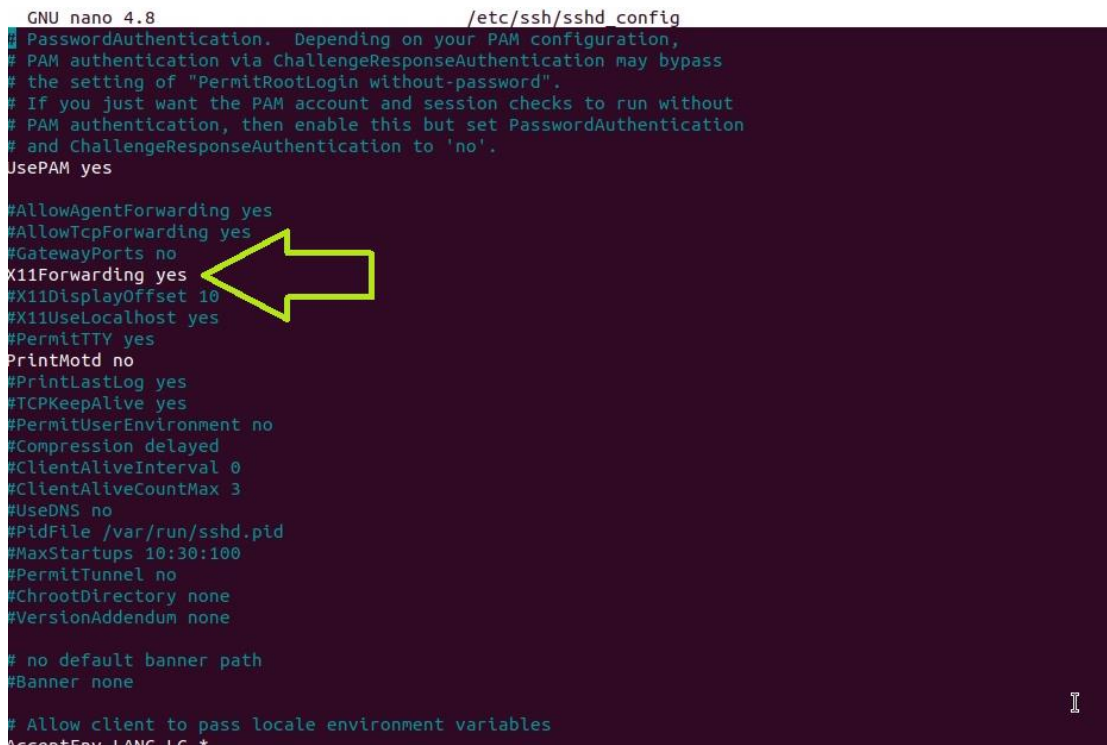
```
$ sudo apt-get install xauth (Ya lo tenía)
```

```
usuario@usuario-VirtualBox:~/Escritorio$ sudo apt-get install xauth
[sudo] contraseña para usuario:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
xauth ya está en su versión más reciente (1:1.1-0ubuntu1).
fijado xauth como instalado manualmente.
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 5 no actualizados.
```

Luego usaremos “nano” para modificar el archivo de configuración **ssh\_config**, añadiendo la siguiente línea para habilitar X11 Forwarding:

```
$ sudo nano /etc/ssh/sshd_config
```

*“X11Forwarding yes”*



```
GNU nano 4.8 /etc/ssh/sshd_config
# PasswordAuthentication. Depending on your PAM configuration,
# PAM authentication via ChallengeResponseAuthentication may bypass
# the setting of "PermitRootLogin without-password".
# If you just want the PAM account and session checks to run without
# PAM authentication, then enable this but set PasswordAuthentication
# and ChallengeResponseAuthentication to 'no'.
UsePAM yes

#AllowAgentForwarding yes
#AllowTcpForwarding yes
#GatewayPorts no
X11Forwarding yes
#X11DisplayOffset 10
#X11UseLocalhost yes
#PermitTTY yes
PrintMotd no
#PrintLastLog yes
#TCPKeepAlive yes
#PermitUserEnvironment no
#Compression delayed
#ClientAliveInterval 0
#ClientAliveCountMax 3
#UseDNS no
#PidFile /var/run/sshd.pid
#MaxStartups 10:30:100
#PermitTunnel no
#ChrootDirectory none
#VersionAddendum none

# no default banner path
#Banner none

# Allow client to pass locale environment variables
AcceptEnv LANG LC_*
```

