

TEMA 2

**INSTALACIÓN, CONFIGURACIÓN Y
DOCUMENTACIÓN DEL ENTORNO
DE DESARROLLO Y DEL ENTORNO
DE EXPLOTACIÓN**

EJERCICIOS

REBECA SÁNCHEZ PÉREZ
IES LOS SAUCES

a. USED – UBUNTU SERVER	5
1. Configuración inicial.....	5
1.1 Características de la máquina virtual.....	5
1.2 Instalación.....	7
1.3 Nombre y configuración de red	14
1.4 Actualizar el sistema	16
1.5 Cortafuegos local	16
2. Cuentas de administración	17
3. Apache	18
3.1 ¿Qué es y para qué sirve?.....	18
3.2 Módulos a instalar	18
3.3 Comandos de control del servicio	18
3.4 Instalación y configuración	19
3.5 Estructura de directorios y ficheros de configuración.....	19
3.6 Configuracion de Logs de errores y acceso	20
3.7 Habilitar el uso de .htaccess	21
3.8 Modulo Alias	22
3.9 Modulo Rewrite	23
3.10 HTTPS	23
3.11 Autentificación y autorización	27
3.11.1 Entorno de desarrollo (nuestra maquina virtual).....	27
3.11.1 Entorno de explotación (servidor ionos)	32
4. PHP	34
4.1 Módulos	34
4.2 Instalación de php y configuracion de php.ini.....	34
5. MySQL.....	37
5.1 Instalación y configuracion	37
5.2 PHPMyAdmin	39
6. XDebug	42
6.1 Instalación.....	42
7. Servidores DNS.....	43
7.1 Instalación del servidor DNS	43
7.2 Declaración de la zona directa.....	44
7.3 Declaración de la zona inversa	45

7.4 Comprobación de funcionamiento	45
8. Cuentas de desarrollo y hosting virtual	47
8.1 Creación de operadorweb	47
8.2 Creación del grupo ftpuser y enjaulamiento	47
9. Apache Tomcat	50
9.1 Instalación en Ubuntu.....	50
9.2 Instalación en Windows.....	55
9.3 Creación y despliegue de un proyecto en Windows	59
9.4 Actualizar cuenta de usuario SFTP.....	63
9.5 Utilizar tu propio dominio con Tomcat.....	64
9.6 Tomcat seguro	66
10. LDAP	69
10.1 Configuracion del servidor LDAP	69
10.2 Configuracion del DNS en Windows 10	73
10.3 Instalar OpenLDAP	74
10.4 Comprobación de la instalacion	79
10.5 Crear usuarios y unidades organizativas	81
10.6 Modificar usuarios	83
b. WXED – WINDOWS X	87
1. Nombre y configuración de red	87
2. Cuentas administradoras y cuenta de desarrollador	88
2.1 Conexión SSH desde Windows 10	88
2.2 Cuentas administradoras del sistema operativo	89
3. Navegadores	90
3.1 Google Chrome	90
3.2 Microsoft Edge	90
3.2 Mozilla Firefox.....	90
4. FileZilla	91
4.1 Versión	91
4.2 Conexión a un servidor	91
4.3 Transferencia de archivos.....	92
4.4 Gestión de sitios.....	92
5. NetBeans.....	94
5.1 Instalación y configuración inicial (plugings)	94
5.2 Creación de proyectos, modificación, borrado, prueba	94
5.3 Conexión al servidor remoto SFTP. (Almacenamiento local/almacenamiento remoto)	97

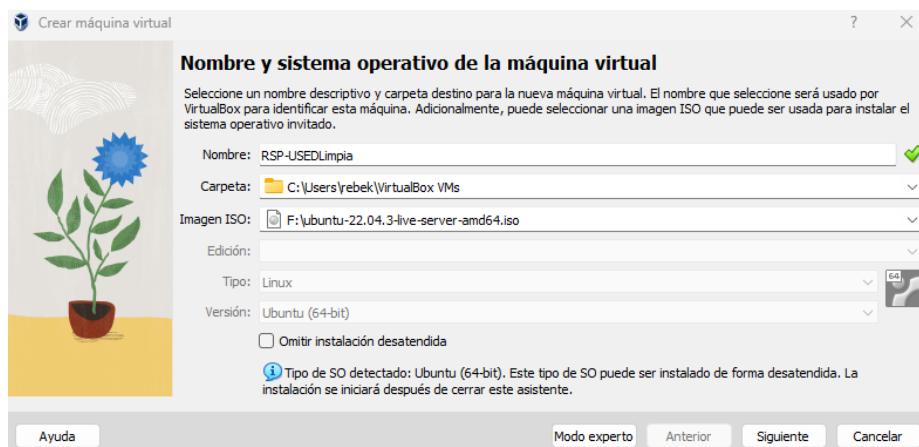
5.4 Administración de la base de datos.....	97
5.5 Depuración - Configuración de la ejecución para la depuración	100
5.6 Paso a explotación	100
5.7 PHP Doc.....	100
5.8 CSS / JS / AJAX / XML / JSON.....	105
5.9 LDAP	105
c. GITHUB – INTERNET	107
1. Creación de token desde GitHub	107
2. Comandos básicos de Git.....	110
3. Creación de un repositorio desde NetBeans	111
4. Clonar un repositorio	115
6. Crear una release	119
7.1 Generar clave SSH.....	121
7.2 Conexión SSH desde NetBeans	123

a. USED – UBUNTU SERVER

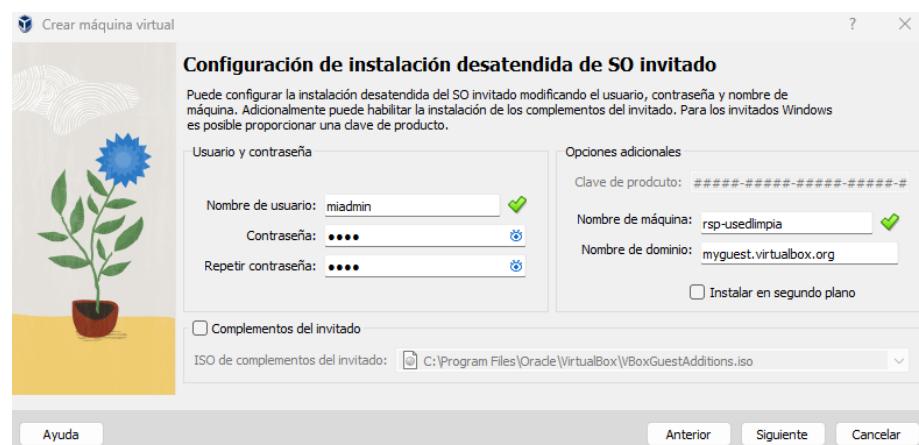
1. Configuración inicial

1.1 Características de la máquina virtual

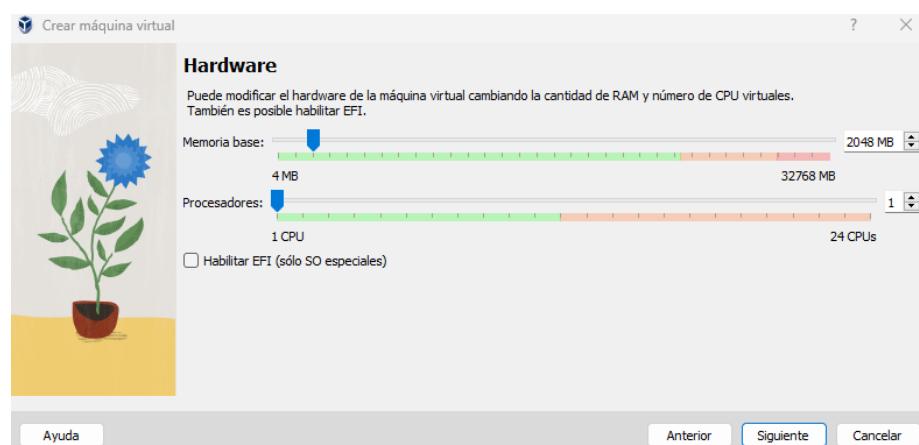
En primer lugar, abriremos el VirtualBox y crearemos una nueva máquina virtual que se llame RSP-USDEDLimpia (Rebeca Sánchez Pérez - Ubuntu Server Entorno de Desarrollo Maquina limpia) con una ISO de un Ubuntu server 22.04.3



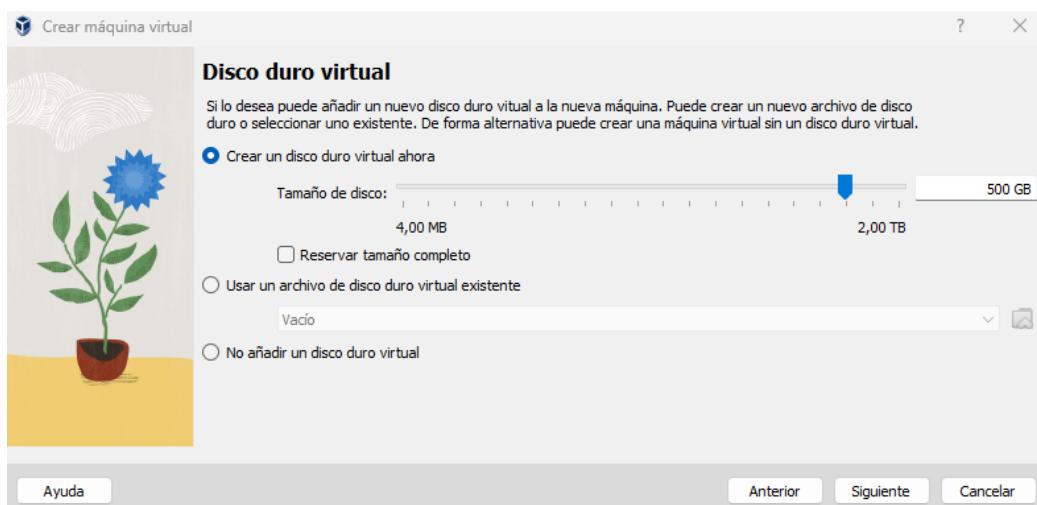
A continuación, elegiremos un nombre de usuario y una contraseña, en mi caso miadmin y paso.



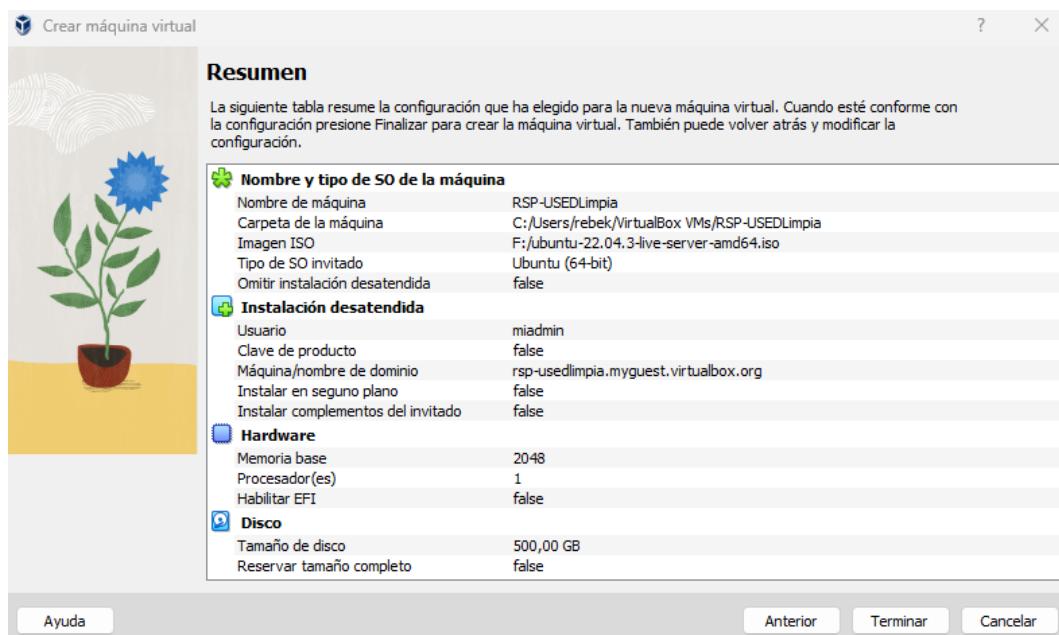
Seguimos con la configuración y reservamos 2048MB para la memoria RAM



Reservamos 500GB para la memoria del Disco Duro

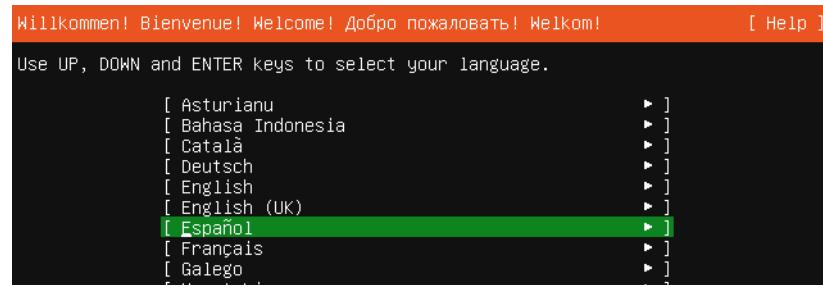


Este será el resumen de la configuración de nuestra máquina de Ubuntu Server:

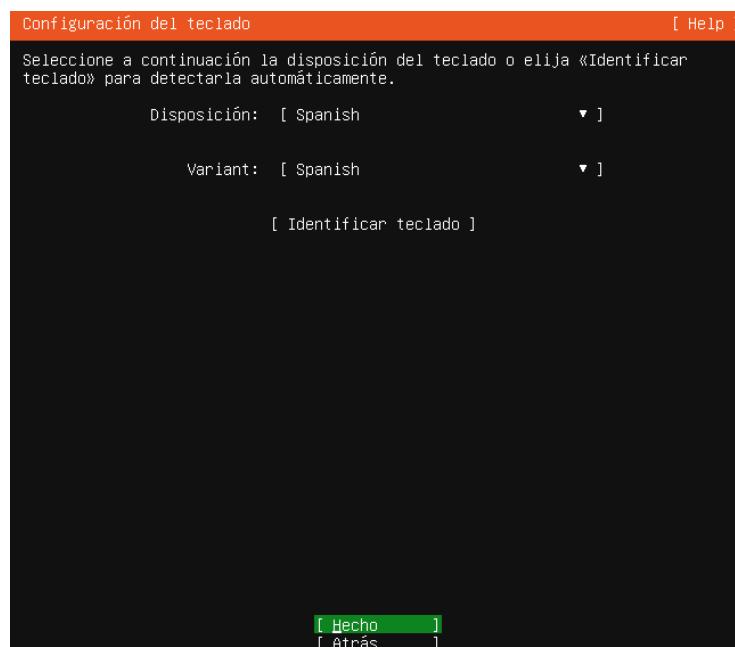
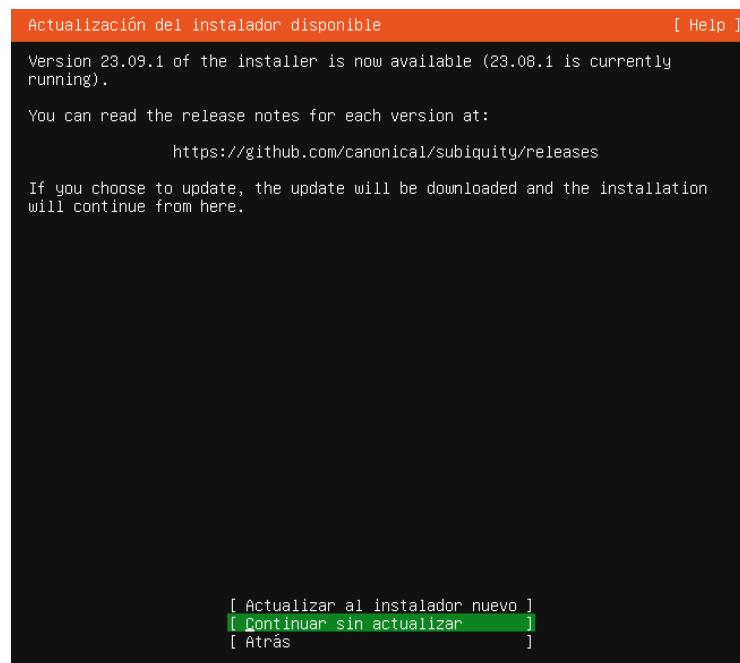


1.2 Instalación

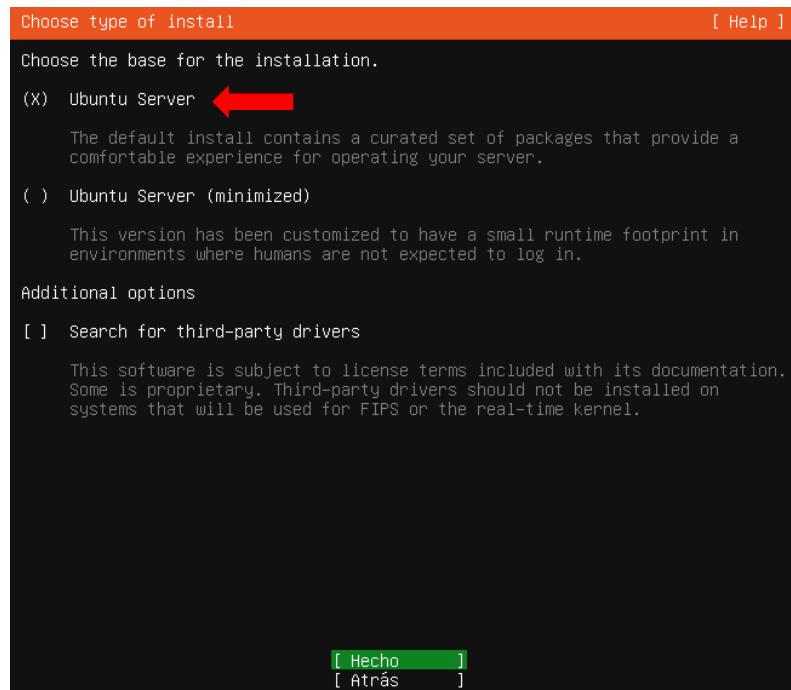
Una vez creada la máquina, la iniciamos y comenzamos con la instalación seleccionando el idioma que queremos (español)



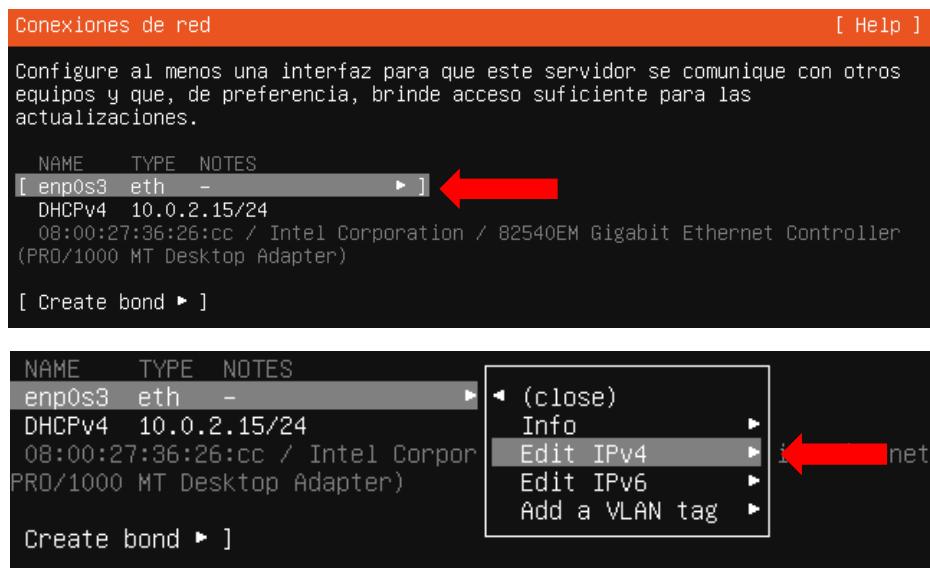
Seguiremos con la instalación sin actualizar a la versión que nos recomiendan



Seleccionamos el Ubuntu Server que aparece en primer lugar

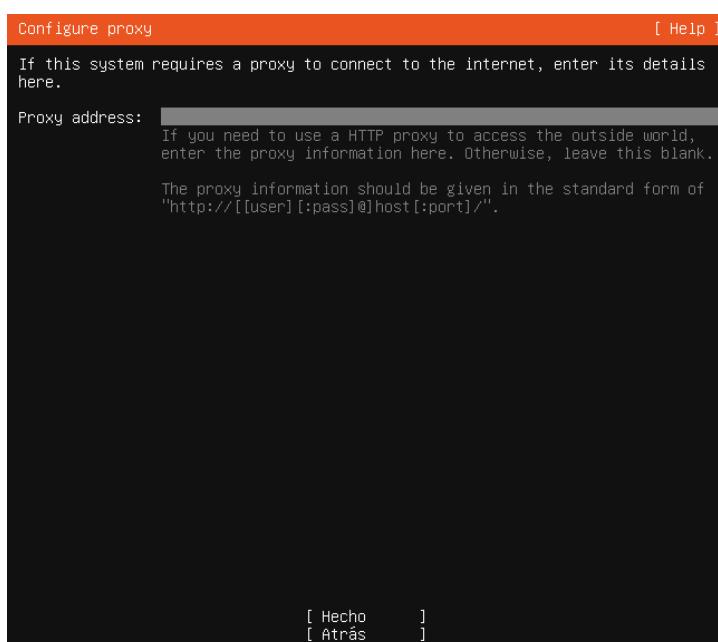


A continuación, realizaremos la configuración de red. Esta configuración cambiará dependiendo de si la instalación la realizamos en casa o en clase ya que no coincidirán las direcciones IP de las maquinas anfitrionas (pertenecen a redes distintas). Configuramos la IPv4 de forma manual dándole la subred de 192.168.1.0/24 (en clase le daríamos [192.168.3.0/24](#)), la dirección IP de 192.168.1.204 (en clase seria la [192.168.3.204](#)) y la puerta de enlace de 192.168.1.1 (y en clase seria la [192.168.3.1](#)). En ambos casos utilizaremos los servidores de DNS de [8.8.8.8](#) de Google.

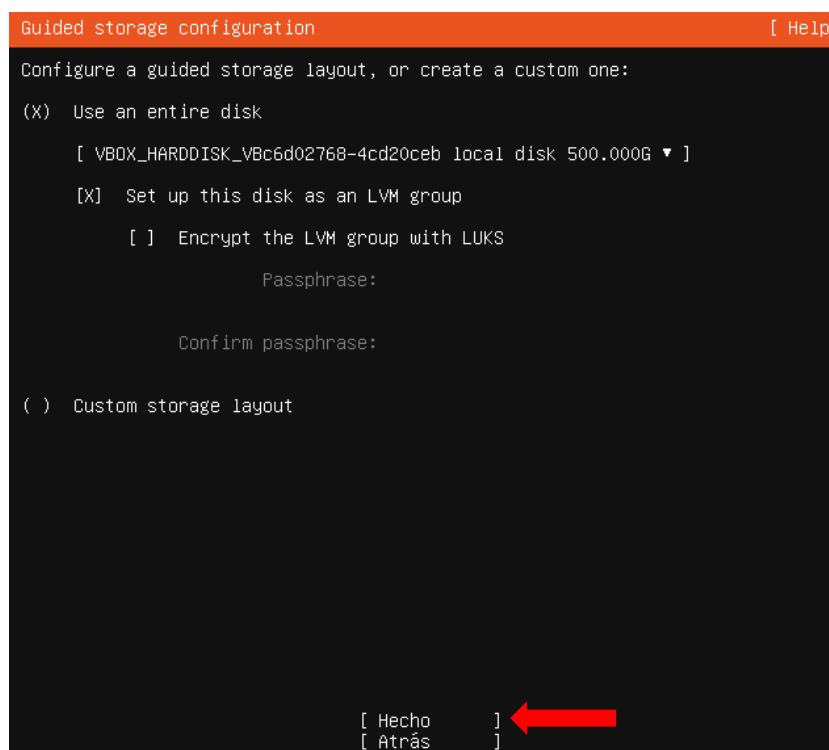
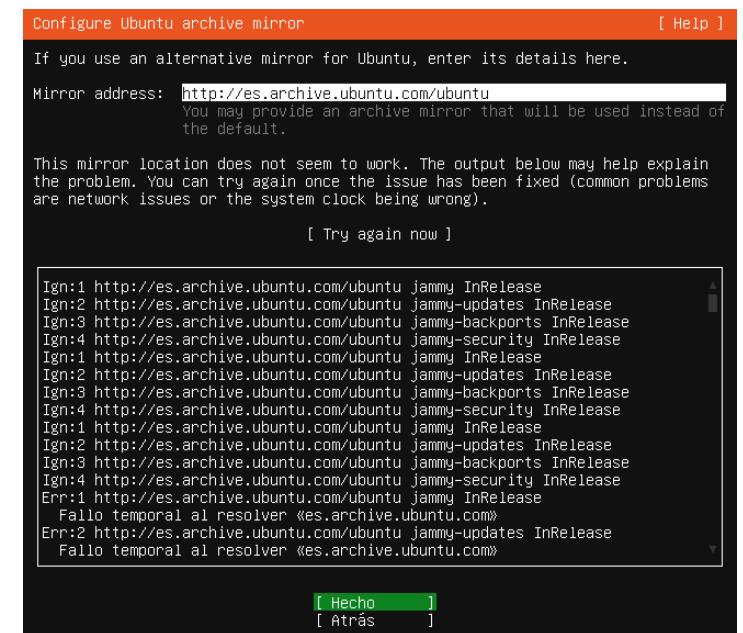




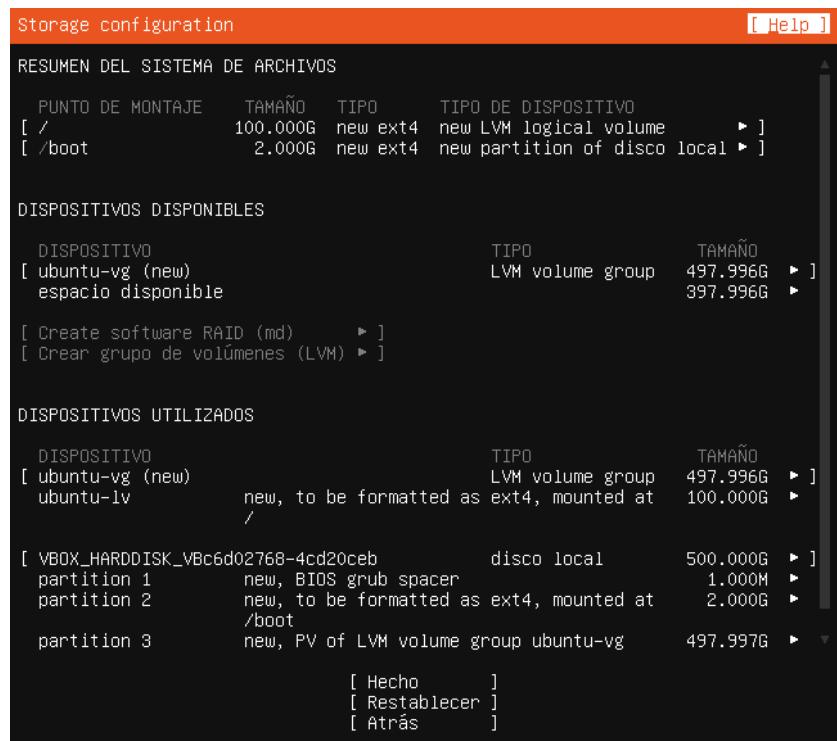
En la siguiente pantalla nos pedirán una dirección proxy, simplemente tenemos que darle a "Hecho"



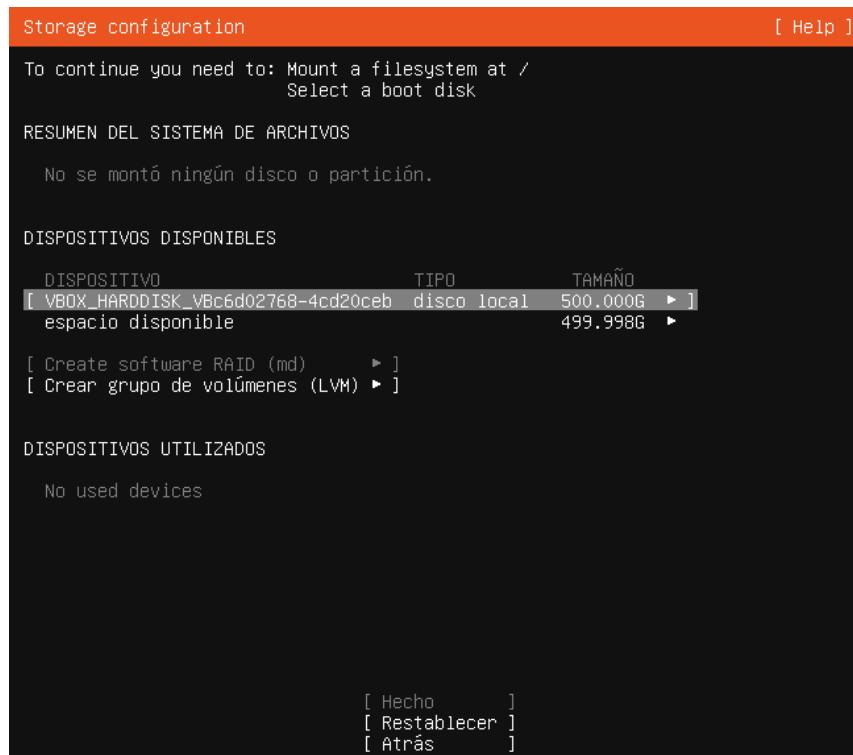
Ahora esperamos a que se instalen las nuevas configuraciones y le damos a “Hecho”



En la siguiente pantalla nos aparece un listado de las particiones que tiene nuestro disco duro. Tendremos que cambiarlas para que cumplan los requisitos de la máquina, para ello seleccionamos “Restablecer”



Una vez pulsado el botón, aparece esta pantalla en la que tendremos que formatear las particiones del disco de forma que el Sistema (/) ocupe 150Gb y los datos (/var) el resto del disco, es decir, 349.997Gb.



```

DISPOSITIVOS DISPONIBLES

DISPOSITIVO          TIPO      TAMAÑO
[ VBOX_HARDDISK_VBc300254e-cd846d61  disco local  500.000G ▶
  espacio disponible   499.998G ▶ [ (close) ] [ Add GPT Partition ▶ ]
[ Create software RAID (md) ▶ ]
[ Crear grupo de volúmenes (LVM) ▶ ]

----- Adding GPT partition to VBOX_HARDDISK_VBc300254e-cd846d61 -----
Size (max 499.998G): 150G

Formato: [ ext4 ] ▾
Mount: [ / ] ▾
[ Crear ] [ Cancelar ]
----- Adding GPT partition to VBOX_HARDDISK_VBc300254e-cd846d61 -----
Size (max 349.997G): [ 349.997G ] ▾

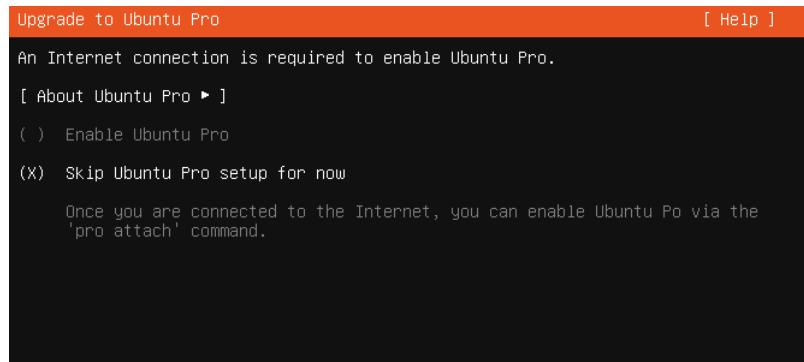
Formato: [ ext4 ] ▾
Mount: [ /var ] ▾
[ Crear ] [ Cancelar ]
----- RESUMEN DEL SISTEMA DE ARCHIVOS -----
PUNTO DE MONTAJE  TAMAÑO  TIPO      TIPO DE DISPOSITIVO
[ /                150.000G new ext4  new partition of disco local ▶ ]
[ /var             349.997G new ext4  new partition of disco local ▶ ]

DISPOSITIVOS DISPONIBLES
No available devices
[ Create software RAID (md) ▶ ]
[ Crear grupo de volúmenes (LVM) ▶ ]

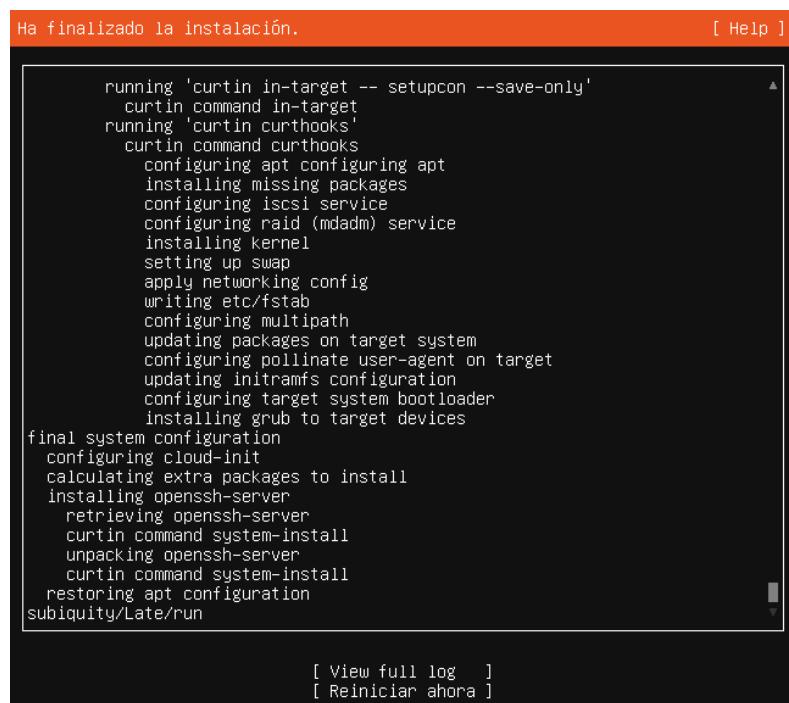
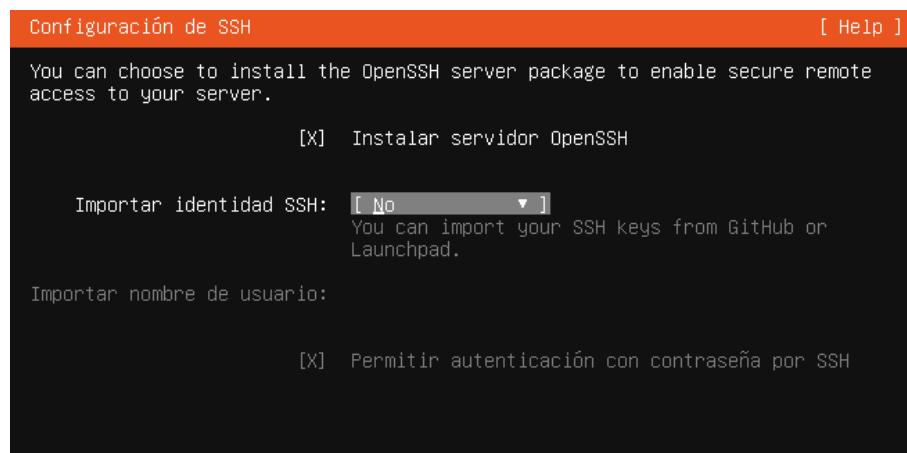
```

A continuación, elegiremos el nombre de la máquina, el usuario y la contraseña

Configuración de perfil		[Help]
Proporcione el nombre de usuario y la contraseña que utilizará para acceder al sistema. Puede configurar el acceso SSH en la pantalla siguiente, pero aun se necesita una contraseña para sudo.		
Su nombre:	rsp-usedlimpia	
El nombre del servidor:	rsp-usedlimpia The name it uses when it talks to other computers.	
Elija un nombre de usuario:	miadmin	
Elija una contraseña:	*****	
Confirme la contraseña:	*****	



Instalamos el servidor OpenSSH, esperamos a que se complete la instalación y reiniciamos



1.3 Nombre y configuración de red

Para cambiar el nombre del equipo, abrimos el nuevo Ubuntu Server, usamos el comando “`sudo hostnamectl set-hostname rsp-used`” y modificamos el fichero host con “`sudo nano /etc/hosts`” sustituyendo “rsp-usedlimpia” por “rsp-used”

```
miadmin@rsp-usedlimpia:~$ sudo hostnamectl set-hostname rsp-used
[sudo] password for miadmin:
miadmin@rsp-usedlimpia:~$ sudo nano /etc/hosts
```

```
127.0.0.1 localhost
127.0.1.1 rsp-usedlimpia ←

# 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
```

Una vez guardados los cambios, reiniciamos con “reboot” y escribimos “`hostname`” para confirmar que se han realizado los cambios

```
miadmin@rsp-usedlimpia:~$ reboot

miadmin@rsp-used:~$ hostname
rsp-used
```

Si queremos acceder a la configuración de la red, nos dirigimos a la ruta de los archivos de configuración escribiendo “`cd /etc/netplan`”. Para modificar el archivo de red usaremos en comando “`sudo nano 00-installer-config.yaml`”

```
miadmin@rsp-usedlimpia:~$ cd /etc/netplan/
miadmin@rsp-usedlimpia:/etc/netplan$ sudo nano 00-installer-config.yaml
```

A continuación, aparecerá esta pantalla con el contenido de ese archivo. Si queremos cambiar la IP de nuestra maquina basta con editar la que se muestra en el archivo (es muy importante respetar la tabulación y el espaciado de este archivo). Para aplicar los cambios usamos el comando “`sudo netplan apply`” y se cambiará la dirección IP por la nueva.

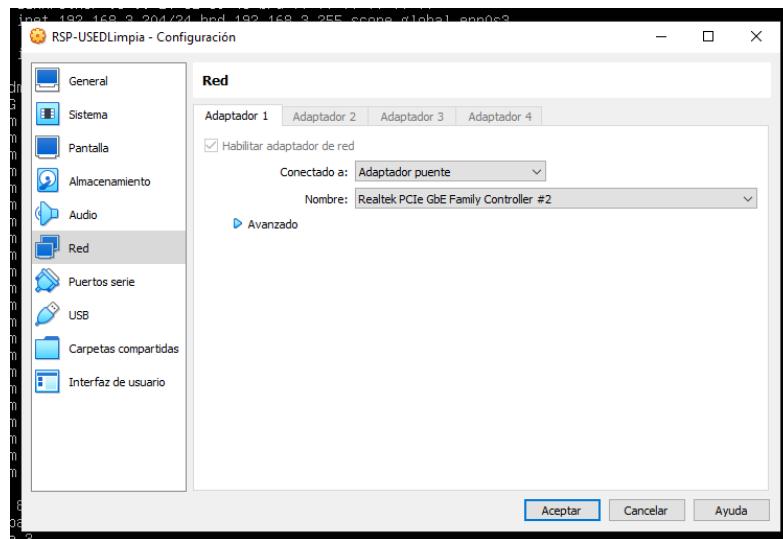
```
# This is the network config written by 'subiquity'
network:
  ethernets:
    enp0s3:
      addresses:
        - 192.168.1.204/24 ←
      nameservers:
        addresses:
          - 8.8.8.8
          - 8.8.8.8
        search: []
      routes:
        - to: default
          via: 192.168.1.1
  version: 2
```

```
miadmin@rsp-usedlimpia:/etc/netplan$ sudo netplan apply
```

Iniciamos sesión con el usuario “miadmin” y comprobamos que tenemos conexión a internet con el comando “`ping google.es`”. En el caso de que no se conecte tendremos que cambiar

la configuración de red de la maquina desde VirtualBox a adaptador puente y reiniciar la maquina con el comando “reboot”

```
miadmin@rsp-usedlimpia:~$ ping google.es
ping: google.es: Temporary failure in name resolution
```



1.4 Actualizar el sistema

Cuando tengamos conexión a internet escribimos “[sudo apt update](#)” y “[sudo apt upgrade](#)” para actualizar los repositorios y librerías.

Para configurar la fecha y la hora antes debemos conocer cuál es la fecha y hora activa en nuestro server, para ello escribimos “date” y para cambiar a la franja horaria de Madrid usaremos el comando “[sudo timedatectl set-timezone Europe/Madrid](#)”

```
miadmin@rsp-usedlimpia:~$ date
mar 03 oct 2023 09:01:54 UTC
```

```
miadmin@rsp-usedlimpia:~$ sudo timedatectl set-timezone Europe/Madrid
[sudo] password for miadmin:
miadmin@rsp-usedlimpia:~$ date
mar 03 oct 2023 11:07:55 CEST
miadmin@rsp-usedlimpia:~$ _
```

1.5 Cortafuegos local

En primer lugar, tenemos que saber cuál es el estado del cortafuegos con el comando “[sudo ufw status](#)”, si está inactivo lo activamos con “[systemctl status ufw](#)” y “[sudo ufw enable](#)”

```
miadmin@rsp-usedlimpia:~$ sudo ufw status
Status: inactive
```

```
miadmin@rsp-usedlimpia:~$ systemctl status ufw
● ufw.service - Uncomplicated firewall
   Loaded: loaded (/lib/systemd/system/ufw.service; enabled; vendor preset: enabled)
     Active: active (exited) since Tue 2023-10-03 08:17:01 UTC; 17min ago
       Docs: man:ufw(8)
   Main PID: 555 (code=exited, status=0/SUCCESS)
     CPU: 1ms

oct 03 08:17:01 rsp-usedlimpia systemd[1]: Starting Uncomplicated firewall...
oct 03 08:17:01 rsp-usedlimpia systemd[1]: Finished Uncomplicated firewall.
miadmin@rsp-usedlimpia:~$ _
```

```
miadmin@rsp-usedlimpia:~$ sudo ufw enable
Firewall is active and enabled on system startup
miadmin@rsp-usedlimpia:~$ _
```

Cuando tenemos el cortafuegos activo abrimos el puerto 22 con “[sudo ufw allow 22](#)” y comprobamos que está abierto con “[sudo ufw status](#)”

```
miadmin@rsp-usedlimpia:~$ sudo ufw allow 22
Rule added
Rule added (v6)
miadmin@rsp-usedlimpia:~$ sudo ufw status
Status: active

To                         Action      From
--                         ----      --
22                         ALLOW      Anywhere
22 (v6)                     ALLOW      Anywhere (v6)

miadmin@rsp-usedlimpia:~$ _
```

2. Cuentas de administración

Para la creación de un usuario nuevo usamos el comando “sudo adduser miadmin2” y rellenamos los campos que queramos por ejemplo la contraseña

```
miadmin@rsp-used:~$ sudo adduser miadmin2
[sudo] password for miadmin:
Adding user `miadmin2' ...
Adding new group `miadmin2' (1001) ...
Adding new user `miadmin2' (1001) with group `miadmin2' ...
Creating home directory `/home/miadmin2' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for miadmin2
Enter the new value, or press ENTER for the default
      Full Name []:
      Room Number []:
      Work Phone []:
      Home Phone []:
      Other []:
Is the information correct? [Y/n]
miadmin@rsp-used:~$ _
```

Si queremos añadir el nuevo usuario al grupo de super usuarios (es decir, el grupo de los admins o root) escribiremos el comando “sudo usermod -aG sudo miadmin2”

```
miadmin@rsp-used:~$ sudo usermod -aG sudo miadmin2
miadmin@rsp-used:~$ _
```

Para comprobar que hemos creado el usuario correctamente y que pertenece al grupo que se le ha especificado usamos el comando “grep “miadmin2” /etc/group”

```
miadmin@rsp-used:~$ grep "miadmin2" /etc/group
sudo:x:27:miadmin,miadmin2
miadmin2:x:1001:
```

3. Apache

3.1 ¿Qué es y para qué sirve?