Reiniciamos sshd para que se apliquen los cambios:

```
$ sudo systemctl restart sshd
```

Y con esto ya estarían configurados X11 Forwarding y SSH. Podemos entonces seguir con **Sftp** cuyo requisito era tener preparado SSH, por lo que podemos confirmar que los pasos anteriores han funcionado conectándonos desde otro Pc (en este caso la máquina host Windows 10) a través de la consola (cmd) con el comando:

```
$ ssh -p 10000 usuario2@10.6.1.126
```

*“\$ ssh -p nºdelpuerto nombredelusuario@IPdelamáquina”*

A continuación nos pedirá la contraseña del usuario con el que nos intentemos conectar.

Para salir podemos introducir:

```
exit
```

## SFTP

Como ha sido posible, procedemos a hacerlo con SFTP. Normalmente sería mediante el siguiente comando:

```
$ sftp usuario2@10.6.1.126
```

Pero en nuestro caso nos dirá que se ha rechazado la conexión, al haber modificado el puerto por defecto durante la configuración de SSH, que es el puerto 22:

```
C:\Users\USER> sftp usuario2@10.6.1.126
ssh: connect to host 10.6.1.126 port 22: Connection refused
Connection closed
```

Así que lo readaptamos como sigue:

```
$ sftp -oPort=10000 usuario2@10.6.1.126
```

*“\$ sftp -oPort=nºdelpuerto nombredelusuario@IPdelamáquina”*

```
C:\Users\USER> sftp -oPort=10000 usuario2@10.6.1.126
Microsoft Windows [Versión 10.0.19042.1526]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\USER> sftp -oPort=10000 usuario2@10.6.1.126
unknown option -- 0
usage: sftp [-46aCfpqrv] [-B buffer_size] [-b batchfile] [-c cipher]
           [-D sftp_server_path] [-F ssh_config] [-i identity_file]
           [-J destination] [-l limit] [-o ssh_option] [-P port]
           [-R num_requests] [-S program] [-s subsystem | sftp_server]
           destination

C:\Users\USER> sftp -oPort=10000 usuario2@10.6.1.126
usuario2@10.6.1.126's password:
Connected to 10.6.1.126.
sftp> pwd
Remote working directory: /home/usuario2
sftp> cd /home
sftp> ls
asun      pablo     samu      usuario2
sftp>
```



## Postfix + Dovecot + Thunderbird

Comenzaremos instalando **postfix**:

```
$ sudo apt install mailutils
```

```
usuario@usuario-VirtualBox:~/Escritorio$ sudo apt install mailutils
[sudo] contraseña para usuario:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  libgsasl7 libkyotocabinet16v5 libmailutils6 libntlm0 mailutils-common postfix
Paquetes sugeridos:
  mailutils-mh mailutils-doc procmail postfix-mysql postfix-pgsql postfix-ldap postfix-pcre postfix-lmdb postfix-sqlite
Se instalarán los siguientes paquetes NUEVOS:
  libgsasl7 libkyotocabinet16v5 libmailutils6 libntlm0 mailutils mailutils-common postfix
0 actualizados, 7 nuevos se instalarán, 0 para eliminar y 5 no actualizados.
Se necesita descargar 2.494 kB de archivos.
Se utilizarán 10,8 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s
Des:1 http://es.archive.ubuntu.com/ubuntu focal-updates/universe amd64 libntlm0 amd64 1.5-2ubuntu0.1 [14,7 kB]
Des:2 http://es.archive.ubuntu.com/ubuntu focal/universe amd64 libgsasl7 amd64 1.8.1-1 [114 kB]
Des:3 http://es.archive.ubuntu.com/ubuntu focal/universe amd64 libkyotocabinet16v5 amd64 1.2.76-4.2build1 [318 kB]
Des:4 http://es.archive.ubuntu.com/ubuntu focal/universe amd64 mailutils-common all 1:3.7-2.1 [272 kB]
Des:5 http://es.archive.ubuntu.com/ubuntu focal/universe amd64 libmailutils6 amd64 1:3.7-2.1 [437 kB]
Des:6 http://es.archive.ubuntu.com/ubuntu focal/universe amd64 mailutils amd64 1:3.7-2.1 [138 kB]
Des:7 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 postfix amd64 3.4.13-0ubuntu1.2 [1.201 kB]
Descargados 2.494 kB en 1s (4.824 kB/s)
```

Aparecerá una pantalla de configuración, pero si no lo hace, se puede usar el siguiente comando, que también sirve para reconfigurarlo:

```
$ sudo dpkg-reconfigure postfix
```

En la pantalla elegiremos “Internet Site”:

Postfix Configuration

Please select the mail server configuration type that best meets your needs.

No configuration:  
Should be chosen to leave the current configuration unchanged.

Internet site:  
Mail is sent and received directly using SMTP.

Internet with smarthost:  
Mail is received directly using SMTP or by running a utility such as fetchmail. Outgoing mail is sent using a smarthost.

Satellite system:  
All mail is sent to another machine, called a 'smarthost', for delivery.

Local only:  
The only delivered mail is the mail for local users. There is no network.

General type of mail configuration:

No configuration

Internet Site

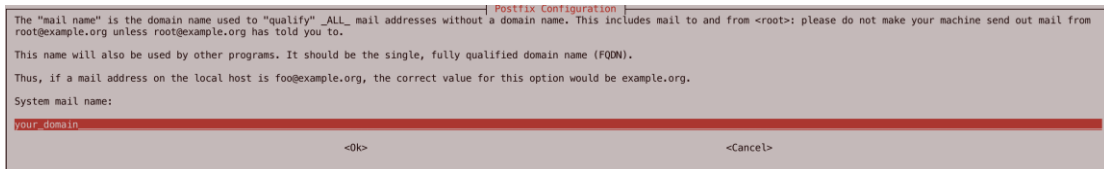
Internet with smarthost

Satellite system

Local only

<Ok><Cancel>

Se nos preguntará por el nombre de mail del sistema (el que asignamos al servidor durante su creación):



The "mail name" is the domain name used to "qualify" \_ALL\_ mail addresses without a domain name. This includes mail to and from <root>; please do not make your machine send out mail from root@example.org unless root@example.org has told you to.

This name will also be used by other programs. It should be the single, fully qualified domain name (FQDN).

Thus, if a mail address on the local host is foo@example.org, the correct value for this option would be example.org.

System mail name:

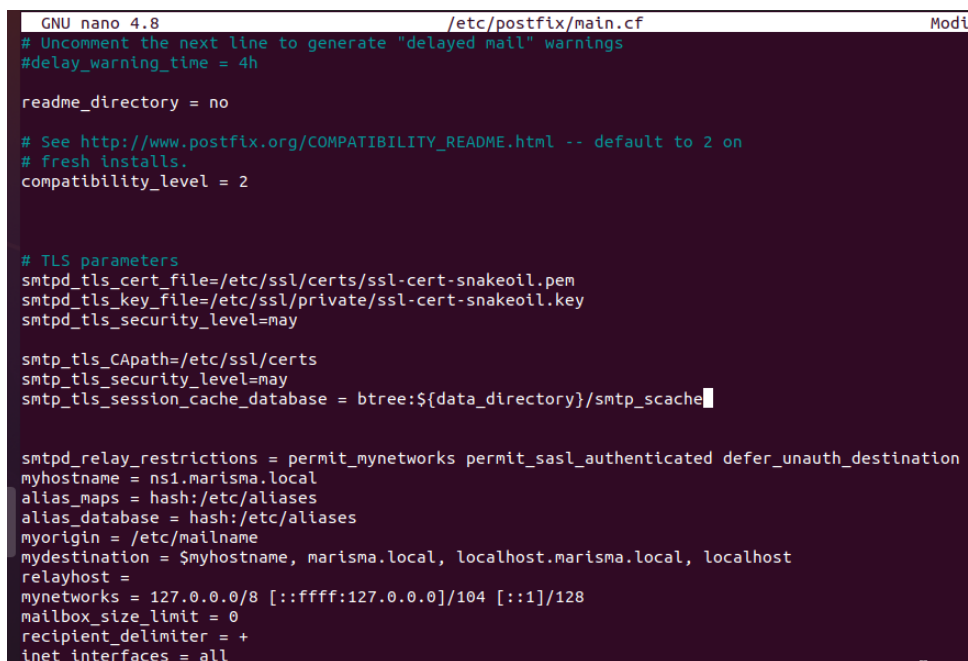
<Ok> <Cancel>

Y ahora, empezaremos con su configuración. Vamos provocar que Postfix solo pueda escuchar en la interfaz **loopback**, para que se comunique internamente. Para ello modificamos el archivo **/etc/postfix/main.cf**. Tenemos también que indicar nuestro dominio (marisma.local) en **mydestination**.

```
$ sudo nano /etc/postfix/main.cf
```