Apache es un servidor web HTTP de código abierto que se usa para poder desarrollar y acceder a aplicaciones web con modelos cliente/servidor a través de los protocolos HTTP y HTTPS. La funcionalidad principal de este servicio web es abastecer a los usuarios de todos los ficheros necesarios para visualizar una web. Las solicitudes de los usuarios se hacen normalmente mediante un navegador (Chrome, Firefox, Safari, etc.).

3.2 Módulos a instalar

Para listar los módulos activos de apache escribiremos “apache2ctl -M”. A continuación, se muestra un listado de todos los módulos de Apache que se han instalado en nuestro servidor web:

- Status
- Alias
- Rewrite
- Php

```
miadmin@rsp-used:~$ sudo apache2ctl -M
Loaded Modules:
 core_module (static)
 so_module (static)
 watchdog_module (static)
 http_module (static)
 log_config_module (static)
 logio_module (static)
 version_module (static)
 unixd_module (static)
 access_compat_module (shared)
 alias_module (shared)
 auth_basic_module (shared)
 authn_core_module (shared)
 authn_file_module (shared)
 authz_core_module (shared)
 authz_host_module (shared)
 authz_user_module (shared)
 autoindex_module (shared)
 deflate_module (shared)
 dir_module (shared)
 env_module (shared)
 filter_module (shared)
 mime_module (shared)
 mpm_prefork_module (shared)
 negotiation_module (shared)
 php_module (shared)
 reqtimeout_module (shared)
 setenvif_module (shared)
 status_module (shared)
```

3.3 Comandos de control del servicio

- sudo apache2ctl -S: muestra los sitios activos
- sudo apache2ctl -M: muestra los módulos activos

- `sudo apache2ctl {start|stop|restart|gracefullconfigtest}`: en orden arranca, para, reinicia, reinicia de forma breve y valida los ficheros de configuración (este último comando, te muestra los posibles errores o warning que has podido cometer en algún fichero de configuracion)
- `a2ensite nombre-del-sitio`: activa el sitio virtual que se le especifica
- `a2dissite nombre-del-sitio`: desactiva el sitio virtual que se le especifica
- `a2enmod nombre-del-modulo`: activa el módulo que se le especifica
- `a2dismod nombre-del-modulo`: desactiva el módulo que se le especifica
- `a2enconfmi nombre-del-archivo`: habilita archivos de configuración

3.4 Instalación y configuración

En la maquina RSP-USDED instalaremos el servicio Apache2, pero antes de eso tenemos que actualizar los repositorios de Linux con el comando “`sudo apt-get update`”.

Para instalar Apache escribiremos “`sudo apt-get install apache2`”:

```
miadmin@rsp-used:~$ sudo apt-get install apache2
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
apache2 ya está en su versión más reciente (2.4.52-1ubuntu4.6).
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 5 no actualizados.
```

Una vez instalado Apache, tenemos que permitir el acceso a Apache al puerto 80 del cortafuegos de nuestro server mediante el comando “`sudo ufw allow apache`” y para comprobarlo escribimos “`sudo ufw status`”

```
miadmin@rsp-used:~$ sudo ufw allow apache
Rule added
Rule added (v6)
```

```
miadmin@rsp-used:~$ sudo ufw status
Status: active

To                         Action      From
---                         ----      ---
22                         ALLOW      Anywhere
Apache ←                   ALLOW      Anywhere
22 (v6)                    ALLOW      Anywhere (v6)
Apache (v6) ←              ALLOW      Anywhere (v6)
```

3.5 Estructura de directorios y ficheros de configuración

Para comprobar cuál es la estructura de directorios que tiene por defecto apache, basta con editar el fichero de configuracion con “`sudo nano /etc/apache2/sites-available/000-default.conf`” y leer las lineas comentadas

```
# It is split into several files forming the configuration below, all located in the /etc/apache2/ directory:
#
#      /etc/apache2/
#          |-- apache2.conf
#              '-- ports.conf
#          |-- mods-enabled
#              '-- *.load
#                  '-- *.conf
#          |-- conf-enabled
#              '-- *.conf
#          '-- sites-enabled
#              '-- *.conf
```

- apache2.conf: es el archivo de configuración principal. Reúne las piezas al incluir todos los archivos de configuración restantes al iniciar el servidor web.
- ports.conf: siempre se incluye en el archivo de configuración principal. Determina los puertos de escucha para las conexiones entrantes que se pueden personalizar en cualquier momento.
- Los archivos de configuración en los directorios mods-enabled/, conf-enabled/ y sites-enabled/ contienen fragmentos de configuración particulares que administran módulos, fragmentos de configuración global o configuraciones de host virtual, respectivamente.
- Los archivos de configuración en los directorios mods-available/, conf-available/ y sites-available/ contienen fragmentos de configuración particulares que administran módulos disponibles, fragmentos de configuración global disponibles o configuraciones de host virtual disponibles, respectivamente.

3.6 Configuración de Logs de errores y acceso

Modificamos con “sudo nano /etc/apache2/sites-available/000-default.conf” el fichero de configuración para grabar los registros logs en un directorio específico. Para ello escribimos estas 2 líneas:

```
# 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
ErrorLog /var/www/html/log/error.log
CustomLog /var/www/html/log/access.log combined

# For most configuration files from conf-available/, which
# includes most .conf files, the settings below are copied
# at the end of the file.  This permits you to overwrite
# any defaults in conf-available/ without having to
# change the individual #include statements.
```



- La línea ErrorLog /var/www/html/log/error.log sirve para generar un archivo (error.log) con registros logs de los errores en la ruta que se especifica (en /var/www/html/log).
- La línea CustomLog /var/www/html/log/access.log combined sirve para generar una serie de registros de solicitudes o acceso a archivos en un servidor web.

Después de esto creamos el directorio para almacenar los logs que hemos especificado, para ello usamos “sudo mkdir /var/www/html/log” y reiniciamos el servicio de apache.

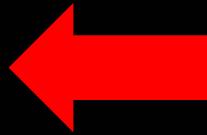
3.7 Habilitar el uso de .htaccess

Configuraremos el archivo de configuración de apache con “sudo nano /etc/apache2/apache2.conf” y escribimos lo siguiente:

```
        AllowOverride None
        Require all granted
</Directory>

<Directory /var/www/>
        Options -Indexes +FollowSymLinks
        AllowOverride All
        Require all granted
</Directory>

#<Directory /srv/>
#       Options Indexes FollowSymLinks
```



- Options -Indexes +FollowSymLinks: Define las opciones para el directorio. Significa que se desactiva la función de mostrar automáticamente el listado de archivos si no hay un archivo de índice en el directorio (-Indexes) y se permite el seguimiento de enlaces simbólicos (+FollowSymLinks)
- AllowOverride All: Permite la anulación de configuraciones mediante archivos .htaccess en ese directorio. Esto permite a los usuarios locales del directorio personalizar algunas configuraciones para sus propios fines.
- Require all granted: Esto establece que se permite el acceso a este directorio a cualquier usuario o cliente que intente acceder

Ahora creamos el archivo .htaccess escribiendo con “sudo nano /var/www/html/.htaccess” y escribimos:

```
GNU nano 6.2
DirectoryIndex index.php index.html
```

- La directiva “DirectoryIndex index.php index.html” indica el nombre del archivo que se debe servir como página de inicio cuando se accede a un directorio en el servidor web, tiene preferencia el primer archivo (index.php) y en el caso de que no se encuentre el fichero, se buscará el segundo nombre (index.html) y así consecutivamente.

Para validar los archivos de configuración de Apache escribimos el comando “sudo apache2ctl configtest” y en nuestro caso aparece un warning que tenemos que subsanar modificando el fichero de configuración del servicio

```
miadmin@rsp-used:~$ sudo apache2ctl configtest
[sudo] password for miadmin:
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0
.1.1. Set the 'ServerName' directive globally to suppress this message
Syntax OK
```

Usamos “sudo nano /etc/apache2/apache2.conf” y al final del archivo escribimos “ServerName rsp-used” y reiniciamos el servicio

```
# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
ServerName rsp-used
```

- La directiva “ServerName rsp-used” se utiliza para especificar el nombre del host o el dominio al que el servidor web debe responder (en nuestro caso es el mismo nombre que el de la máquina virtual del Ubuntu server: rsp-used)

Para instalar un navegador de texto usamos el comando “sudo apt install lynx” y para activar el módulo de apache status usamos “sudo a2enmod status” y reiniciamos apache2

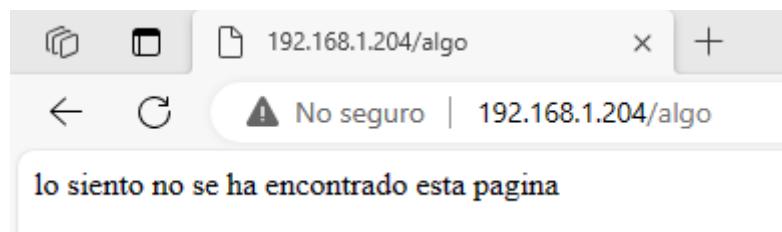
```
miadmin@rsp-used:~$ sudo a2enmod status
Module status already enabled
```

- El módulo mod_status sirve para saber el rendimiento de nuestro servidor Apache de forma remota a través de una página HTML que accedemos desde el navegador

Para ver los informes escribimos en la terminal “apache2ctl fullstatus” y para la personalización de errores podemos editar el fichero .htaccess (con “sudo nano /var/www/html/.htaccess”)

```
miadmin@rsp-used:~$ apache2ctl fullstatus
[...]
DirectoryIndex index.php index.html
ErrorDocument 404 "lo siento no se ha encontrado esta pagina"
```

- La directiva “ErrorDocument 404 “lo siento...”” sirve para personalizar los mensajes de error. En primer lugar, se especifica el código de error HTTP que queremos personalizar (404) y a continuación escribimos el mensaje de error que queremos que aparezca en el navegador entre comillas (“lo siento, esta página no está disponible”)



3.8 Modulo Alias

Instalaremos un módulo de apache llamado Alias que nos permite el acceso a directorios superiores a los que puede acceder un usuario (directorios virtuales). En primer lugar, comprobamos que el módulo Alias esté activo con “ls /etc/apache2/mods-enabled/ | grep alias”.

```
miadmin@rsp-used:~$ ls /etc/apache2/mods-enabled/ | grep alias
alias.conf
alias.load
```

Si está activo, modificamos el fichero de configuración del sitio virtual (“sudo nano /etc/apache2/sites-enabled/000-default.conf”) escribiendo las siguientes líneas:

```

ServerAdmin webmaster@localhost
DocumentRoot /var/www/html
Alias /documentos /home/miadmin/doc
<Directory /home/miadmin/doc>
    Options +FollowSymLinks +Indexes
    AllowOverride None
    Require all granted
</Directory>

```

- La directiva “Alias /documentos /home/miadmin/doc” vincula una carpeta existente en nuestro proyecto (en nuestro caso /documentos) con un directorio virtual o externo al proyecto especificando su ruta absoluta (/home/miadmin/doc)

Para hacer una prueba creamos una carpeta *doc* en */home/miadmin* y dentro de esta creamos 2 archivos de texto *file1.txt* y *file2.txt*. Usamos el comando “sudo nano /etc/apache2/mods-enabled/alias.conf” para modificar el archivo de configuración de alias y escribimos las siguientes directivas y usamos “sudo service apache2 restart” para reiniciar Apache

```

</Directory>
Alias /documentos /home/miadmin/doc
<Directory /home/miadmin/doc>
    Options FollowSymLinks Indexes
    AllowOverride None
    Require all granted
</Directory> -

```

3.9 Modulo Rewrite

El módulo Rewrite de Apache nos permite crear direcciones URL alternativas a las generadas por nuestro sitio web. Habilitamos el módulo con “sudo a2enmod rewrite” y reiniciamos el servicio

```

miadmin@rsp-used:~$ sudo a2enmod rewrite
Enabling module rewrite.
To activate the new configuration, you need to run:
    systemctl restart apache2

```

Como ejemplo, editaremos un archivo *.htaccess* y le escribiremos la siguiente linea:

```
Redirect 301 /educa https://www.educa.jcyl.es/es
```

- Esta directiva redirige la primera ruta (/educa) automáticamente a la segunda (https://www.educa.es) como un enlace

A continuación, escribimos en el navegador la URL de nuestro proyecto seguido de */educa* y cuando carguemos la página, nos redimirá automáticamente a la página web de *educa*.

3.10 HTTPS

Para trabajar con un protocolo seguro de transferencia de hipertexto (HTTPS), deberemos configurar el protocolo SSL/TLS. Con esto lo que hacemos es generar una clave publica y un certificado en nuestro servidor.

En primer lugar, generamos una clave privada en nuestro servidor con el comando “openssl genrsa 2048 > claveprivada.key”

```
miadmin@rsp-used:~$ pwd
/home/miadmin
miadmin@rsp-used:~$ openssl genrsa 2048 > claveprivada.key
miadmin@rsp-used:~$
```

A continuación, haremos una solicitud del certificado a partir de la clave privada que hemos creado con el comando “openssl req -new -key claveprivada.key > certificado.csr” especificando el nombre del archivo con extensión csr (esta es la extensión del certificado) que queremos darle. A medida que se crea nos va pidiendo datos que tendremos que facilitarle como el nombre del país, la provincia, el municipio, el dominio...

```
miadmin@rsp-used:~$ openssl req -new -key claveprivada.key > certificado.csr
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]:Zamora
Locality Name (eg, city) []:Benavente
Organization Name (eg, company) [Internet Widgits Pty Ltd]:IES Los Sauces
Organizational Unit Name (eg, section) []:Informatica
Common Name (e.g. server FQDN or YOUR name) []:daw204.rebeca.local
Email Address []:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
miadmin@rsp-used:~$
```

Se firma el certificado con una autofirma que genera un archivo con la extensión “.crt” con el comando “openssl x509 -req -days 365 -in certificado.csr -signkey claveprivada.key -out certificado.crt” y comprobamos con un ls que se han generado los 3 ficheros correctamente

```
miadmin@rsp-used:~$ openssl x509 -req -days 365 -in certificado.csr -signkey claveprivada.key -out certificado.crt
Certificate request self-signature ok
subject=C = ES, ST = Zamora, L = Benavente, O = IES Los Sauces, OU = Informatica, CN = daw204.rebeca.local
miadmin@rsp-used:~$ ls
204DAWProyectoDAW certificado.crt certificado.csr claveprivada.key doc
```

A continuación, pasaremos a configurar apache para poder usar la clave privada y el certificado que hemos generado. Para ello, en primer lugar, activamos el módulo SSL con el comando “sudo a2enmod ssl”

```
miadmin@rsp-used:~$ sudo a2enmod ssl
[sudo] password for miadmin:
Considering dependency setenvif for ssl:
Module setenvif already enabled
Considering dependency mime for ssl:
Module mime already enabled
Considering dependency socache_shmcb for ssl:
Enabling module socache_shmcb.
Enabling module ssl.
See /usr/share/doc/apache2/README.Debian.gz on how to configure SSL and create self-signed certificates.
To activate the new configuration, you need to run:
  systemctl restart apache2
miadmin@rsp-used:~$
```

Movemos la clave privada al directorio /etc/ssl/private con “sudo mv claveprivada.key /etc/ssl/private”, cambiamos el propietario para que pertenezca al grupo root:ssl-cert con “sudo chown root:ssl-cert /etc/ssl/private/claveprivada.key”, cambiamos los permisos para que tenga 640 con “sudo chmod 640 /etc/ssl/private/claveprivada.key” y comprobamos listando el contenido de ese directorio

```
miadmin@rsp-used:~$ sudo mv claveprivada.key /etc/ssl/private
miadmin@rsp-used:~$ sudo chown root:ssl-cert /etc/ssl/private/claveprivada.key
miadmin@rsp-used:~$ sudo ls -l /etc/ssl/private/claveprivada.key
-rw-rw-r-- 1 root ssl-cert 1704 nov 24 08:48 /etc/ssl/private/claveprivada.key
miadmin@rsp-used:~$ sudo chmod 640 /etc/ssl/private/claveprivada.key
miadmin@rsp-used:~$ sudo ls -l /etc/ssl/private/claveprivada.key
-rw-r----- 1 root ssl-cert 1704 nov 24 08:48 /etc/ssl/private/claveprivada.key
```

Movemos el certificado al directorio certs con “sudo mv certificado.crt /etc/ssl/certs” y cambiamos el propietario para que pertenezca a root:root con “sudo chown root:root /etc/ssl/certs/certificado.crt” y comprobamos listando el contenido de ese directorio

```
miadmin@rsp-used:~$ sudo mv certificado.crt /etc/ssl/certs
miadmin@rsp-used:~$ sudo chown root:root /etc/ssl/certs/certificado.crt
miadmin@rsp-used:~$ sudo ls -l /etc/ssl/certs/certificado.crt
-rw-rw-r-- 1 root root 1281 nov 24 09:06 /etc/ssl/certs/certificado.crt
```

A continuación, vamos a /etc/apache2/sites-available y copiamos el sitio “default-ssl.conf” creando uno nuevo llamado “rebeca-ssl.conf” con el comando “sudo cp default-ssl.conf rebeca-ssl.conf”

```
miadmin@rsp-used:/etc/apache2/sites-available$ sudo cp default-ssl.conf rebeca-ssl.conf
miadmin@rsp-used:/etc/apache2/sites-available$ sudo nano rebeca-ssl.conf
```

Editamos “rebeca-ssl.conf”: insertamos SSLEngine on, modificamos las líneas SSLCertificateFile y SSLCertificateKeyFile especificando la ruta donde se encuentran el certificado y la clave privada respectivamente y añadimos un dominio que tenga de puerto el 443 (HTTPS)

```
# Enable/Disable SSL for this virtual host.
SSLEngine on

# A self-signed (snakeoil) certificate can be created by installing
# the ssl-cert package. See
# /usr/share/doc/apache2/README.Debian.gz for more info.
# If both key and certificate are stored in the same file, only the
# SSLCertificateFile directive is needed.
SSLCertificateFile      /etc/ssl/certs/certificado.crt
SSLCertificateKeyFile  /etc/ssl/private/claveprivada.key

# Server Certificate Chain:
# Point SSLCertificateChainFile at a file containing the
# -----BEGIN CERTIFICATE-----
```

`<VirtualHost *:443>
 ServerAdmin webmaster@localhost
 ServerName daw204.rebeca.local
 DocumentRoot /var/www/html

 # Available loglevels: trace8, ..., trace1,
 # error, crit, alert, emerg.`

Guardamos el documento y comprobamos que no hay errores con “sudo apache2ctl configtest”

```
miadmin@rsp-used:/etc/apache2/sites-available$ sudo apache2ctl configtest
Syntax OK
```

Si todo está bien activamos el sitio con “sudo a2ensite rebeca-ssl.conf” y reiniciamos apache

```
miadmin@rsp-used:/etc/apache2/sites-available$ sudo a2ensite rebeca-ssl.conf
Enabling site rebeca-ssl.
To activate the new configuration, you need to run:
  systemctl reload apache2
miadmin@rsp-used:/etc/apache2/sites-available$ sudo systemctl reload apache2
```

Abrimos el puerto 443 para poder trabajar con https con “`sudo ufw allow 443`” y comprobamos que este abierto con “`sudo ufw status`”

```
miadmin@rsp-used:/etc/apache2/sites-available$ sudo ufw allow 443
Rule added
Rule added (v6)
miadmin@rsp-used:/etc/apache2/sites-available$ sudo ufw status
Status: active

To                         Action      From
--                         --          --
22                         ALLOW       Anywhere
Apache                      ALLOW       Anywhere
9003                        ALLOW      Anywhere
3306                        ALLOW      Anywhere
443                         ALLOW      Anywhere
22 (v6)                     ALLOW      Anywhere (v6)
Apache (v6)                  ALLOW      Anywhere (v6)
9003 (v6)                   ALLOW      Anywhere (v6)
3306 (v6)                   ALLOW      Anywhere (v6)
443 (v6)                    ALLOW      Anywhere (v6)
```

Para comprobar que hemos activado el certificado escribimos en el navegador `https://192.168.3.204` y debería aparecer lo siguiente:



La conexión no es privada

Es posible que los atacantes estén intentando robar tu información de **192.168.3.204** (por ejemplo, contraseñas, mensajes o tarjetas de crédito). [Más información](#)

NET::ERR_CERT_AUTHORITY_INVALID

Q Para disfrutar del máximo nivel de seguridad en Chrome, [activa la protección mejorada.](#)

[Configuración avanzada](#)

[Volver para estar a salvo](#)

Modificamos el fichero `000-default` y escribimos estas líneas:

```
miadmin@rsp-used:/etc/apache2/sites-available$ sudo nano 000-default.conf
[sudo] password for miadmin:
# value is not decisive as it is used as a test
# However, you must set it for any further virtu
#ServerName www.example.com

ServerName daw204.rebeca.local ←
ServerAdmin webmaster@localhost
DocumentRoot /var/www/html

# Available loglevels: trace8, ..., trace1, debu
# error, crit, alert, emerg.
# It is also possible to configure the loglevel
```