Cambiamos "**inet\_interfaces**" a loopback-only

```
mailbox_size_limit = 0
recipient_delimiter = +
inet_interfaces = loopback-only
inet_protocols = all
```



```
GNU nano 4.8 /etc/postfix/main.cf Modi
# Uncomment the next line to generate "delayed mail" warnings
#delay_warning_time = 4h

readme_directory = no

# See http://www.postfix.org/COMPATIBILITY_README.html -- default to 2 on
# fresh installs.
compatibility_level = 2

# TLS parameters
smtpd_tls_cert_file=/etc/ssl/certs/ssl-cert-snakeoil.pem
smtpd_tls_key_file=/etc/ssl/private/ssl-cert-snakeoil.key
smtpd_tls_security_level=may

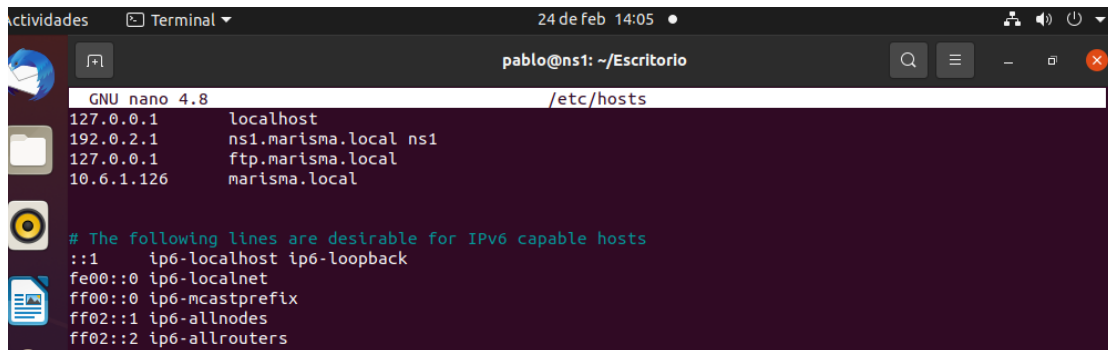
smtp_tls_CApath=/etc/ssl/certs
smtp_tls_security_level=may
smtp_tls_session_cache_database = btree:${data_directory}/smtp_scache

smtpd_relay_restrictions = permit_mynetworks permit_sasl_authenticated defer_unauth_destination
myhostname = ns1.marisma.local
alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases
myorigin = /etc/mailname
mydestination = $myhostname, marisma.local, localhost.marisma.local, localhost
relayhost =
mynetworks = 127.0.0.0/8 [::ffff:127.0.0.0]/104 [::1]/128
mailbox_size_limit = 0
recipient_delimiter = +
inet_interfaces = all
```



En **/etc/hosts** necesitamos tener nuestro dominio con la ip apropiada:

```
$ sudo nano /etc/hosts
```



The screenshot shows a terminal window titled 'pablo@ns1: ~/Escritorio'. The terminal is running the command 'nano /etc/hosts'. The file content is as follows:

```
GNU nano 4.8 /etc/hosts
127.0.0.1    localhost
192.0.2.1    ns1.marisma.local ns1
127.0.0.1    ftp.marisma.local
10.6.1.126   marisma.local

# The following lines are desirable for IPv6 capable hosts
::1         ip6-localhost ip6-loopback
fe00::0     ip6-localnet
ff00::0     ip6-mcastprefix
ff02::1     ip6-allnodes
ff02::2     ip6-allrouters
```

Y por último modificamos nuestro **hostname**:

```
$ sudo nano /etc/hostname
```

```
usuario-Standard-PC-i440FX-
marisma.local
```

Y ya deberíamos poder hacer ping a nuestro servidor con el nombre del dominio:

```
$ ping marisma.local
```

```
pablo@ns1:~/Escritorio$ ping marisma.local
PING marisma.local (10.6.1.126) 56(84) bytes of data.
64 bytes from marisma.local (10.6.1.126): icmp_seq=1 ttl=64 time=0.072 ms
64 bytes from marisma.local (10.6.1.126): icmp_seq=2 ttl=64 time=0.055 ms
64 bytes from marisma.local (10.6.1.126): icmp_seq=3 ttl=64 time=0.053 ms
^C
--- marisma.local ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2032ms
rtt min/avg/max/mdev = 0.053/0.060/0.072/0.008 ms
```

A continuación creamos un usuario (**usuario2**) mediante **\$ sudo adduser usuario2** y le escribimos un mensaje con el comando:

```
$ echo "Cuerpo del mensaje" | mail -s "El asunto del mensake" dirección_de_correo
```

Luego accedemos como usuario2 y comprobamos el buzón de entrada para comprobar que ha llegado con los comandos:

```
$ su usuario2
```

\$ mail

```
es Terminal 24 de feb 14:12
usuario2@ns1: /home/pablo/Escritorio

pablo@ns1:~/Escritorio$ sudo adduser usuario2
Añadiendo el usuario 'usuario2' ...
Añadiendo el nuevo grupo 'usuario2' (1003) ...
Añadiendo el nuevo usuario 'usuario2' (1003) con grupo 'usuario2' ...
Creando el directorio personal '/home/usuario2' ...
Copiando los ficheros desde '/etc/skel' ...
Nueva contraseña:
Vuelva a escribir la nueva contraseña:
passwd: contraseña actualizada correctamente
Cambiano la información de usuario para usuario2
Introduzca el nuevo valor, o presione INTRO para el predeterminado
Nombre completo []:
Número de habitación []:
Teléfono del trabajo []:
Teléfono de casa []:
Otro []:
¿Es correcta la información? [S/n] s
pablo@ns1:~/Escritorio$ echo "bodyyyyyyy" | mail -s "Tremenda subject" usuario2@marisma.local
pablo@ns1:~/Escritorio$ su usuario2
Contraseña:
usuario2@ns1:/home/pablo/Escritorio$ mail
"/var/mail/usuario2": 1 mensaje 1 nuevo
>N 1 Pablo           jue feb 24 14:10 13/468 Tremenda subject
? 1
Return-Path: <pablo@ns1.marisma.local>
X-Original-To: usuario2@marisma.local
Delivered-To: usuario2@marisma.local
Received: by ns1.marisma.local (Postfix, from userid 1000)
        id 26DCCE1DBF; Thu, 24 Feb 2022 14:10:39 +0100 (CET)
Subject: Tremenda subject
To: <usuario2@marisma.local>
X-Mailer: mail (GNU Mailutils 3.7)
Message-Id: <20220224131039.26DCCE1DBF@ns1.marisma.local>
Date: Thu, 24 Feb 2022 14:10:39 +0100 (CET)
```

Aquí se encuentran. Podemos leerlos indicando el número que nos interesa, mostrado a la izquierda.

Proseguiremos ahora con **Dovecot** (incluyendo ya imap y pop3, si no queremos hacerlo posteriormente). Instalamos con:

\$ sudo apt install dovecot-imapd dovecot-pop3d

```
pablo@ns1:~/Escritorio$ sudo apt install dovecot-imapd
[sudo] contraseña para pablo:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  dovecot-core
Paquetes sugeridos:
  dovecot-gssapi dovecot-ldap dovecot-lmtpd dovecot-lucene
  dovecot-managesieved dovecot-mysql dovecot-pgsql dovecot-pop3d dovecot-sieve
  dovecot-solr dovecot-sqlite dovecot-submissiond ntp
Se instalarán los siguientes paquetes NUEVOS:
  dovecot-core dovecot-imapd
0 actualizados, 2 nuevos se instalarán, 0 para eliminar y 127 no actualizados.
Se necesita descargar 3.078 kB de archivos.
Se utilizarán 10,8 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] S
Des:1 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 dovecot-core amd64 1:2.3.7.2-1ubuntu3
.5 [2.918 kB]
Des:2 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 dovecot-imapd amd64 1:2.3.7.2-1ubuntu
3.5 [160 kB]
Descargados 3.078 kB en 0s (7.141 kB/s)
```