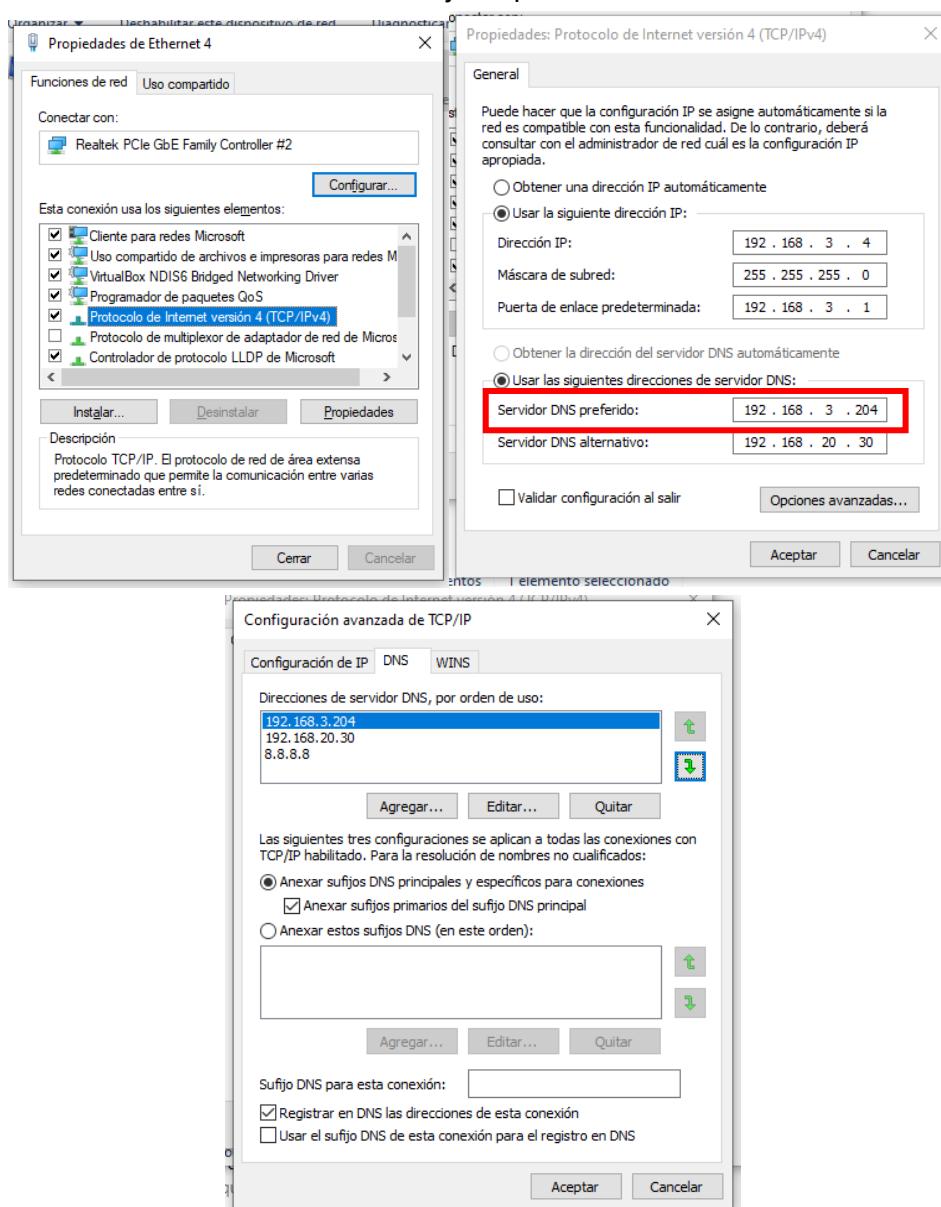
Nos vamos a `/var/www/html/.htaccess` y escribimos las siguientes líneas en el archivo para que cuando escribamos el dominio en el navegador nos redireccione automáticamente a https

```
HTTP/1.1 404 El recurso que intento no existe
Redirect 301 /educa https://www.educa.jcyl.es/es
RewriteEngine On
RewriteCond %{SERVER_PORT} 80
RewriteRule ^(.*)$ https://daw204.rebeca.local/$1 [R,L]
```

Guardamos el fichero y comprobamos que el módulo rewrite este activo con “sudo a2enmod rewrite” y reiniciamos apache y hacemos un configtest para ver que todo ha ido bien

```
miadmin@rsp-used:/var/www/html$ miadmin@rsp-used:/var/www/html$ sudo a2enmod rewrite
Module rewrite already enabled
miadmin@rsp-used:/var/www/html$ sudo apache2ctl configtest
Syntax OK
```

Por último, nos vamos a la configuración de red de nuestra máquina anfitriona añadiendo un servidor DNS con la IP de nuestro servidor y aceptamos los cambios.



3.11 Autenticación y autorización

3.11.1 Entorno de desarrollo (nuestra máquina virtual)

En primer lugar, activaremos el módulo correspondiente a la autenticación digest (auth_digest.load) con el comando “sudo a2enmod auth_digest.load” y para asegurarnos de

que se ha activado correctamente podemos escribir “`ls /etc/apache2/mods-enabled | grep digest`”

```
miadmin@rsp-used:~$ ls /etc/apache2/mods-available | grep digest
auth_digest.load
```

```
miadmin@rsp-used:~$ sudo a2enmod auth_digest.load
Considering dependency authn_core for auth_digest:
Module authn_core already enabled
Enabling module auth_digest.
To activate the new configuration, you need to run:
    systemctl restart apache2
miadmin@rsp-used:~$ sudo service apache2 restart
```

```
miadmin@rsp-used:~$ ls /etc/apache2/mods-enabled | grep digest
auth_digest.load
```

Ahora crearemos los siguientes usuarios:

USERS	REALM	PASSWORD	GROUP
heraclio	profesores	paso	jefes
amor	profesores	paso	curritos
rebeca	alumnos	paso	curritos

Para ello, entramos en esta página web para generar el htdigest y rellenamos los campos que aparecen: <https://websistent.com/tools/htdigest-generator-tool/>

htdigest Generator Tool Online



Generate the contents of a .htdigest file for your [web server](#) to perform **HTTP Digest Authentication**. This tool encrypts the password entered here so that it is secure and usable in a .htdigest file. One major advantage of digest authentication over basic(.htpasswd) authentication is that the former(.htdigest) transfers the password from the user's computer to the server in an encrypted form. Type your desired username, realm and password and click generate.

Username:

Realm:

Password:

Confirm Password:

GENERATE

- Username: nombre del usuario (lo llamaremos “heraclio”)
- Realm: dominio que indica un criterio de agrupación de los usuarios (en nuestro caso heraclio pertenece a profesores)
- Password: la contraseña que queremos que tenga el usuario

- Confirm Password: confirmar la contraseña anterior

Username:
heracio

Realm:
profesores

Password:

Confirm Password:

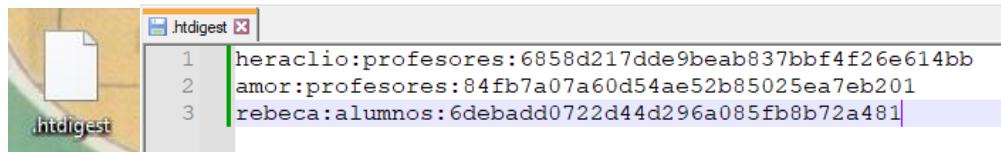
Se generará el siguiente código que guardaremos, cada código hace referencia a un usuario:

Paste the following line into your .htdigest file

heracio:profesores:6858d217dde9beab837bbf4f26e614bb

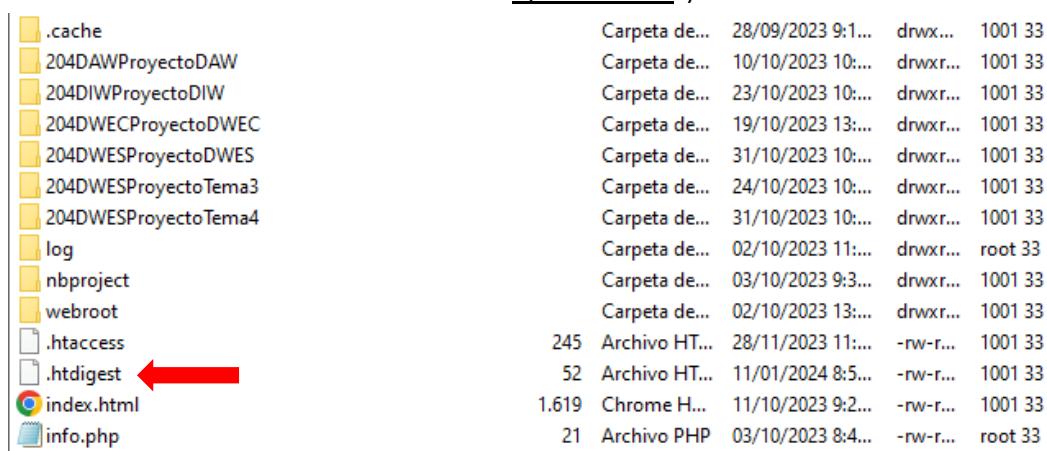
Creamos además de heracio otros 2 usuarios: amor (que va a pertenecer al dominio profesores) y rebeca (del grupo alumnos)

Seguimos las instrucciones, creamos un archivo con ese nombre (.htdigest) y agregamos ese contenido:



```
1 heracio:profesores:6858d217dde9beab837bbf4f26e614bb
2 amor:profesores:84fb7a07a60d54ae52b85025ea7eb201
3 rebeca:alumnos:6debadd0722d44d296a085fb8b72a481
```

Este fichero lo publicaremos en el directorio /var/www/html del servidor mediante filezilla (nos conectamos con nuestro usuario administrador "operadorweb")



.cache		Carpetas de...	28/09/2023 9:1...	drwx...	1001 33
204DAWProyectoDAW		Carpetas de...	10/10/2023 10:...	drwxr...	1001 33
204DIWProyectoDIW		Carpetas de...	23/10/2023 10:...	drwxr...	1001 33
204DWECProyectoDWECP		Carpetas de...	19/10/2023 13:...	drwxr...	1001 33
204DWESProyectoDWES		Carpetas de...	31/10/2023 10:...	drwxr...	1001 33
204DWESProyectoTema3		Carpetas de...	24/10/2023 10:...	drwxr...	1001 33
204DWESProyectoTema4		Carpetas de...	31/10/2023 10:...	drwxr...	1001 33
log		Carpetas de...	02/10/2023 11:...	drwxr...	root 33
nbproject		Carpetas de...	03/10/2023 9:3...	drwxr...	1001 33
webroot		Carpetas de...	02/10/2023 13:...	drwxr...	1001 33
.htaccess					
.htdigest		245 Archivo HT...	28/11/2023 11:...	-rw-r...	1001 33
		52 Archivo HT...	11/01/2024 8:5...	-rw-r...	1001 33
		1.619 Chrome H...	11/10/2023 9:2...	-rw-r...	1001 33
		21 Archivo PHP	03/10/2023 8:4...	-rw-r...	root 33

Ahora en el proyecto del tema 4 en la carpeta de scriptDB creamos un .htaccess que contenga el siguiente contenido:

The screenshot shows a file manager interface. At the top, it says "Sitio remoto: /html/204DWESProyectoTema4/scriptDB". Below this is a tree view of directory contents:

- 204DWECProyectoDWEC
- 204DWESProyectoDWES
- 204DWESProyectoTema3
- 204DWESProyectoTema4

Below the tree view is a table showing files in the current directory:

Nombre de archivo	Tamaño d...	Tipo de arc...	Últin
..			
.htaccess	149	Archivo HT...	11/01/2023
CreaDAW204DBDepartamentos.sql	570	Archivo SQL	31/11/2022

A red arrow points to the ".htaccess" file in the list.

Below the table is the content of the ".htaccess" file:

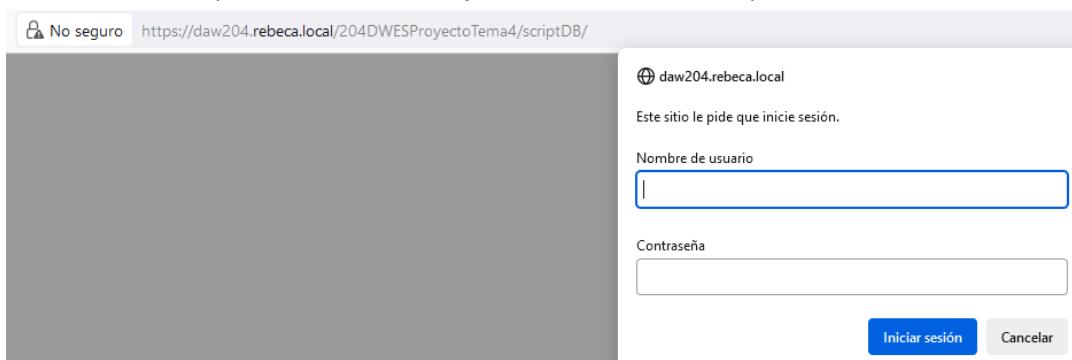
```

Options Indexes FollowSymLinks
AuthType Digest
AuthName "profesores"
AuthDigestProvider file
AuthUserFile /var/www/html/.htdigest
Require valid-user

```

- AuthType: Tipo de autenticación que se usará, en este caso será Digest.
- AuthName: Nos servirá como realm, el dominio al que pertenecen los usuarios. Algo así como un grupo (en nuestro caso autorizamos a profesores)
- AuthDigestProvider: Indica qué proveedor se utiliza para autenticar a los usuarios para esta ubicación. Por defecto usaremos file.
- AuthUserFile: Ruta donde se encuentra el fichero de passwords (.htdigest).
- Require: El/los requerimiento/s que deben ser satisfechos en orden para garantizar el acceso.

Comprobamos en nuestro navegador que nos pide el usuario y si no introducimos un usuario que no pertenezca al dominio profesores no nos dejará acceder a la ruta especificada



A continuación, agruparemos lo usuario mediante los grupos de apache. Activamos el módulo authz_groupfile.load y reiniciamos apache2

```
miadmin@rsp-used:~$ ls /etc/apache2/mods-available | grep groupfile
authz_groupfile.load
```

```
miadmin@rsp-used:~$ sudo a2enmod authz_groupfile.load
[sudo] password for miadmin:
Considering dependency authz_core for authz_groupfile:
Module authz_core already enabled
Enabling module authz_groupfile.
To activate the new configuration, you need to run:
    systemctl restart apache2
miadmin@rsp-used:~$ _

miadmin@rsp-used:~$ sudo service apache2 restart
miadmin@rsp-used:~$
```

Dentro del directorio `/var/www/html` creamos desde filezilla un archivo `.htgroup` con el siguiente contenido:

	<code>204DAWProyectoDAW</code>
	<code>204DIWProyectoDIW</code>
	<code>204DWECProyectoDWEC</code>
	<code>204DWESProyectoDWES</code>
	<code>204DWESProyectoTema3</code>
	<code>204DWESProyectoTema4</code>
	<code>log</code>
	<code>nbproject</code>
	<code>webroot</code>
	<code>.htaccess</code>
	<code>.htdigest</code>
	<code>.htgroup</code> ←
	<code>index.html</code>
	<code>info.php</code>

.htgroup: Bloc de notas
 Archivo Edición Formato Ver Ayuda
jefes: heraclio
curritos: amor rebeca

- En cada línea del archivo aparecerá un grupo seguido de ":" con los usuarios que pertenecen a ese grupo

A continuación, editaremos el `.htaccess` nuevamente y agregaremos estas directivas:

.htaccess: Bloc de notas
 Archivo Edición Formato Ver Ayuda
Options Indexes FollowSymLinks
AuthType Digest
AuthName profesores
AuthDigestProvider file
AuthUserFile /var/www/html/.htdigest
AuthGroupFile /var/www/html/.htgroup ←
Require group jefes

- AuthGroupFile: especifica la ruta del archivo de los grupos (`.htgroup`)
- Require group: le da permisos solo al grupo que se especifica (`jefes`)

Ahora comprobamos que solo podemos conectarnos con heraclio y con ningún usuario más

3.11.1 Entorno de explotación (servidor ionos)

Ahora administraremos la autenticación y la autorización en nuestro entorno de explotación. En primer lugar, comprobamos que el módulo de apache auth_basic este activo con “ls /etc/apache2/mods-enabled | grep basic”

```
miadmin@rsp-used:~$ ls /etc/apache2/mods-enabled | grep basic
auth_basic.load
```

Para crear los usuarios esta vez usaremos esta página web: <https://hostingcanada.org/htpasswd-generator/>

Username
Enter the username you would like to add your .htpasswd file.
heracio

Password
Enter the password to be encrypted.
.....

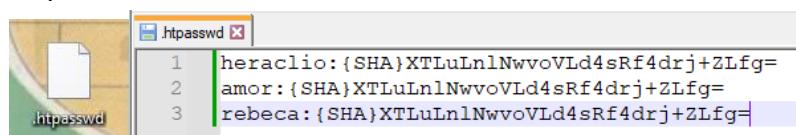
Mode
SHA1 (insecure)

Create .htpasswd file | Clear

- Username: nombre del usuario (lo llamaremos “heracio”)
- Password: la contraseña que queremos que tenga el usuario (paso)
- Mode: tipo de cifrado con el que queremos codificar la contraseña

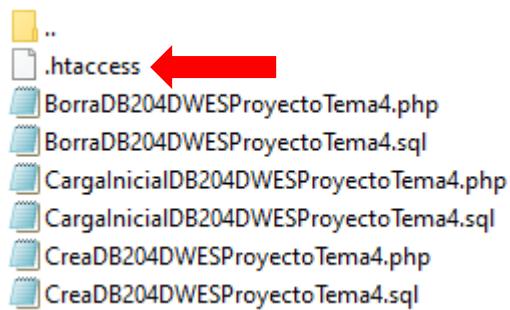
```
heracio:{SHA}XTLuLnINwvoVLd4sRf4drj+ZLfg=
```

Una vez creados los 3 usuarios, copiamos los códigos generados y los pegamos en un fichero llamado .htpasswd



Nuevamente, mediante filezilla, subiremos este fichero a nuestro directorio home del servidor de explotación

204DAWProyectoDAW	Carpeta de...	2:
204DIWProyectoDIW	Carpeta de...	2:
204DWECProyectoDWEC	Carpeta de...	1:
204DWESLoginLogoffTema5	Carpeta de...	1:
204DWESProyectoDWES	Carpeta de...	1:
204DWESProyectoTema3	Carpeta de...	1:
204DWESProyectoTema4	Carpeta de...	0:
204DWESProyectoTema5	Carpeta de...	0:
log	Carpeta de...	2:
webroot	Carpeta de...	2:
.gitignore	12 Document...	2:
.htaccess	229 Archivo HT...	2:
.htpasswd	124 Archivo HT...	1:
index.html	1.599 Chrome H...	2:
info.php	21 Archivo PHP	2:
LICENSE.txt	35.149 Document...	2:
README.md	413 Archivo MD	2:



4. PHP

4.1 Módulos

A continuación, se muestran los módulos instalados de php mostrados por el comando “dpkg --get-selections | grep php”:

```
miadmin@rsp-used:~$ dpkg --get-selections | grep php8.1
libapache2-mod-php8.1                               install
php8.1                                              install
php8.1-bz2                                         install
php8.1-cli                                         install
php8.1-common                                     install
php8.1-curl                                         install
php8.1-gd                                           install
php8.1-mbstring                                    install
php8.1-mysql                                         install
php8.1-opcache                                     install
php8.1-readline                                     install
php8.1-soap                                         install
php8.1-xdebug                                       install
php8.1-xml                                         install
php8.1-zip                                         install
miadmin@rspn-used:~$
```

4.2 Instalación de php y configuración de php.ini

Antes de instalar php tendremos que actualizar el sistema con “sudo apt update” y para instalarlo escribimos “sudo apt -y install php8.1” (instalamos la versión 8.1 porque es la más nueva que es estable) y para comprobar la versión podemos utilizar “sudo php -v”

```
miadmin@rsp-used:~$ sudo php -v
PHP 8.1.2-1ubuntu2.14 (cli) (built: Aug 18 2023 11:41:11) (NTS)
Copyright (c) The PHP Group
Zend Engine v4.1.2, Copyright (c) Zend Technologies
    with Zend OPcache v8.1.2-1ubuntu2.14, Copyright (c), by Zend Technologies
```

Instalamos la librería de módulos de apache con “sudo apt install libapache2-mod-php” y para crear un archivo info.php usamos “sudo nano /var/www/html/info.php” y escribimos lo siguiente:

```
<?php
    phpinfo();
?>
```

- phpinfo() es una función que muestra información detallada sobre la configuración de PHP y el entorno del servidor web

A continuación, reiniciamos el servicio de apache con “sudo service apache2 restart” y para comprobar que todo está bien escribimos en nuestro navegador 192.168.3.204/info.php y si aparece esta pantalla significa que se ha creado correctamente:

PHP Version 8.1.2-1ubuntu2.14	
System	Linux raspberrypi 5.15.0-86-generic #96-Ubuntu SMP Wed Sep 20 08:23:49 UTC 2023 x86_64
Build Date	Aug 18 2023 11:41:11
Build System	Linux
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/8.1/apache2
Loaded Configuration File	/etc/php/8.1/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/8.1/apache2/conf.d
Additional .ini files parsed	/etc/php/8.1/apache2/conf.d/10-opcache.ini, /etc/php/8.1/apache2/conf.d/10-pdo.ini, /etc/php/8.1/apache2/conf.d/20-calendar.ini, /etc/php/8.1/apache2/conf.d/20-type.ini, /etc/php/8.1/apache2/conf.d/20-exif.ini, /etc/php/8.1/apache2/conf.d/20-fn.ini, /etc/php/8.1/apache2/conf.d/20-fileinfo.ini, /etc/php/8.1/apache2/conf.d/20-ftp.ini, /etc/php/8.1/apache2/conf.d/20-gettext.ini, /etc/php/8.1/apache2/conf.d/20-iconv.ini, /etc/php/8.1/apache2/conf.d/20-phar.ini, /etc/php/8.1/apache2/conf.d/20-posix.ini, /etc/php/8.1/apache2/conf.d/20-readline.ini, /etc/php/8.1/apache2/conf.d/20-shmop.ini, /etc/php/8.1/apache2/conf.d/20-sockets.ini, /etc/php/8.1/apache2/conf.d/20-sysvmsg.ini,