Podemos comprobar que está activo con:

```
$ service --status-all
```

Podemos ver los cambios en el archivo de configuración con:

```
$ dovecot -n
```

```
pablo@ns1:~/Escritorio$ dovecot -n
# 2.3.7.2 (3c910f64b): /etc/dovecot/dovecot.conf
# Pigeonhole version 0.5.7.2 ()
# OS: Linux 5.13.0-27-generic x86_64 Ubuntu 20.04.3 LT
# Hostname: ns1.marisma.local
mail_location = mbox:~/mail:INBOX=/var/mail/%u
mail_privileged_group = mail
namespace inbox {
    inbox = yes
    location =
    mailbox Drafts {
        special_use = \Drafts
    }
    mailbox Junk {
        special_use = \Junk
    }
    mailbox Sent {
        special_use = \Sent
    }
    mailbox "Sent Messages" {
        special_use = \Sent
    }
    mailbox Trash {
        special_use = \Trash
    }
    prefix =
}
passdb {
    driver = pam
}
protocols = " imap"
ssl_cert = </etc/dovecot/private/dovecot.pem
ssl_client_ca_dir = /etc/ssl/certs
ssl_dh = # hidden, use -P to show it
ssl_key = # hidden, use -P to show it
userdb {
    driver = passwd
}
```

Como se mantendrá el formato de buzón de **mbox**, que es como viene por defecto, no hay que hacer más. En caso de querer alternarlo al formato Maildir, habría que modificar el archivo **/etc/dovecot/conf.d/10-mail.conf** la directiva **mail\_location**, cambiando de **mail\_location = mbox:~/mail:INBOX=/var/mail/%u** a **mail\_location = maildir:~/Maildir**.

Continuamos con **Thunderbird**, que si no se encuentra ya en nuestro equipo podemos descargarlo e instalarlo con:

```
$ sudo apt install thunderbird
```

```
usuario@usuario-VirtualBox:~/Escritorio$ sudo apt install thunderbird
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
thunderbird ya está en su versión más reciente (1:91.5.0+build1-0ubuntu0.20.04.1).
fijado thunderbird como instalado manualmente.
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 6 no actualizados.
```

Al ejecutarlo por primera vez nos pedirá los datos para crear una cuenta

## Configurar una dirección de correo electrónico existente

Para utilizar su dirección de correo electrónico actual, complete sus credenciales.

Thunderbird buscará automáticamente una configuración de servidor recomendada y que funcione.

Thunderbird buscará automáticamente la configuración de servidor recomendada y que funcione.

Nombre completo

usuario2

Dirección de correo electrónico

usuario2@marisma.local

Contraseña

.....

☒ Recordar contraseña

[Configurar manualmente](#)

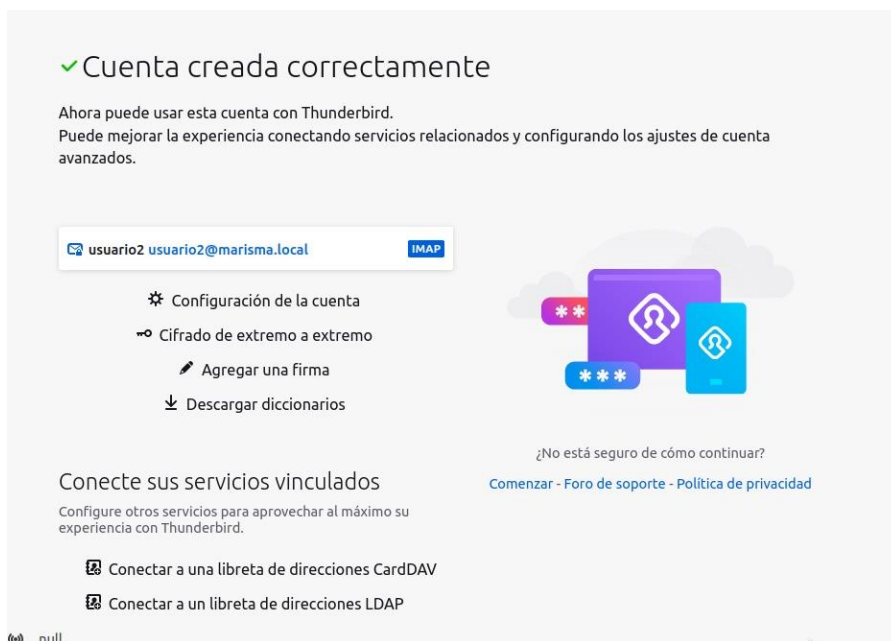
Cancelar

Continuar

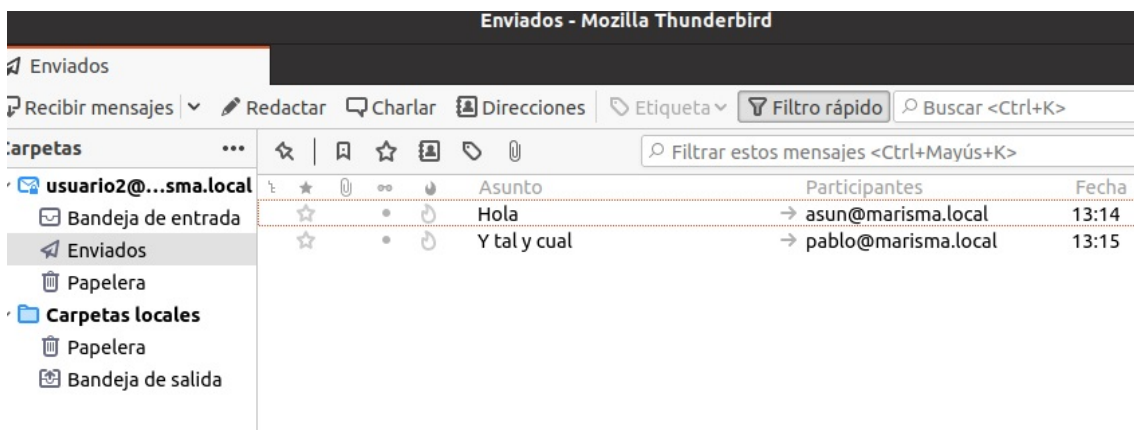
Sus credenciales solo se almacenarán localmente en su computadora.



Es importante recordar que la dirección de correo electrónico es igual a **"nombredeusuario@dominio"**. Al darle a continuar nos salta una ventana para añadir una excepción de seguridad para Thunderbird respecto a la nueva dirección. La confirmamos:



Al loggearnos con la cuenta de usuario2 podemos ver los mensajes enviados hasta ahora:



Podemos ver lo mismo en el correo desde el terminal de Ubuntu:

```
pablo@ns1:~/Escritorio$ mail
"/var/mail/pablo": 2 mensajes 2 nuevos
>N 1 Mail Delivery Syst mié feb 23 10:1 72/2271 Undelivered Mail Returned to Sender
N 2 usuario2 mar mar 1 13:15 20/725 Y tal y cual
? 2
Return-Path: <usuario2@marisma.local>
X-Original-To: pablo@marisma.local
Delivered-To: pablo@marisma.local
Received: from [10.6.1.126] (marisma.local [10.6.1.126])
        by ns1.marisma.local (Postfix) with ESMTPS id 07A27E10D3
        for <pablo@marisma.local>; Tue, 1 Mar 2022 13:15:48 +0100 (CET)
Message-ID: <be7f283a-e51d-7431-fc93-e9d748809837@marisma.local>
Date: Tue, 1 Mar 2022 13:15:48 +0100
MIME-Version: 1.0
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101
        Thunderbird/91.5.0
Content-Language: en-US
To: pablo@marisma.local
From: usuario2 <usuario2@marisma.local>
Subject: Y tal y cual
Content-Type: text/plain; charset=UTF-8; format=flowed
Content-Transfer-Encoding: 7bit

Este es el cuerpo del mensaje

?
```

Si nos movemos por Thunderbird para configurar otra dirección de correo existente, podemos elegir en la configuración del servidor si queremos usar **imap** o **pop3**

### Configurar una dirección de correo electrónico existente

Para utilizar su dirección de correo electrónico actual, complete sus credenciales.  
Thunderbird buscará automáticamente una configuración de servidor recomendada y que funcione.  
Thunderbird buscará automáticamente la configuración de servidor recomendada y que funcione.

Nombre completo  
pablo ⓘ

Dirección de correo electrónico  
pablo@marisma.local ⓘ

Contraseña  
..... ⓘ

☒ Recordar contraseña

**Configuración del servidor**  
**SERVIDOR ENTRANTE**





## Configuración del servidor

### SERVIDOR ENTRANTE

Protocolo:	IMAP
Host	IMAP POP3
Puerto:	
Seguridad de la conexión:	Autodetectar
Método de autenticación:	Autodetectar
Nombre de usuario:	pablo@marisma.local

### SERVIDOR SALIENTE

Host	.marisma.local
Puerto:	
Seguridad de la conexión:	Autodetectar
Método de autenticación:	Autodetectar

### SERVIDOR ENTRANTE

Protocolo:	POP3
Host	.marisma.local
Puerto:	110
Seguridad de la conexión:	STARTTLS
Método de autenticación:	Contraseña normal
Nombre de usuario:	pabo@marisma.local