Instalamos los módulos de SOAP y de xml con el comando “sudo apt install -y php8.1-xml php8.1-soap” y reiniciamos el servicio de apache2. En 192.168.3.204/info.php podemos comprobar que se han instalado correctamente

SNMP	Rasmus Lerdorf, Harrie Hazewinkel, Mike Jackson, Steven Lawrence, Johann Ra
SOAP	Brad Lafountain, Shane Caraveo, Dmitry Stogov
Sockets	Chris Vandomelen, Sterling Hughes, Daniel Beulshausen, Jason Greene
Sodium	Frank Denis
SPL	Marcus Boerger, Etienne Kneuss
SQLite 3.x driver for PDO	Wez Furlong
SQLite3	Scott MacVicar, Ilia Alshanetsky, Brad Dewar
System V Message based IPC	Wez Furlong
System V Semaphores	Tom May
System V Shared Memory	Christian Cartus
tidy	John Coggeshall, Ilia Alshanetsky
tokenizer	Andrei Zmievski, Johannes Schlueter
XML	Stig Bakken, Thies C. Arntzen, Sterling Hughes
XMLReader	Rob Richards
XMLWriter	Rob Richards, Pierre-Alain Joye
xml	Christian Shadman, Rob Richards

Ahora cambiaremos el fichero de configuración php.ini pero antes de esto haremos una copia de seguridad. Cambiamos de directorio a “cd /etc/php/8.1/apache2” y usamos el comando “sudo cp php.ini php.ini.backup” para hacer la copia.

```
miadmin@raspberrypi:~$ sudo cp php.ini php.ini.backup
miadmin@raspberrypi:~$ ls
conf.d  php.ini  php.ini.backup
```

Usamos el comando “sudo nano php.ini” para modificar el fichero de configuración de php. Buscamos en el archivo la línea “display errors” y cambiamos la línea de display_errors=Off y la ponemos en On

```
; Development Value: On
; Production Value: Off
; https://php.net/display-errors
display_errors = On ←

; The display of errors which occur
; separately from display_errors. W
```

En ese mismo archivo buscamos la línea “display_startup_errors” y lo activamos igual que el anterior

```
; Development Value: On
; Production Value: Off
;https://php.net/display-startup-errors
display_startup_errors = On_ ←
;
; Besides displaying errors, PHP can also
; server-specific log, STDERR, or a location
```

Guardamos el archivo, reiniciamos el servicio de apache y para comprobarlo volvemos al navegador escribiendo 192.168.3.204/info.php y buscamos las configuraciones que acabamos de activar para comprobar que se han cambiado correctamente

<code>disable_functions</code>	<i>no value</i>	<i>no value</i>
<code>display_errors</code>	On	On
<code>display_startup_errors</code>	On	On
<code>doc_root</code>	<i>no value</i>	<i>no value</i>

Por último, volvemos a modificar el php.ini y cambiamos los valores de "memory_limit" para que tenga una memoria de 256M

```
; Maximum amount of memory a script  
; https://php.net/memory-limit  
memory_limit = 256M ←
```

Como hemos hecho anteriormente hacemos “[sudo service apache2 restart](#)” para reiniciar apache y comprobamos que se haya cambiado la memoria en nuestro navegador

max_multipart_body_parts	-1
memory_limit	256M
open_basedir	no value

5. MySQL

5.1 Instalación y configuración

Primero actualizamos las librerías de Linux con “`sudo apt update`” y “`sudo apt upgrade`” y después instalamos el servicio de MySQL con “`sudo apt install mysql-server`”. Para visualizar la versión que se ha instalado, usamos el comando “`mysql --version`”

```
miadmin@rsp-used:~$ mysql --version
mysql Ver 8.0.34-0ubuntu0.22.04.1 for Linux on x86_64 ((Ubuntu))
miadmin@rsp-used:~$ _
```

Para permitir la conexión desde cualquier IP necesitamos entrar al fichero de configuración de MySQL (“`sudo nano /etc/mysql/mysql.conf.d/mysqld.cnf`”) y comentar las siguientes líneas: “bind-address = 127.0.0.1” y “mysqlx-bind-address = 127.0.0.1”

```
# Instead of skip-networking the default is now to 1
# localhost which is more compatible and is not less
#bind-address      = 127.0.0.1 ←
#mysqlx-bind-address = 127.0.0.1 ←
#
# * Fine Tuning
#
```

Guardamos el archivo y reiniciamos mysql con “`sudo service mysql restart`” y después abrimos el puerto 3306 con “`sudo ufw allow 3306`”

```
miadmin@rsp-used:~$ sudo ufw status
Status: active

To                         Action      From
--                         ----       ---
22                         ALLOW      Anywhere
Apache                      ALLOW      Anywhere
9003                       ALLOW      Anywhere
3306                       ALLOW      Anywhere ←
22 (v6)                     ALLOW      Anywhere (v6)
Apache (v6)                  ALLOW      Anywhere (v6)
9003 (v6)                   ALLOW      Anywhere (v6)
3306 (v6)                   ALLOW      Anywhere (v6) ←
```

A continuación, ejecutaremos un script para realizar una configuración inicial de mysql con el comando “`sudo mysql_secure_installation`” y responderemos “yes” a todos los pasos excepto el de nivel de seguridad de contraseña que elegiremos el “LOW”

```
Normally, root should only be allowed to connect from
'localhost'. This ensures that someone cannot guess at
the root password from the network.

Disallow root login remotely? (Press y|Y for Yes, any other key for No) : y ←

By default, MySQL comes with a database named 'test' that
anyone can access. This is also intended only for testing,
and should be removed before moving into a production
environment.

Remove test database and access to it? (Press y|Y for Yes, any other key for No) : y ←
- Dropping test database...
Success.

- Removing privileges on test database...
Success.

Reloading the privilege tables will ensure that all changes
made so far will take effect immediately.

Reload privilege tables now? (Press y|Y for Yes, any other key for No) : y ←
Success.

All done!
```

Ahora volvemos a reiniciar MySQL e instalamos las librerías de php que son necesarias para la conexión de mysql con php con el comando “[sudo apt install libapache2-mod-php8.1 php8.1-mysql](#)” y reiniciamos apache. Ejecutamos mysql con “[sudo mysql](#)” y una vez dentro creamos el usuario administrador de las bases de datos de nuestro servidor. Para ello escribimos:

- [Create user ‘adminsql’@‘%’ identified by ‘paso’;](#)
- [Grant all privileges on *.* to ‘adminsql’@‘%’ with grant option;](#)

```
mysql> select user,plugin,host from user where user like 'adminsql';
+-----+-----+-----+
| user | plugin           | host |
+-----+-----+-----+
| adminsql | caching_sha2_password | % |
+-----+-----+-----+
1 row in set (0,00 sec)
```

En el caso de que no nos permita crear el usuario saltando el error de que la contraseña no cumple con los requisitos, tendremos que cambiar las variables globales de mysql para admitir contraseñas con 4 caracteres como mínimo. Para ello dentro de la consola de mysql escribiremos “[show variables like ‘validate_password%’](#)” y se mostrara la tabla de las variables que controlan los requisitos de las contraseñas de los usuarios.

```
mysql> show variables like 'validate_pass%';
+-----+-----+
| Variable_name          | Value |
+-----+-----+
| validate_password.changed_characters_percentage | 0      |
| validate_password.check_user_name                | ON     |
| validate_password.dictionary_file               |        |
| validate_password.length                       | 8      |
| validate_password.mixed_case_count            | 1      |
| validate_password.number_count                | 1      |
| validate_password.policy                      | MEDIUM |
| validate_password.special_char_count         | 1      |
+-----+-----+
8 rows in set (0,00 sec)
```

Una vez aquí, cambiaremos el valor de “length” y “policy” con “[set global validate_password.length=4;](#)” y “[set global validate_password.policy=LOW;](#)”

```
mysql> set global validate_password.length=4;
Query OK, 0 rows affected (0,00 sec)

mysql> set global validate_password.policy=LOW;
Query OK, 0 rows affected (0,00 sec)

mysql> show variables like 'validate_password%';
+-----+-----+
| Variable_name          | Value |
+-----+-----+
| validate_password.changed_characters_percentage | 0      |
| validate_password.check_user_name                | ON     |
| validate_password.dictionary_file               |        |
| validate_password.length                       | 4      | ←
| validate_password.mixed_case_count            | 1      |
| validate_password.number_count                | 1      |
| validate_password.policy                      | LOW    | ←
| validate_password.special_char_count         | 1      |
+-----+-----+
8 rows in set (0,00 sec)
```

5.2 PHPMyAdmin

Antes de la instalación de PHPMyAdmin tenemos que cambiar la contraseña del usuario de MySQL a “paso” para que cumpla con los requisitos requeridos. Para ello entraremos en MySQL con “sudo mysql”

- Filtramos las variables que contengan parámetros para las contraseñas con “show variables like ‘validate_password%’;”

```
mysql> show variables like 'validate_password%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| validate_password.changed_characters_percentage | 0      |
| validate_password.check_user_name                | ON     |
| validate_password.dictionary_file               |        |
| validate_password.length                      | 8      |
| validate_password.mixed_case_count            | 1      |
| validate_password.number_count                | 1      |
| validate_password.policy                     | MEDIUM |
| validate_password.special_char_count          | 1      |
+-----+-----+
8 rows in set (0,00 sec)
```

- Cambiamos los valores de longitud (para que tenga un máximo de 4 caracteres) y de política (para que sea menos segura) con “set global validate_password.length=4;” y “set global validate_password.policy=LOW;” respectivamente;

```
mysql> set global validate_password.length=4;
Query OK, 0 rows affected (0,00 sec)

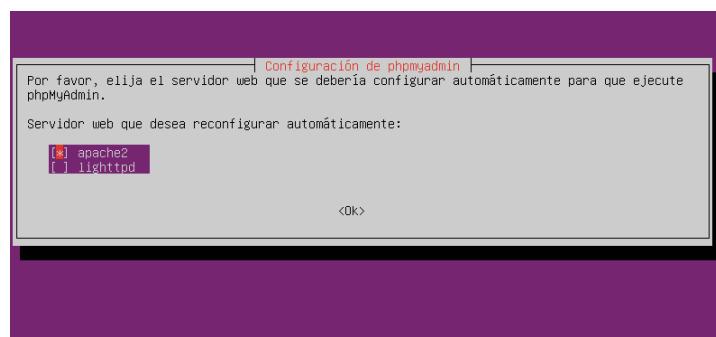
mysql> set global validate_password.policy=LOW;
Query OK, 0 rows affected (0,00 sec)
```

- Ahora creamos el usuario “create user ‘adminssql’@‘%’ identified by ‘paso’;” y le damos los permisos “grant all privileges on *.* to ‘adminssql’@‘%’ with grant option;”

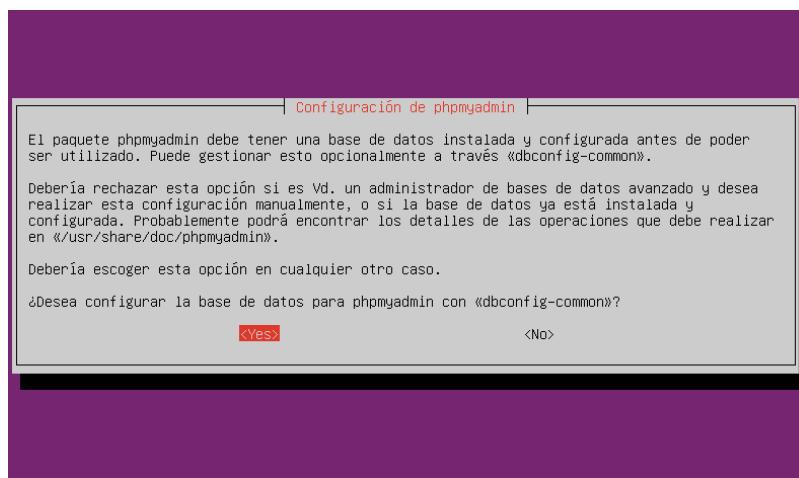
```
mysql> create user 'adminssql'@'%' identified by 'paso';
Query OK, 0 rows affected (0,00 sec)

mysql> grant all privileges on *.* to 'adminssql'@'%' with grant option;
Query OK, 0 rows affected (0,01 sec)
```

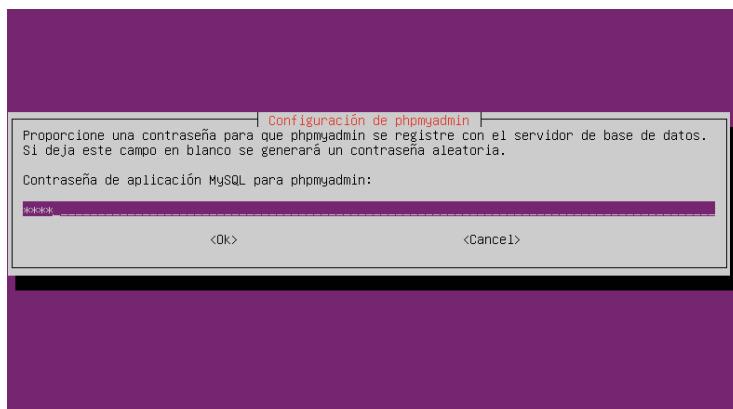
A continuación, salimos de mysql con “exit”. Para instalar PHPMyAdmin y los módulos que necesitamos escribiremos el comando “sudo apt install phpmyadmin php-mbstring php-zip php-gd php-json php-curl”. Cuando lo ejecutemos, aparecerá una interfaz del Ubuntu diferente en la que nos harán una serie de preguntas:



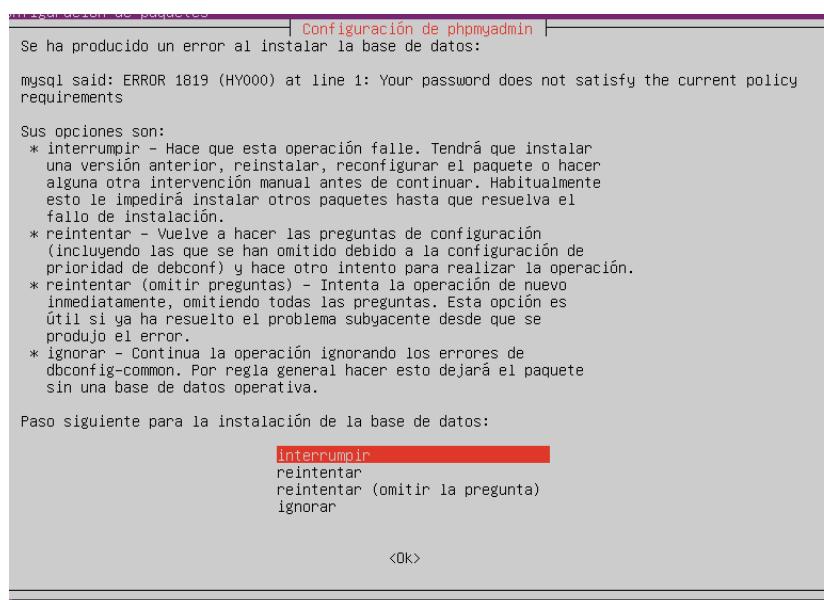
- Seleccionamos con la barra espaciadora la opción de *apache2*



- Ahora seleccionamos la opción yes para configurar la base de datos



- Introducimos la contraseña, en nuestro caso: *paso*



- Seleccionamos *interrumpir*

Una vez terminada la instalación, nos volvemos a conectar a mysql pero con el nuevo usuario "mysql -u adminssql -p"

```
miadmin@rsp-used:~$ mysql -u adminsql -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 15
Server version: 8.0.35-Ubuntu0.22.04.1 (Ubuntu)

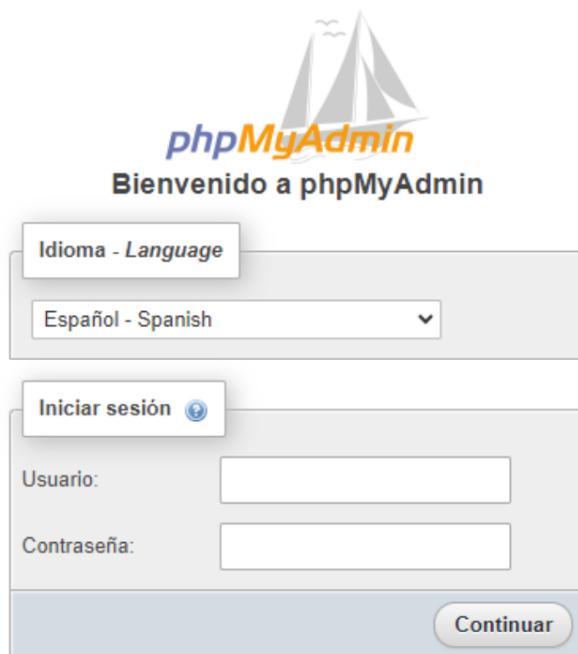
Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Ejecutamos el comando "uninstall component "file://component validate password";" y volvemos a cerrar mysql. Ahora es necesaria una purga de phpmyadmin "sudo apt purge phpmyadmin" y lo volvemos a instalar de igual manera con "sudo apt install phpmyadmin". Llegados a este punto volveremos a acceder a mysql con el usuario adminsql y volveremos a hacer el comando "install component "file://component validate password";". Para comprobar que todo ha ido bien, escribiremos en el navegador la IP de nuestro servidor seguida de "phpmyadmin" y debería aparecer esta ventana



6. XDebug

6.1 Instalación

Antes de instalar XDebug comprobaremos que no esté instalado anteriormente con “php -m |grep sdebug”, si la terminal no devuelve nada significa que ese modulo no está instalado. Para instarlo escribimos “sudo apt install php8.1-xdebug”

```
miadmin@rsp-used:~$ sudo apt install php8.1-xdebug
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
php8.1-xdebug ya está en su versión más reciente (3.1.2
0 actualizados, 0 nuevos se instalarán, 0 para eliminar
```

A continuación, cambiaremos de directorio a “cd /etc/php/8.1/mods-available/” y modificamos el archivo 20-xdebug.ini con “sudo nano 20-xdebug.ini” y escribimos las siguientes líneas:

```
zend_extension=xdebug.so
xdebug.extended_info=on
xdebug.remote_host=192.168.3.204
xdebug.remote_port=9003
xdebug.remote_handler=dbgp
xdebug.remote_autostart=on
xdebug.remote_enable = on
xdebug.mode=debug
xdebug.client_host=localhost
xdebug.client_port=9003
xdebug.idekey="netbeans-xdebug"
```

Para que se pueda conectar por el puerto 9003 tendremos que abrirlo mediante “sudo ufw allow 9003” y con “sudo ufw status” vemos los puertos que están abiertos

```
miadmin@rsp-used:/etc/php/8.1/apache2/conf.d$ sudo ufw allow 9003
Rule added
Rule added (v6)
miadmin@rsp-used:/etc/php/8.1/apache2/conf.d$ sudo ufw status
Status: active

To                         Action      From
--                         --         --
22                         ALLOW      Anywhere
Apache                      ALLOW      Anywhere
9003 ←                   ALLOW      Anywhere
22 (v6)                    ALLOW      Anywhere (v6)
Apache (v6)                 ALLOW      Anywhere (v6)
9003 (v6) ←               ALLOW      Anywhere (v6)
```

Para terminar, reiniciamos el servicio con “sudo service apache2 restart”.

7. Servidores DNS

7.1 Instalación del servidor DNS

Comprobamos que el paquete de DNS está instalado con “aptitude show bind9” y visualizamos el paquete con “dpkg -l | grep bind9”

```
miadmin@rsp-used:~$ aptitude show bind9
Command 'aptitude' not found, did you mean:
  command 'aptitude' from deb aptitude (0.8.13-3ubuntu1)
Try: sudo apt install <deb name>
miadmin@rsp-used:~$ dpkg -l | grep bind9
ii  bind9-dnsutils                               1:9.18.18-0ubuntu0.22.04.1      amd64      Client
ts provided with BIND 9
ii  bind9-host                                    1:9.18.18-0ubuntu0.22.04.1      amd64      DNS L
ookup Utility
ii  bind9-libs:amd64                             1:9.18.18-0ubuntu0.22.04.1      amd64      Share
d Libraries used by BIND 9
miadmin@rsp-used:~$
```

Procedemos a actualizar el sistema con “sudo apt update” e instalamos el servicio con el comando “sudo apt install bind9”. Comprobamos que el estado de bind9 esté activo con “sudo service bind9 status”

```
miadmin@rsp-used:~$ sudo service bind9 status
● named.service - BIND Domain Name Server
   Loaded: loaded (/lib/systemd/system/named.service; enabled; vendor preset: enabled)
     Active: active (running) since Thu 2023-12-14 15:36:19 CET; 1min 30s ago
       Docs: man:named(8)
   Process: 33551 ExecStart=/usr/sbin/named $OPTIONS (code=exited, status=0/SUCCESS)
 Main PID: 33553 (named)
    Tasks: 4 (limit: 2221)
      Memory: 6.8M
        CPU: 29ms
       CGroup: /system.slice/named.service
               └─33553 /usr/sbin/named -u bind

dic 14 15:36:19 rsp-used named[33553]: network unreachable resolving './DNSKEY/IN': 2001:500:2::c#53
dic 14 15:36:19 rsp-used named[33553]: network unreachable resolving './NS/IN': 2001:500:2::c#53
```

Ahora debemos editar la configuración de red, que se aloja en la siguiente ruta, el archivo que tenga como extensión ‘.yaml’, en mi caso se llama ‘00-installer-config.yaml’, para ello, usamos el comando “sudo nano /etc/netplan/00-installer-config.yaml” y modificamos lo siguiente:

```
GNU nano 5.2                                /etc/netplan/00-in
# This is the network config written by 'subiquity'
network:
  ethernets:
    enp0s3:
      addresses:
      - 192.168.1.204/24
      nameservers:
        addresses:
        - 192.168.1.204
        search: [rebeca.local]
      routes:
      - to: default
        via: 192.168.1.1
  version: 2
```

Aplicamos los cambios con “sudo netplan apply”, abrimos el puerto 53 para permitir la conexión con bind9 “sudo ufw allow 53” y comprobamos si funcionamiento haciendo ping a google

```

miadmin@rsp-used:~$ sudo netplan apply
** (generate:33813): WARNING **: 15:44:45.746: Permissions for /etc/netplan/00-installer-config.yaml
are too open. Netplan configuration should NOT be accessible by others.
WARNING:root:Cannot call Open vSwitch: ovsdb-server.service is not running.

** (process:33811): WARNING **: 15:44:46.037: Permissions for /etc/netplan/00-installer-config.yaml
are too open. Netplan configuration should NOT be accessible by others.

** (process:33811): WARNING **: 15:44:46.098: Permissions for /etc/netplan/00-installer-config.yaml
are too open. Netplan configuration should NOT be accessible by others.

** (process:33811): WARNING **: 15:44:46.098: Permissions for /etc/netplan/00-installer-config.yaml
are too open. Netplan configuration should NOT be accessible by others.
miadmin@rsp-used:~$ sudo ufw allow 53
Rule added
Rule added (v6)
miadmin@rsp-used:~$ ping www.google.es
PING www.google.es (216.58.215.163) 56(84) bytes of data.
64 bytes from mad41s07-in-f3.1e100.net (216.58.215.163): icmp_seq=1 ttl=115 time=12.0 ms
64 bytes from mad41s07-in-f3.1e100.net (216.58.215.163): icmp_seq=2 ttl=115 time=11.3 ms
64 bytes from mad41s07-in-f3.1e100.net (216.58.215.163): icmp_seq=3 ttl=115 time=11.9 ms
^C

```

7.2 Declaración de la zona directa

Modificaremos el archivo con “[sudo nano /etc/bind/named.conf.local](#)” añadiendo una nueva zona como se muestra a continuación:

```

// 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 "rebeca.local" {
    type master;
    file "/etc/bind/db.rebeca.local";
};_

```



Comprobamos los ficheros de configuracion con “[sudo named-checkconf](#)”

```

miadmin@rsp-used:~$ sudo named-checkconf

```

A continuación, crearemos una copia de seguridad con el comando “[sudo cp /etc/bind/db.local /etc/bind/db.rebeca.local](#)” y editaremos esa copia para que resulte de la siguiente manera:

```

; BIND data file for rebeca.local
;
$TTL    604800
@       IN      SOA     rsp-used.rebeca.local. marodriguez.educa.jcyl.es. (
                          1           ; Serial
                          604800      ; Refresh
                          86400       ; Retry
                          2419200     ; Expire
                          3600 )      ; Negative Cache TTL
;
@       IN      NS      rsp-used.rebeca.local.

; Registro Host
@           IN      A      192.168.1.204
rsp-used    IN      A      192.168.1.204

; Registro Alias
daw201      IN      CNAME   rsp-used.rebeca.local.
daw202      IN      CNAME   rsp-used.rebeca.local.
daw204      IN      CNAME   rsp-used.rebeca.local.

```

Comprobamos que la sintaxis esta correcta con “[sudo named-checkzone rebeca.local /etc/bind/db.rebeca.local](#)” y reiniciamos el servicio con “[sudo service bind9 restart](#)”

```
miadmin@rsp-used:~$ sudo named-checkzone rebeca.local /etc/bind/db.rebeca.local
zone rebeca.local/IN: loaded serial 1
OK
miadmin@rsp-used:~$
```



```
miadmin@rsp-used:~$ sudo service bind9 restart
miadmin@rsp-used:~$
```

7.3 Declaración de la zona inversa

Volvemos a repetir los pasos, modificaremos el archivo con “[sudo nano /etc/bind/named.conf.local](#)” y escribimos:

```
    type master;
    file "/etc/bind/db.3.168.192.in-addr.arpa";
};
```

Como medida de seguridad hacemos una copia de este archivo “[sudo cp /etc/bind/db.rebeca.local /etc/bind/db.3.168.192.in-addr.arpa](#)” y lo editamos “[sudo nano /etc/bind/db.3.168.192.in-addr.arpa](#)” de la siguiente forma:

```
; BIND data for 3.168.192.in-addr.arpa
;

$TTL    604800

@       IN      SOA    rsp-used.rebeca.local. root.localhost. (
                      1           ; Serial
                      604800      ; Refresh
                      86400       ; Retry
                      2419200     ; Expire
                      3600        ; Negative Cache TTL
);
@       IN      NS     rsp-used.rebeca.local.
; Registros punteros o pointers
204      IN      PTR    rsp-used.rebeca.local.
204      IN      PTR    rebeca.local.
```

Comprobamos la zona con “[sudo named-checkzone 204.168.192.in-addr.arpa /etc/bind/db.3.168.192.in-addr.arpa](#)” y reiniciamos bind9

```
miadmin@rsp-used:~$ sudo named-checkzone 204.168.192.in-addr.arpa /etc/bind/db.3.168.192.in-addr.arpa
zone 204.168.192.in-addr.arpa/IN: loaded serial 1
OK
miadmin@rsp-used:~$ sudo service bind9 restart
miadmin@rsp-used:~$
```

7.4 Comprobación de funcionamiento

Escribimos en la terminal “host rsp-used.rebeca.local”

```
miadmin@rsp-used:~$ host rsp-used.rebeca.local
rsp-used.rebeca.local has address 192.168.1.204
```

Ahora comprobamos con este nombre “host daw204.rebeca.local”

```
miadmin@rsp-used:~$ host daw204.rebeca.local
daw204.rebeca.local is an alias for rsp-used.rebeca.local.
rsp-used.rebeca.local has address 192.168.1.204
```

Y por último “host rebeca.local”

```
miadmin@rsp-used:~$ host rebeca.local
rebeca.local has address 192.168.1.204
```

A continuación, comprobamos con el comando “nslookup” -> “rsp-used.rebca.local”

```
miadmin@rsp-used:~$ nslookup
> rsp-used.rebeca.local
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
Name:   rsp-used.rebeca.local
Address: 192.168.1.204
>
```

Ahora hacemos “daw204.rebeca.local”

```
> daw204.rebeca.local
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
daw204.rebeca.local      canonical name = rsp-used.rebeca.local.
Name:   rsp-used.rebeca.local
Address: 192.168.1.204
```

Y por último “rebeca.local”

```
> rebeca.local
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
Name:   rebeca.local
Address: 192.168.1.204
```

8. Cuentas de desarrollo y hosting virtual

8.1 Creación de operadorweb

Tenemos que crear un usuario que se encargue de conectarse al servidor mediante http/s desde la maquina anfitriona que usaremos de cliente. Para eso usamos el comando “[sudo adduser --home /var/www/html --no-create-home --ingroup www-data operadorweb](#)” que creara el usuario operadorweb en el grupo www-data (grupo por defecto de usuarios de apache) y con el home en el directorio html.

```
miadmin@rsp-used:~$ sudo adduser --home /var/www/html --no-create-home --ingroup www-data operadorwe
b
Adding user `operadorweb' ...
Adding new user `operadorweb' (1002) with group `www-data' ...
Not creating home directory `/var/www/html'.
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for operadorweb
Enter the new value, or press ENTER for the default
    Full Name []:
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n]
miadmin@rsp-used:~$ -
```

Para confirmar que se ha creado usamos el comando “[id operadorweb](#)” y se mostrará la información sobre ese usuario.

```
miadmin@rsp-used:~$ id operadorweb
uid=1002(operadorweb) gid=33(www-data) groups=33(www-data)
```

A continuación, cambiamos el propietario del directorio /var/www/html para que sea operadorweb con “[sudo chown -R operadorweb:www-data /var/www/html](#)” y seguido de esto cambiamos los permisos del archivo que se encuentra en ese directorio para que pueda ser legible, modificado y ejecutado por ese usuario con “[sudo chmod -R 2775 /var/www/html](#)”

```
miadmin@rsp-used:~$ sudo chown -R operadorweb:www-data /var/www/html
miadmin@rsp-used:~$ sudo chmod -R 2775 /var/www/html

miadmin@rsp-used:~$ ll /var/www/html/
total 20
drwxrwsr-x 2 operadorweb www-data 4096 oct  4 00:25 ../
drwxr-xr-x 3 root      root     4096 oct  4 00:25 ...
-rwxrwsr-x 1 operadorweb www-data 10671 oct  4 00:25 index.html*
```

8.2 Creación del grupo ftpuser y enjaulamiento

Creamos el grupo ftpuser al que pertenecerá el operadorweb con el comando “[sudo groupadd ftpuser](#)”

```
miadmin@rsp-used:~$ sudo groupadd ftpuser
[sudo] password for miadmin:
```

Después de esto, cambiaremos de directorio a “[cd /etc/ssh](#)” y crearemos una copia de seguridad del fichero de configuracion sshd_config.d con el comando “[sudo cp -r ssh_config.d sshd_config.d.backup](#)”. Una vez creada la copia, editaremos el archivo original (con “[sudo nano sshd config](#)”) de la siguiente manera:

```
# override default of no subsystems
#Subsystem      sftp    /usr/lib/openssh/sftp-server
Subsystem sftp internal-sftp ←

# Example of overriding settings on a per-user basis
#Match User anoncvs
#      X11Forwarding no
#      AllowTcpForwarding no
#      PermitTTY no
#      ForceCommand cvs server
Match Group ftpuser ←
ChrootDirectory %h
ForceCommand internal-sftp -u 2
AllowTcpForwarding yes
PermitTunnel no
X11Forwarding no_
```

- Comentamos la línea “Subsystem sftp /usr/lib/openssh/sftp-server” que viene por defecto y escribimos la que viene a continuación “Subsystem sftp internal-sftp”
- Al final del fichero agregamos las líneas de texto que vienen indicadas por la segunda flecha

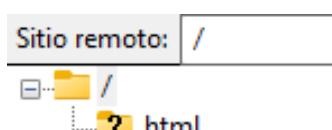
Una vez guardada la configuración, reiniciaremos ssh con “sudo service ssh restart”. Esta configuración nos permite controlar el servicio SFTP con el grupo de usuarios “ftpuser”. Ahora procedemos a enjaular a nuestro usuario operadorweb en el directorio “/var/www”, esto significa que el usuario operadorweb nunca va a tener acceso a los archivos y directorios que estén por encima del directorio en el que se encuentra enjaulado. Para ello debemos parar el servicio de apache con “sudo service apache2 stop” y después usaremos el comando “sudo usermod operadorweb -d /var/www” que modifica el usuario para que su home sea /var/www

```
miadmin@rsp-used:/etc/ssh$ sudo usermod operadorweb -d /var/www
miadmin@rsp-used:/etc/ssh$ _
```

A continuación, usamos el comando “sudo chown root:root /var/www” para cambiar el propietario de ese directorio y seguido de esto con “sudo chmod 555 /var/www” y “sudo usermod -G ftpuser operadorweb” le daremos permisos de lectura y ejecución y agregaremos el usuario operadorweb al grupo ftpuser.

```
drwxrwxrwt  6 root root    4096 oct 30 18:48 tmp/
dr-xr-xr-x  3 root root    4096 oct  4 00:25 www/ ←
miadmin@rsp-used:/var$
```

Si queremos comprobar que el enjaulamiento está correcto, nos conectamos desde FileZilla el servidor con el usuario operadorweb y si el directorio raíz contiene /www significa que ha ido correctamente.



Ahora haremos un ejemplo de creación de un usuario nuevo DAW204 y de su enjaulamiento. Usamos el comando “sudo useradd -g www-data -G ftpuser -m -d /var/www/DAW201 DAW201” para crear un usuario con directorio principal el especificado y dentro de los grupos especificados.

Comprobamos que esta creado con “id DAW201”.

Cambiamos el propietario de la carpeta DAW201 para que sea de root con “sudo chown root:root /var/www/DAW201” y le cambiamos los permisos con “sudo chmod 555 /var/www/DAW201”

Ahora creamos la carpeta donde estará alojado el proyecto “sudo mkdir /var/www/DAW201/public_html” y le damos los permisos sobre esa carpeta con “sudo chmod 2775 /var/www/DAW201/public_html”

Cambiamos el propietario de esa carpeta con “ht”

Comprobamos desde filezilla

9. Apache Tomcat

9.1 Instalación en Ubuntu

Antes de instalar Tomcat actualizaremos el sistema con “[sudo apt update & sudo apt upgrade](#)”

```
Preparando para desempaquetar .../systemd-hue-hwdb_249.11.3_amd64.deb
Desempaquetando systemd-hue-hwdb (249.11.3) sobre (249.11.3) ...
Preparando para desempaquetar .../55-cloud-init_23.3.3-0ubuntu0~22.04.
Desempaquetando cloud-init (23.3.3-0ubuntu0~22.04.1) sobre (23.3.1-0ub
Preparando para desempaquetar .../56-kpartx_0.8.8-1ubuntu1.22.04.4_amd64.
Desempaquetando kpartx (0.8.8-1ubuntu1.22.04.4) sobre (0.8.8-1ubuntu1.
Preparando para desempaquetar .../57-multipath-tools_0.8.8-1ubuntu1.22.04.
Desempaquetando multipath-tools (0.8.8-1ubuntu1.22.04.4) sobre (0.8.8-1
Configurando libapparmor1:amd64 (3.0.4-2ubuntu2.3) ...
Configurando cloud-init (23.3.3-0ubuntu0~22.04.1) ...
Configurando apt-utils (2.4.11) ...
Configurando irqbalance (1.8.0-1ubuntu0.2) ...
Configurando linux-firmware (20220329.git681281e4-0ubuntu3.24) ...
update-initramfs: Generating /boot/initrd.img-5.15.0-89-generic

Progreso: [ 59%] [#########################################.....]
```

En primer lugar, instalamos el JDK de java con el comando “[sudo apt install default-jdk](#)” y para comprobar su instalación usamos “[java -version](#)”

```
miadmin@rsp-used:~$ java -version
openjdk version "11.0.21" 2023-10-17
OpenJDK Runtime Environment (build 11.0.21+9-post-Ubuntu-0ubuntu122.04)
OpenJDK 64-Bit Server VM (build 11.0.21+9-post-Ubuntu-0ubuntu122.04, mixed mode, sharing)
miadmin@rsp-used:~$ _
```

Ejecutamos el comando “[sudo update-java-alternatives -l](#)”

```
miadmin@rsp-used:~$ sudo update-java-alternatives -l
java-1.11.0-openjdk-amd64      1111      /usr/lib/jvm/java-1.11.0-openjdk-amd64
miadmin@rsp-used:~$ _
```

A continuación, vamos a crear un grupo llamado tomcat con “[sudo groupadd tomcat](#)” al que pertenecerá un usuario tomcat “[sudo useradd -s /bin/false -g tomcat -d /opt/tomcat tomcat](#)” y también creamos una carpeta en la que se instalará el servidor Apache Tomcat en /opt “[sudo mkdir /opt/tomcat](#)”

```
miadmin@rsp-used:~$ sudo groupadd tomcat
miadmin@rsp-used:~$ sudo useradd -s /bin/false -g tomcat -d /opt/tomcat tomcat
miadmin@rsp-used:~$ sudo mkdir /opt/tomcat
miadmin@rsp-used:~$ _
```

Cambiamos de directorio “[cd /tmp](#)” y ejecutamos el siguiente comando para descargar el comprimido de tomcat “[wget https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.85/bin/apache-tomcat-9.0.85.tar.gz](#)”

```
miadmin@rsp-used:/tmp$ wget https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.85/bin/apache-tomcat-9.0.85.tar.gz
--2024-01-19 08:54:11-- https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.85/bin/apache-tomcat-9.0.85.tar.gz
Resolving dlcdn.apache.org (dlcdn.apache.org)... 151.101.2.132, 2a04:4e42::644
Connecting to dlcdn.apache.org (dlcdn.apache.org)|151.101.2.132|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 11809177 (11M) [application/x-gzip]
Saving to: 'apache-tomcat-9.0.85.tar.gz'

apache-tomcat-9.0.85.tar 100%[=====] 11,26M 31,7MB/s in 0,4s
2024-01-19 08:54:12 (31,7 MB/s) - 'apache-tomcat-9.0.85.tar.gz' saved [11809177/11809177]
```

Con “`sudo tar -xzvf apache-tomcat-9.0.85.tar.gz -c /opt/tomcat --strip-components=1`” descomprimimos el archivo especificándole la carpeta en la que debe situarse y mostramos el contenido de esa carpeta para asegurarnos de que todo ha ido bien

```
miadmin@rsp-used:/tmp$ cd /opt/tomcat
miadmin@rsp-used:/opt/tomcat$ ll
total 156
drwxr-xr-x 9 root root 4096 ene 19 08:56 .
drwxr-xr-x 3 root root 4096 ene 19 08:49 ../
drwxr-x--- 2 root root 4096 ene 19 08:56 bin/
-rw-r----- 1 root root 20307 ene 5 09:28 BUILDING.txt
drwxr----- 2 root root 4096 ene 5 09:28 conf/
-rw-r----- 1 root root 6210 ene 5 09:28 CONTRIBUTING.md
drwxr-x--- 2 root root 4096 ene 19 08:56 lib/
-rw-r----- 1 root root 57092 ene 5 09:28 LICENSE
drwxr-x--- 2 root root 4096 ene 5 09:28 logs/
-rw-r----- 1 root root 2333 ene 5 09:28 NOTICE
-rw-r----- 1 root root 3398 ene 5 09:28 README.md
-rw-r----- 1 root root 6901 ene 5 09:28 RELEASE-NOTES
-rw-r----- 1 root root 16505 ene 5 09:28 RUNNING.txt
drwxr-x--- 2 root root 4096 ene 19 08:56 temp/
drwxr-x--- 7 root root 4096 ene 5 09:28 webapps/
drwxr-x--- 2 root root 4096 ene 5 09:28 work/
```

Ahora ejecutamos los siguientes comandos para cambiar los propietarios de los directorios de tomcat

- `Sudo chgrp -R tomcat /opt/tomcat`
- `Sudo chmod -R g+r /opt/tomcat/conf/`
- `Sudo chmod -R u+x /opt/tomcat/bin/`
- `Sudo chown -R tomcat /opt/tomcat`

```
miadmin@rsp-used:/opt/tomcat$ sudo chgrp -R tomcat /opt/tomcat
miadmin@rsp-used:/opt/tomcat$ sudo chmod -R g+r /opt/tomcat/conf/
miadmin@rsp-used:/opt/tomcat$ sudo chmod -R u+x /opt/tomcat/bin/
miadmin@rsp-used:/opt/tomcat$ sudo chown -R tomcat /opt/tomcat
miadmin@rsp-used:/opt/tomcat$
```

Modificamos el archivo de configuración del servicio de tomcat con el comando “`sudo nano /etc/systemd/system/tomcat.service`” para que resulte como se muestra en la captura:

```
GNU nano 6.2                               /etc/systemd/system/tomcat.service *
[Unit]
Description=Apache Tomcat Web Application Container
After=network.target
[Service]
Type=forking
Environment="JAVA_HOME=/usr/lib/jvm/java-1.11.0-openjdk-amd64"
Environment="CATALINA_PID=/opt/tomcat/temp/tomcat.pid"
Environment="CATALINA_HOME=/opt/tomcat"
Environment="CATALINA_BASE=/opt/tomcat"
Environment="CATALINA_OPTS=-Xms512M -Xmx1024M -server -XX:+UseParallelGC"
Environment="JAVA_OPTS=-Djava.awt.headless=true -Djava.security.egd=file:/dev/urandom"
ExecStart=/opt/tomcat/bin/startup.sh
ExecStop=/opt/tomcat/bin/shutdown.sh
User=tomcat
Group=tomcat
UMask=0007
RestartSec=10
Restart=always
[Install]
WantedBy=multi-user.target
```

Ejecutamos la orden “`sudo systemctl daemon-reload`” que nos permite recargar todos los servicios de nuevo, seguido de esto arrancamos Tomcat “`sudo systemctl start tomcat`” y comprobamos su estado “`sudo systemctl status tomcat`”

```
miadmin@rsp-used:~$ miadmin@rsp-used:~$ sudo systemctl daemon-reload
miadmin@rsp-used:~$ sudo systemctl start tomcat
miadmin@rsp-used:~$ sudo systemctl status tomcat
● tomcat.service - Apache Tomcat Web Application Container
   Loaded: loaded (/etc/systemd/system/tomcat.service; disabled; vendor preset: enabled)
     Active: active (running) since Fri 2024-01-19 09:09:05 CET; 11s ago
       Process: 52179 ExecStart=/opt/tomcat/bin/startup.sh (code=exited, status=0/SUCCESS)
      Main PID: 52186 (java)
         Tasks: 17 (limit: 2221)
        Memory: 123.8M
          CPU: 10.201s
        CGroup: /system.slice/tomcat.service
                 └─52186 /usr/lib/jvm/java-1.11.0-openjdk-amd64/bin/java -Djava.util.logging.config.file=/opt/tomcat/conf/1

ene 19 09:09:05 rsp-used systemd[1]: Starting Apache Tomcat Web Application Container...
ene 19 09:09:05 rsp-used startup.sh[52179]: Tomcat started.
ene 19 09:09:05 rsp-used systemd[1]: Started Apache Tomcat Web Application Container.
lines 1-14/14 (END)
```

Habilitamos el servicio “`sudo systemctl enable tomcat`” y comprobamos que está activo con “`ps -ef | grep tomcat`”