### SERVIDOR SALIENTE

Host	.marisma.local
Puerto:	25
Seguridad de la conexión:	STARTTLS
Método de autenticación:	Contraseña normal
Nombre de usuario:	pabo@marisma.local

## Scripts

### Crear Subdominio y Usuario:

```
1 #!/bin/bash
2
3 #crear_subdominio.sh nombre_subdominio ip
4 if [ $# -le 1 ];then
5     echo Error!. Introduce subdominio e IP!
6     exit 1;
7 fi
8
9 # Variables
10 USER=$1
11 IP=$2
12 SUB_DOMAIN="${USER}.marisma.local"
13 DOCUMENT="/var/www/html/${USER}"
14 ZONE_FILE="/etc/bind/db.marisma.local"
15
16 adduser -s /bin/bash $USER
17 passwd $user
18
19 echo "Creando carpeta de usuario"
20 mkdir /var/www/html/$USER -m 644
21
22 echo "Actualizando fichero de zona"
23
24 echo "\$ORIGIN ${SUB_DOMAIN}." >>$ZONE_FILE
25 echo "@ IN A   ${IP}" >>$ZONE_FILE
26 echo "www  IN A   ${IP}" >>$ZONE_FILE
27
28 echo "Reiniciar servicios"
29
30 service apache2 reload > /dev/null
31 service bind9 reload > /dev/null
32 service proftpd reload > /dev/null
```

En la primera parte del script podemos ver una comprobación que se cerciora de que se hayan metido los valores necesarios para que el script opere adecuadamente (`$# -le 1`). Luego se indican las variables, y en la línea 16 se hace la creación del usuario. En las líneas 24-26 se crea el subdominio, y por último se reinician los servicios de la 30 a la 32, para que se apliquen los cambios.



## Crear Vhost:

```
1 #!/bin/bash
2 #crear_vhost.sh usuario
3
4 if [ $# -eq 0 ];then
5     echo Error!. Introduce usuario !
6     exit 1;
7 fi
8
9 USER=$1
10 CONF="{USER}.marisma.conf"
11 PATH_AVAILABLE="/etc/apache2/sites-available/${CONF}"
12 PATH_ENABLED="/etc/apache2/sites-enabled/${CONF}"
13 SUB_DOMAIN="{USER}.marisma.local"
14 DOCUMENT_ROOT="/var/www/html/${1}"
15 INDEX="{DOCUMENT_ROOT}/index.html"
16
17 if ! [ -d $DOCUMENT_ROOT ] ; then
18     echo "Creando documento root"
19     mkdir -p "$DOCUMENT_ROOT"
20 fi
21
22 touch $PATH_AVAILABLE
23 if [ -f $PATH_AVAILABLE ] ; then
24     echo "creando fichero de config"
25     echo "<VirtualHost *:80>
26         ServerAdmin admin@${SUB_DOMAIN}
27         ServerName www.${SUB_DOMAIN}
28         DocumentRoot $DOCUMENT_ROOT
29         <Directory $DOCUMENT_ROOT>
30             DirectoryIndex index.html
31             Options Indexes FollowSymLinks MultiViews
32             AllowOverride all
33             Require all granted
34         </Directory>
35         ErrorLog /var/log/apache2/${SUB_DOMAIN}.errorLog.log
36         LogLevel error
37         CustomLog /var/log/apache2/${SUB_DOMAIN}.customLog.log combined
38     </VirtualHost>" >>$PATH_AVAILABLE
39
40     #index.html
41     echo "Creando index.html"
42     echo "<p>Subdominio: ${SUB_DOMAIN}</p>" >>$INDEX
43     echo "<p>usuario: ${USER}</p>" >>$INDEX
44
45     a2ensite $CONF
46 fi
47
```

En la primera sección se descartan intentos que no introduzcan valores, con la línea 4. De la 9 a la 15 se establecen distintas variables y rutas para el host. 17-20 se encargan de comprobar que el usuario no tenga previamente directorio para el alojamiento web, y en ese caso crearla.

De las líneas 22 a 38 el comando **touch** crea la configuración que contenga la info del server, como nombre, el DocumentRoot, la directriz de sobre escritura y el LogLevel. Por último se crea una página index.html básica indicando el subdominio y el nombre del usuario en cuestión en 40-43, y se habilita su archivo de configuración con “a2ensite” (ahora aparecerá el archivo de configuración en sites-enabled).