```
miadmin@rsp-used:~$ sudo systemctl enable tomcat
Created symlink /etc/systemd/system/multi-user.target.wants/tomcat.service → /etc/systemd/system/tomcat.service.
miadmin@rsp-used:~$ ps -ef | grep tomcat
tomcat    52186   1 22 09:09 ?        00:00:14 /usr/lib/jvm/java-1.11.0-openjdk-amd64/bin/java -Djava.util.logging.config.file=/opt/tomcat/conf/logging.properties -Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager -Djava.awt.headless=true -Djava.security.egd=file:/dev/urandom -Djdk.tls.ephemeralDHKeySize=2048 -Djava.protocol.handler.pkgs=org.apache.catalina.webresources -Dorg.apache.catalina.security.SecurityListener.UMASK=0027 -Xms512M -Xmx1024M -server -XX:+UseParallelGC -Dignore.endorsed.dirs= -classpath /opt/tomcat/bin/bootstrap.jar:/opt/tomcat/bin/tomcat-juli.jar -Dcatalina.base=/opt/tomcat -Dcatalina.home=/opt/tomcat -Djava.io.tmpdir=/opt/tomcat/temp org.apache.catalina.startup.Bootstrap start
trap start
miadmin    52343  52186  0 09:10 pts/0    00:00:00 grep --color=auto tomcat
```

ahora listamos los puertos que están escuchando “`ss -ltn`” para encontrar el que utiliza tomcat y lo abrimos con “`sudo ufw allow 8080`”

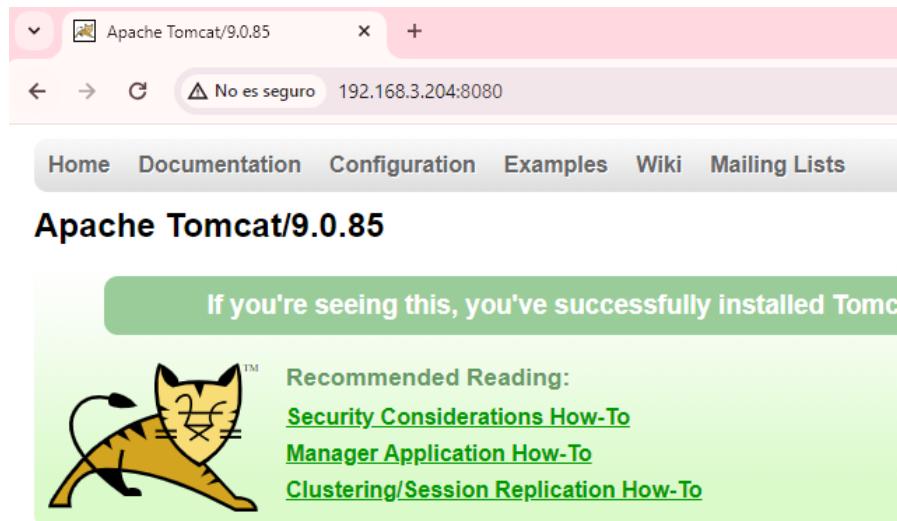
```
miadmin@rsp-used:~$ ss -ltn
State      Recv-Q      Send-Q      Local Address:Port          Peer Address:Port      Process
LISTEN      0            10          192.168.3.204:53        0.0.0.0:*
LISTEN      0            10          127.0.0.1:53           0.0.0.0:*
LISTEN      0            4096        127.0.0.53%lo:53        0.0.0.0:*
LISTEN      0            128          0.0.0.0:22             0.0.0.0:*
LISTEN      0            5            127.0.0.1:953          0.0.0.0:*
LISTEN      0            100          *:8080                  *:*
LISTEN      0            511          *:80                   *:*
LISTEN      0            10          [::1]:53                [::]:*
LISTEN      0            10          [fe80::a00:27ff:fea2:264e] %enp0s3:53      [::]:*
LISTEN      0            128          [::]:22                 [::]:*
LISTEN      0            5            [::1]:953              [::]:*
LISTEN      0            511          *:443                  *:*
LISTEN      0            70            *:33060                *:*
LISTEN      0            1            [::ffff:127.0.0.1]:8005  *:*
LISTEN      0            151          *:3306                *:*
```



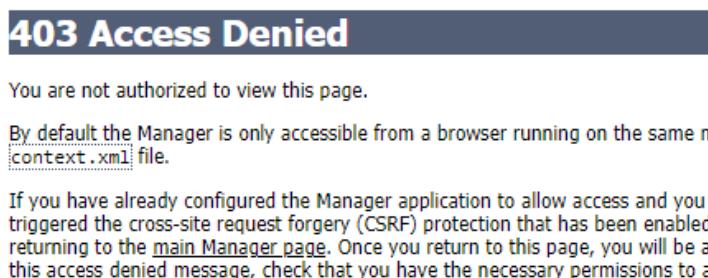
```
miadmin@rsp-used:~$ sudo ufw allow 8080
Rule added
Rule added (v6)
miadmin@rsp-used:~$ sudo ufw status
Status: active

To                         Action      From
--                         ----      --
22                         ALLOW      Anywhere
Apache                      ALLOW      Anywhere
9003                      ALLOW      Anywhere
3306                      ALLOW      Anywhere
443                       ALLOW      Anywhere
53                        ALLOW      Anywhere
8080                      ALLOW      Anywhere
22 (v6)                    ALLOW      Anywhere (v6)
Apache (v6)                 ALLOW      Anywhere (v6)
9003 (v6)                  ALLOW      Anywhere (v6)
3306 (v6)                  ALLOW      Anywhere (v6)
443 (v6)                   ALLOW      Anywhere (v6)
53 (v6)                    ALLOW      Anywhere (v6)
8080 (v6)                  ALLOW      Anywhere (v6)
```

Para comprobar el correcto funcionamiento de nuestro servidor Tomcat, escribimos la IP de nuestra máquina virtual en un explorador especificándole con dos puntos detrás el puerto por el cual actúa tomcat



En el apartado de “Manager App” que aparece a la derecha podemos encontrar nuestras aplicaciones, en nuestro caso aun no tenemos ninguna app ni el acceso autorizado para poder verlas



Modificamos el fichero tomcat-users.xml con “[sudo nano /opt/tomcat/conf/tomcat-users.xml](#)” y descomentamos las siguientes lineas:

```
<!--
<role rolename="tomcat"/>
<role rolename="role1"/>
<user username="tomcat" password="" roles="tomcat"/>
<user username="both" password="" roles="tomcat,role1"/>
<user username="role1" password="" roles="role1"/>
-->
<role rolename="tomcat" />
<user username="tomcat" password="paso" roles="manager-gui,admin-gui" />
</tomcat-users>
```

A continuación, listamos el contenido de la carpeta webapps “`sudo ls /opt/tomcat/webapps`” y de los directorios que aparecen modificaremos el `context.xml` de manager, host-manager, docs y examples de la siguiente manera:

```
miadmin@rsp-used:~$ sudo ls /opt/tomcat/webapps
docs examples host-manager manager ROOT
```

- `sudo nano /opt/tomcat/webapps/manager/META-INF/context.xml`
- `sudo nano /opt/tomcat/webapps/host-manager/META-INF/context.xml`

```
GNU nano 6.2                               /opt/tomcat/webapps/manager/META-INF/context.xml *
<?xml version="1.0" encoding="UTF-8"?>
<!--
Licensed to the Apache Software Foundation (ASF) under one or more
contributor license agreements. See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
-->
<Context antiResourceLocking="false" privileged="true" >
  <CookieProcessor className="org.apache.tomcat.util.http.Rfc6265CookieProcessor"
    sameSiteCookies="strict" />
<!--  <Valve className="org.apache.catalina.valves.RemoteAddrValve"
    allow="127.\d+.\d+.\d+|::1|0:0:0:0:0:0:1" /> -->
  <Manager sessionAttributeValueClassNameFilter="java\.lang.(?:Boolean|Integer|Long|Number|String)|org\.apache\.catali
</Context>
```

- `sudo nano /opt/tomcat/webapps/docs/META-INF/context.xml`

```
GNU nano 6.2                               /opt/tomcat/webapps/docs/META-INF/context.xml *
<?xml version="1.0" encoding="UTF-8"?>
<!--
Licensed to the Apache Software Foundation (ASF) under one or more
contributor license agreements. See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
-->
<Context antiResourceLocking="false" >
<!--  <Valve className="org.apache.catalina.valves.RemoteAddrValve"
    allow="127.\d+.\d+.\d+|::1|0:0:0:0:0:0:1" /> -->
</Context>
```

- `sudo nano /opt/tomcat/webapps/examples/META-INF/context.xml`

```
limitations under the License.
-->
<Context>
  <CookieProcessor className="org.apache.tomcat.util.http.Rfc6265CookieProcessor"
    sameSiteCookies="strict" />
<!--  <Valve className="org.apache.catalina.valves.RemoteAddrValve"
    allow="127.\d+.\d+.\d+|::1|0:0:0:0:0:0:1" /> -->
</Context>
```

Si volvemos a ir a nuestro buscador ya no aparecerá el error de autentificación, aparecerá otra presentación con las aplicaciones que hemos modificado

The screenshot shows a web browser window with the URL `192.168.3.204:8080/manager/html`. The page title is "Gestor de Aplicaciones Web de Tomcat". At the top left, there is a cartoon cat icon. On the right, the Apache Software Foundation logo is displayed. Below the title, there is a message box containing "Mensaje: OK". The main content area has tabs: "Listar Aplicaciones" (selected), "Ayuda HTML de Gestor", "Ayuda de Gestor", and "Estado de Servidor". The "Aplicaciones" table lists the following entries:

Ruta	Versión	Nombre a Mostrar	Ejecutándose	Sesiones	Comandos
<code>/</code>	Ninguno especificado	Welcome to Tomcat	true	0	Arrancar Parar Recargar Replegar <input type="button" value="Expirar sesiones sin trabajar ≥ 30 minutos"/>
<code>/docs</code>	Ninguno especificado	Tomcat Documentation	true	0	Arrancar Parar Recargar Replegar <input type="button" value="Expirar sesiones sin trabajar ≥ 30 minutos"/>
<code>/examples</code>	Ninguno especificado	Servlet and JSP Examples	true	0	Arrancar Parar Recargar Replegar <input type="button" value="Expirar sesiones sin trabajar ≥ 30 minutos"/>
<code>/host-manager</code>	Ninguno especificado	Tomcat Host Manager Application	true	1	Arrancar Parar Recargar Replegar <input type="button" value="Expirar sesiones sin trabajar ≥ 30 minutos"/>
<code>/manager</code>	Ninguno especificado	Tomcat Manager Application	true	1	Arrancar Parar Recargar Replegar <input type="button" value="Expirar sesiones sin trabajar ≥ 30 minutos"/>

9.2 Instalación en Windows

Para la instalación de Tomcat en Windows, accederemos desde un buscador a siguiente enlace y descargaremos la versión de 64-bit: <https://tomcat.apache.org/download-90.cgi>

The screenshot shows the Apache Tomcat download page for version 9.0.85. The page header says "9.0.85". Below it, a note says "Please see the [README](#) file for packaging information. It explains what every distribution contains". A section titled "Binary Distributions" lists several download options:

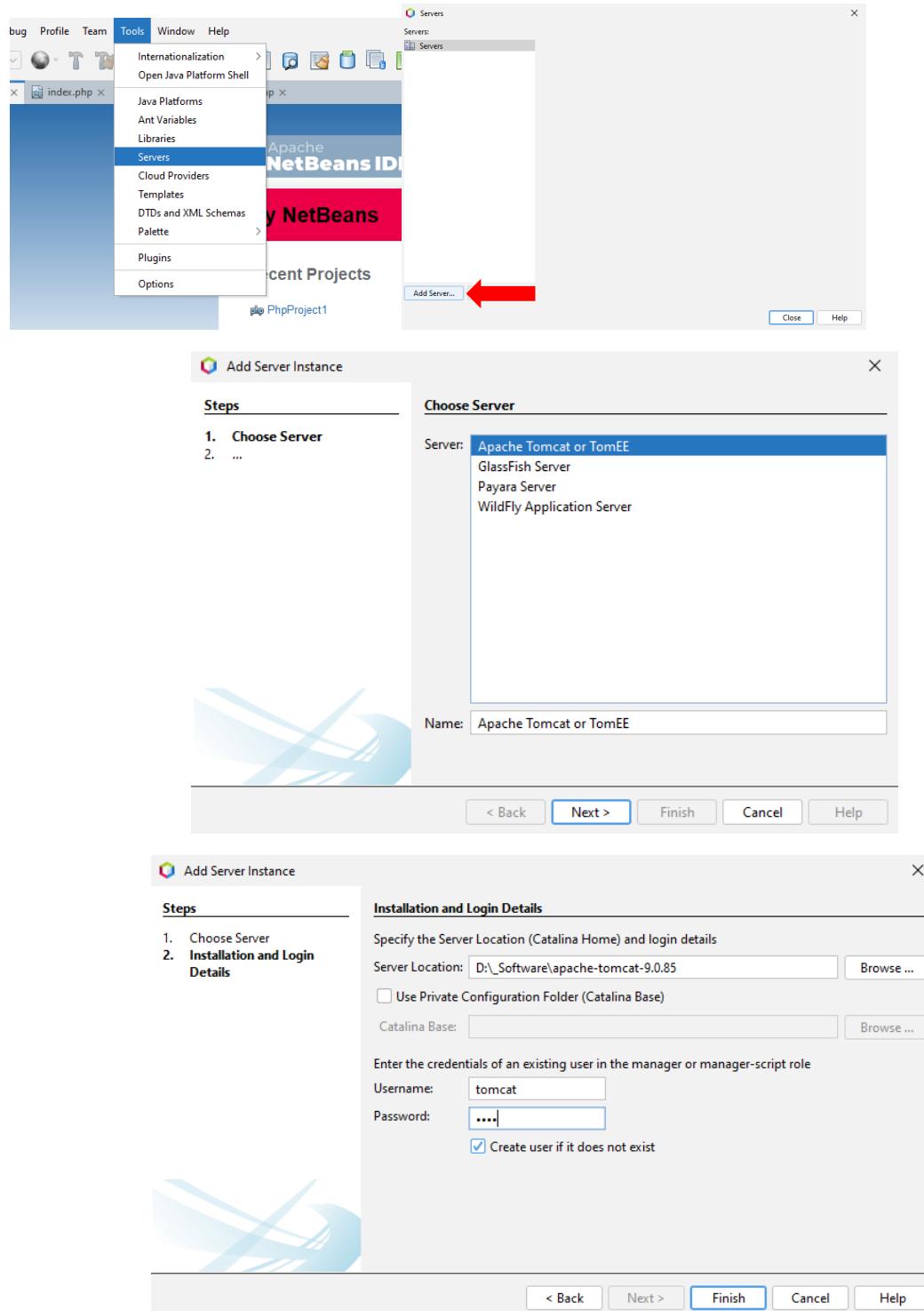
- Core:
 - [zip \(pgp, sha512\)](#)
 - [tar.gz \(pgp, sha512\)](#)
 - [32-bit Windows zip \(pgp, sha512\)](#)
 - [64-bit Windows zip \(pgp, sha512\) ←](#)
 - [32-bit/64-bit Windows Service Installer \(pgp, sha512\)](#)
- Full documentation:
 - [tar.gz \(pgp, sha512\)](#)
- Deployer:
 - [zip \(pgp, sha512\)](#)

Descomprimimos el archivo en la carpeta “_Software” de nuestro disco D:

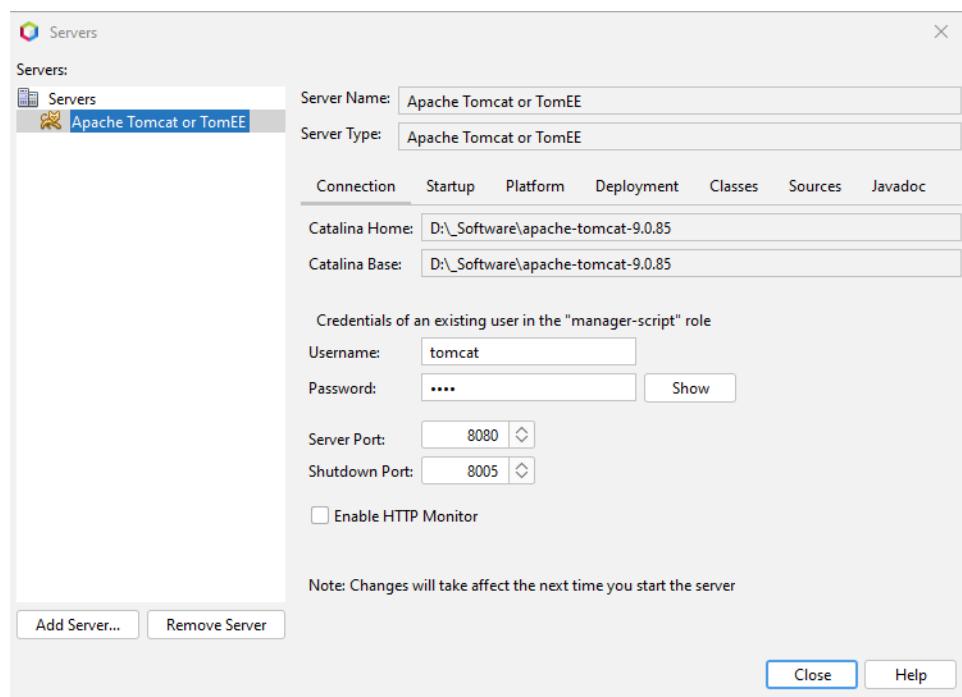
The screenshot shows a Windows File Explorer window with the path "DATOS (D:) > _Software". The contents of the folder are listed in a table:

Nombre	Fecha de modificación
apache-tomcat-9.0.85 ←	23/01/2024 8:47
xampp	22/09/2023 9:21
7z2021-arm64.exe	08/06/2023 12:54
Apache-NetBeans-18-bin-windows-x64....	15/06/2023 10:33

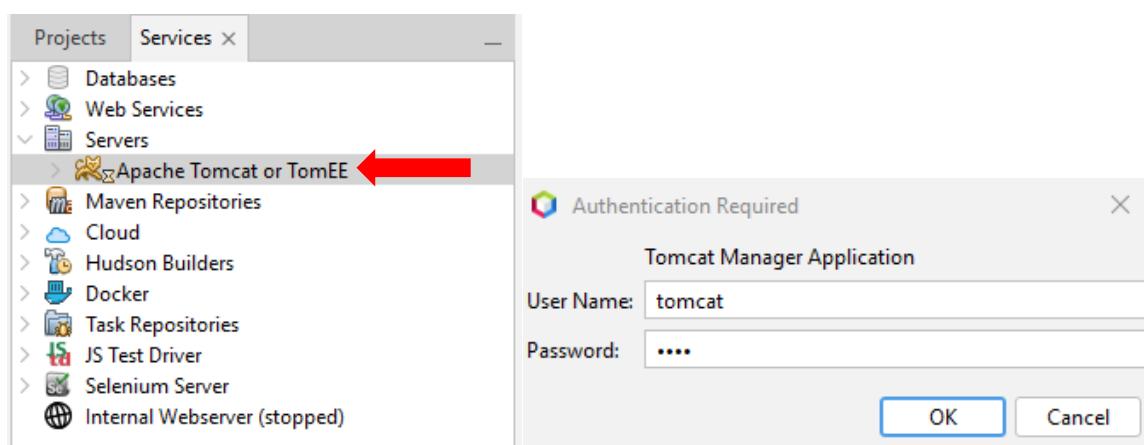
Desde NetBeans accedemos a “Tools>Servers” y agregamos el server descargado configurándolo de la siguiente manera



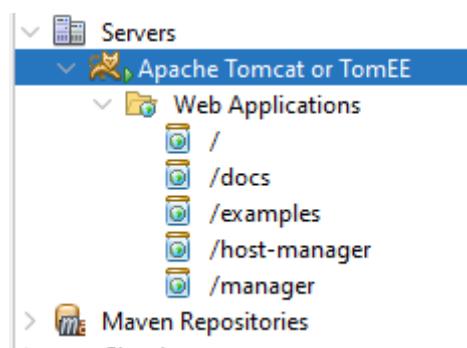
- Server Location: seleccionamos la carpeta en la que hemos descomprimido el tomcat
- Username: tomcat
- Password: paso



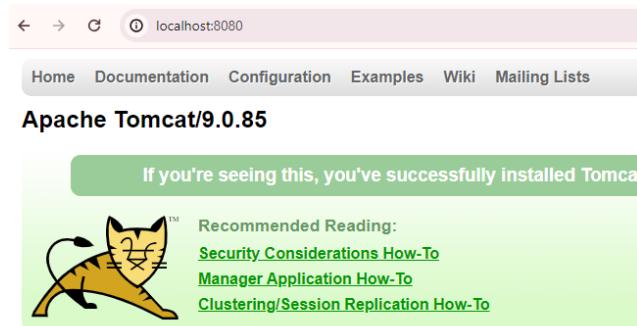
Arrancamos el servidor de Tomcat haciendo doble clic en el apartado de “Services” e introduciendo el usuario y contraseña



Aparecerán las aplicaciones que trae Tomcat por defecto:



Comprobamos la configuración del servidor desde el navegador escribiendo “localhost:8080”



Editamos el archivo “tomcat-users.xml”, como hicimos en la instalación en Ubuntu, para dar permisos a nuestro usuario

Viewing file: DATOS (D:) \ _Software \ apache-tomcat-9.0.85 \ conf \ tomcat-users.xml

Nombre	Fecha de modificación	Tipo	Tamaño
Catalina	23/01/2024 8:57	Carpeta de archivos	
catalina.policy	23/01/2024 8:47	Archivo POLICY	13 KB
catalina.properties	23/01/2024 8:47	Archivo PROPERTY	8 KB
context.xml	23/01/2024 8:47	Microsoft Edge H...	2 KB
jaspic-providers.xml	23/01/2024 8:47	Microsoft Edge H...	2 KB
jaspic-providers.xsd	23/01/2024 8:47	Archivo XSD	3 KB
logging.properties	23/01/2024 8:47	Archivo PROPERTY	5 KB
server.xml	23/01/2024 8:47	Microsoft Edge H...	8 KB
tomcat-users.xml	23/01/2024 8:56	Microsoft Edge H...	3 KB
tomcat-users.xsd	23/01/2024 8:47	Archivo XSD	3 KB
web.xml	23/01/2024 8:57	Microsoft Edge H...	157 KB

```

<!--
<role rolename="tomcat"/>
<role rolename="role1"/>
<user username="tomcat" password="" roles="tomcat"/>
<user username="both" password="" roles="tomcat,role1"/>
<user username="role1" password="" roles="role1"/>
-->
<user password="paso" roles="manager-script,admin,admin-gui,manager-gui" username="tomcat"/>
</tomcat-users>

```

Comprobamos que ahora si tenemos acceso a “Manager App”

Viewing file: DATOS (D:) \ _Software \ apache-tomcat-9.0.85 \ manager \ webapps \ manager \ WEB-INF \ classes \ org \ apache \ tomcat \ manager \ auth \ tomcat-users.xml

Viewing file: localhost:8080/manager/html

Mensaje:	OK																																				
Gestor de Aplicaciones Web de Tomcat																																					
Listar Aplicaciones Ayuda HTML de Gestor Ayuda de Gestor																																					
Aplicaciones <table border="1"> <thead> <tr> <th>Ruta</th> <th>Versión</th> <th>Nombre a Mostrar</th> <th>Ejecutándose</th> <th>Sesiones</th> <th>Comandos</th> </tr> </thead> <tbody> <tr> <td>/</td> <td>Ninguno especificado</td> <td>Welcome to Tomcat</td> <td>true</td> <td>0</td> <td> Arrancar Parar Recargar Re Expirar sesiones sin trabajar ≥ 30 </td> </tr> <tr> <td>/docs</td> <td>Ninguno especificado</td> <td>Tomcat Documentation</td> <td>true</td> <td>0</td> <td> Arrancar Parar Recargar Re Expirar sesiones sin trabajar ≥ 30 </td> </tr> <tr> <td>/examples</td> <td>Ninguno especificado</td> <td>Servlet and JSP Examples</td> <td>true</td> <td>0</td> <td> Arrancar Parar Recargar Re Expirar sesiones sin trabajar ≥ 30 </td> </tr> <tr> <td>/host-manager</td> <td>Ninguno especificado</td> <td>Tomcat Host Manager Application</td> <td>true</td> <td>0</td> <td> Arrancar Parar Recargar Re Expirar sesiones sin trabajar ≥ 30 </td> </tr> <tr> <td>/manager</td> <td>Ninguno especificado</td> <td>Tomcat Manager Application</td> <td>true</td> <td>1</td> <td> Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 </td> </tr> </tbody> </table>		Ruta	Versión	Nombre a Mostrar	Ejecutándose	Sesiones	Comandos	/	Ninguno especificado	Welcome to Tomcat	true	0	Arrancar Parar Recargar Re Expirar sesiones sin trabajar ≥ 30	/docs	Ninguno especificado	Tomcat Documentation	true	0	Arrancar Parar Recargar Re Expirar sesiones sin trabajar ≥ 30	/examples	Ninguno especificado	Servlet and JSP Examples	true	0	Arrancar Parar Recargar Re Expirar sesiones sin trabajar ≥ 30	/host-manager	Ninguno especificado	Tomcat Host Manager Application	true	0	Arrancar Parar Recargar Re Expirar sesiones sin trabajar ≥ 30	/manager	Ninguno especificado	Tomcat Manager Application	true	1	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30
Ruta	Versión	Nombre a Mostrar	Ejecutándose	Sesiones	Comandos																																
/	Ninguno especificado	Welcome to Tomcat	true	0	Arrancar Parar Recargar Re Expirar sesiones sin trabajar ≥ 30																																
/docs	Ninguno especificado	Tomcat Documentation	true	0	Arrancar Parar Recargar Re Expirar sesiones sin trabajar ≥ 30																																
/examples	Ninguno especificado	Servlet and JSP Examples	true	0	Arrancar Parar Recargar Re Expirar sesiones sin trabajar ≥ 30																																
/host-manager	Ninguno especificado	Tomcat Host Manager Application	true	0	Arrancar Parar Recargar Re Expirar sesiones sin trabajar ≥ 30																																
/manager	Ninguno especificado	Tomcat Manager Application	true	1	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30																																

Comprobamos que el puerto 8080 este escuchando desde el cmd con este comando:

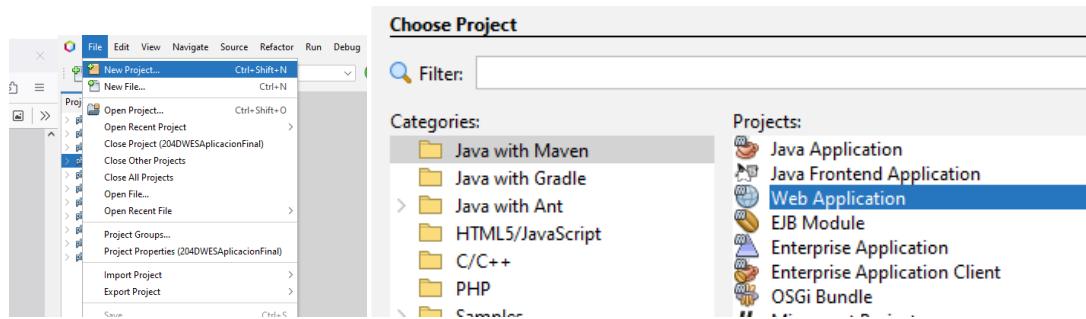
```
C:\Users\daw2>netstat -a -p TCP -n |findstr 8080
  TCP      0.0.0.0:8080          0.0.0.0:0              LISTENING
```

Comprobamos que las apps que aparecen en nuestro navegador se localicen en el directorio de webapps de nuestra maquina local (a mayores aparecería una llamada ROOT que es la aplicación que se abre por defecto)

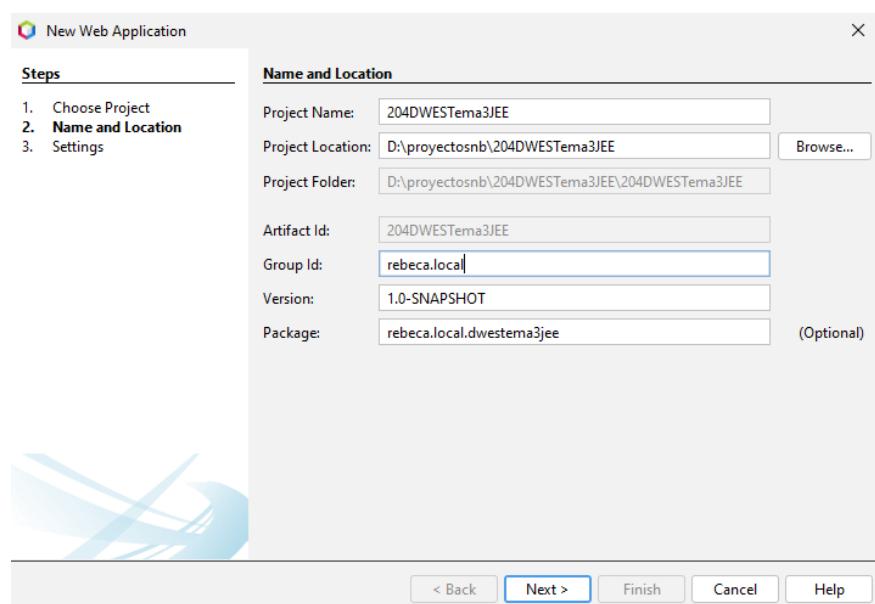
DATOS (D:) > _Software > apache-tomcat-9.0.85 > webapps			
Nombre	Fecha de modificación	Tipo	Tan
docs	23/01/2024 8:47	Carpeta de archivos	
examples	23/01/2024 8:47	Carpeta de archivos	
host-manager	23/01/2024 8:47	Carpeta de archivos	
manager	23/01/2024 8:47	Carpeta de archivos	
ROOT	23/01/2024 8:47	Carpeta de archivos	

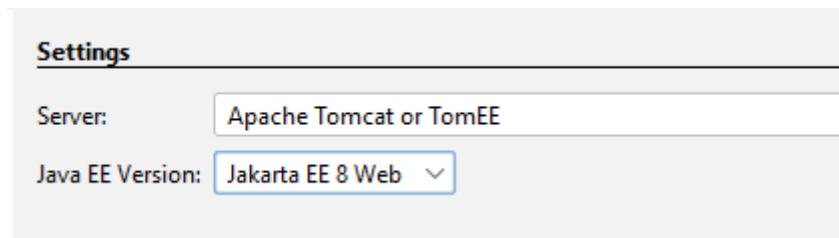
9.3 Creación y despliegue de un proyecto en Windows

Para crear un proyecto de Java en NetBeans nos vamos a “File>New Project” y seleccionamos “Java with Maven>Web Application”

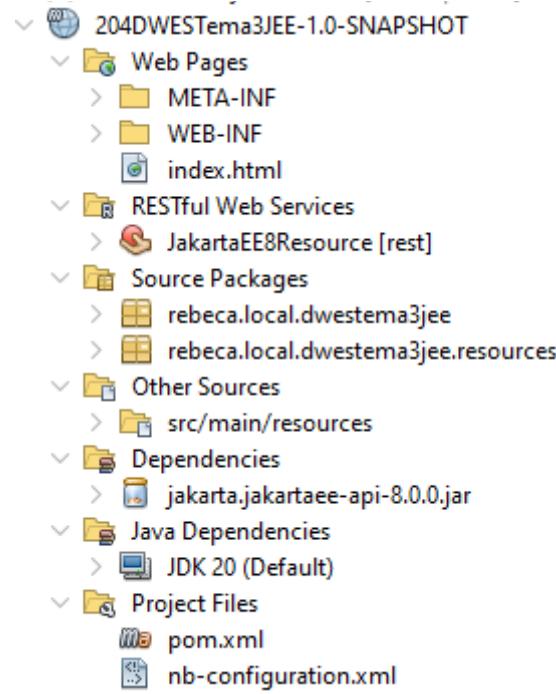


Ahora insertamos los datos como el nombre que le queremos dar al proyecto y si ID de grupo como se muestra en la captura:





Estructura de ficheros de la aplicación generada por defecto:



A la hora de ejecutar el “index” para comprobar el correcto funcionamiento del Hola Mundo, aparece un error que resolvemos cambiando el fichero “pom.xml” de nuestro proyecto. Este error de ejecución aparece porque el proyecto intenta ejecutarse con una versión de Maven que no es la correcta. Lo cambiaremos de la siguiente manera:

```

40      </configuration>
41    </plugin>
42    <plugin>
43      <groupId>org.apache.maven.plugin</groupId>
44      <artifactId>maven-war-plugin</artifactId>
45      <version>2.3</version>
46      <configuration>
47        <failOnMissingWebXml>false</failOnMissingWebXml>
48      </configuration>
49    </plugin>

```

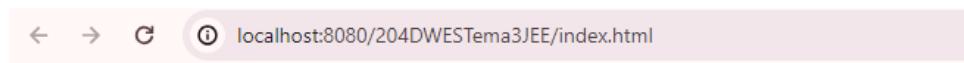


```

40      </configuration>
41    </plugin>
42    <plugin>
43      <groupId>org.apache.maven.plugin</groupId>
44      <artifactId>maven-war-plugin</artifactId>
45      <version>3.3.1</version>
46      <configuration>
47        <failOnMissingWebXml>false</failOnMissingWebXml>
48      </configuration>
49    </plugin>

```

Una vez más lo ejecutamos y comprobamos que funciona correctamente (clic derecho sobre el proyecto “Run File”)



Hola este es el proyecto 204DWESTema3JEE

A continuación, borramos el index.html, lo sustituimos por un index.jsp y hacemos cambios para visualizarlos

204DWESTema3JEE-1.0-SNAPSHOT

- Web Pages
- META-INF
- WEB-INF
- index.jsp** ←
- RESTful Web Services

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>JSP Page</title>
    </head>
    <body>
        <h1>Hola este es el proyecto 204DWESTema3JEE desede JSP</h1>
    </body>
</html>
```

localhost:8080/204DWESTema3JEE/index.jsp

Hola este es el proyecto 204DWESTema3JEE desede JSP

Ahora modificamos el fichero y usamos la clase Date para obtener la fecha y hora

```
<%@page import="java.util.Date"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>JSP Page</title>
    </head>
    <body>
        <h1>Hola este es el proyecto 204DWESTema3JEE desede JSP</h1>
        <% out.println(new Date()); %>
    </body>
</html>
```

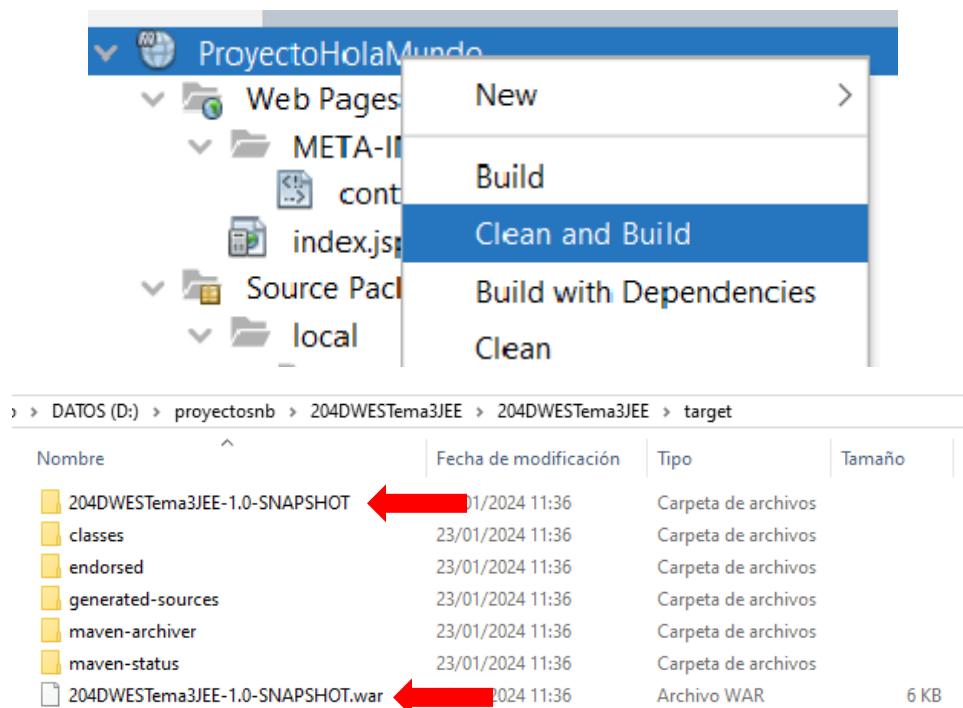
JSP Page

localhost:8080/204DWESTema3JEE/index.jsp

Hola este es el proyecto 204DWESTema3JEE desede JSP

Tue Jan 23 11:33:53 CET 2024

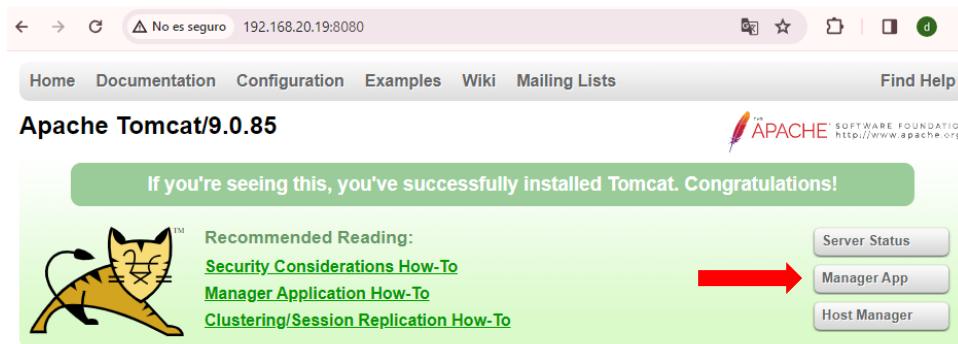
Pasamos al despliegue de la app. Hacemos clic derecho sobre el proyecto “Clean and Build” y se generara automaticamente un archivo .war en la carpeta “target” del proyecto



Subimos esos 2 archivos a nuestro servidor (/public_html/webapps) con el filezilla

Nombre de archivo	Tamaño d...	Tipo de arc...	Última r
..		Carpeta de...	23/01/2024
204DWESTema3JEE		Carpeta de...	23/01/2024
204DWESTema3JEE-1.0-SNAPSHOT		Carpeta de...	23/01/2024
206DWESTema3JEE-1.0-SNAPSHOT		Carpeta de...	23/01/2024
214DWESTema3JEE-1		Carpeta de...	23/01/2024
docs		Carpeta de...	16/01/2024
examples		Carpeta de...	16/01/2024
host-manager		Carpeta de...	16/01/2024
manager		Carpeta de...	16/01/2024
ROOT		Carpeta de...	16/01/2024
202DWESTema3JEE.war	5.180	Archivo W...	23/01/2024
204DWESTema3JEE-1.0-SNAPSHOT.war	5.173	Archivo W...	23/01/2024
206DWESTema3JEE-1.0-SNAPSHOT.war	5.121	Archivo W...	23/01/2024
214DWESTema3JEE-1.war	4.993	Archivo W...	23/01/2024

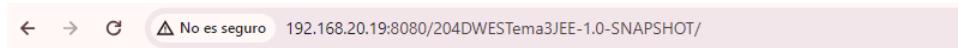
Comprobamos el funcionamiento escribiendo en el navegador la ip de nuestro servidor seguido del puerto 8080



Ahora comprobamos que en las aplicaciones aparece una nueva (que es la que hemos desplegado)

Aplicaciones						
Ruta	Versión	Nombre a Mostrar	Ejecutándose	Sesiones	Comandos	
/	Ninguno especificado	Welcome to Tomcat	true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos	
/202DWESTema3JEE	Ninguno especificado		true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos	
/204DWESTema3JEE-1.0-SNAPSHOT	Ninguno especificado		true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos	
/206DWESTema3JEE-1.0-SNAPSHOT	Ninguno especificado		true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos	
/214DWESTema3JEE-1	Ninguno especificado		true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos	

Si hacemos clic sobre ella, nos redirecciona al index.jsp de nuestro proyecto



Hola este es el proyecto 204DWESTema3JEE desede JSP

Tue Jan 23 11:45:33 CET 2024

9.4 Actualizar cuenta de usuario SFTP

Cambiamos el usuario operadorweb para que pertenezca a al grupo tomcat "sudo usermod -aG tomcat operadorweb" y nos logeamos con operador con "su operadorweb"

```
miadmin@rsp-used:~$ sudo usermod -aG tomcat operadorweb
[sudo] password for miadmin:
miadmin@rsp-used:~$ su operadorweb
Password:
```

Una vez aquí, nos movemos al html "cd /var/www/html", creamos una carpeta llamada webapps "mkdir webapps" y volvemos a conectarnos con miadmin "su miadmin"

```
operadorweb@rsp-used:/home/miadmin$ cd /var/www/html
operadorweb@rsp-used:~/html$ mkdir webapps
mkdir: command not found
operadorweb@rsp-used:~/html$ mkdir webapps
operadorweb@rsp-used:~/html$ su miadmin
Password:
miadmin@rsp-used:/var/www/html$ ls
204DAKProyectoDAK  204DWESElogoff      204DWESElogoffTema4  info.php   webroot
204DIWProyectoDIW  204DWESElogoffTema5  204DWESElogoffTema5  log
204DWEDElogoffTema5 204DWESElogoffTema5  204DWESElogoffTema5  nbproject
204DWESElogoffTema5 204DWESElogoffTema5  204DWESElogoffTema5  webapps
miadmin@rsp-used:~/var/www/html$
```

Montamos de forma temporal (hasta que se apague la maquina) al usuario "sudo mount --bind /opt/tomcat/webapps /var/www/html/webapps"

```
miadmin@rsp-used:/var/www/html$ sudo mount --bind /opt/tomcat/webapps /var/www/html/webapps
miadmin@rsp-used:/var/www/html$ _
```

Mostramos el contenido de webapps “sudo ls -l /var/www/html/webapps”

```
miadmin@rsp-used:/var/www/html$ sudo ls -l /var/www/html/webapps
total 20
drwxr-x--- 16 tomcat tomcat 4096 ene 19 08:56 docs
drwxr-x--- 7 tomcat tomcat 4096 ene 19 08:56 examples
drwxr-x--- 6 tomcat tomcat 4096 ene 19 08:56 host-manager
drwxr-x--- 6 tomcat tomcat 4096 ene 19 08:56 manager
drwxr-x--- 3 tomcat tomcat 4096 ene 19 08:56 ROOT
miadmin@rsp-used:/var/www/html$
```

Modificamos el fichero fstab “sudo nano /etc/fstab” y añadimos en la última línea lo que se muestra en la captura (esto hace que el montaje de esas carpetas sea permanente y se conserve aun habiendo apagado la maquina)

```
GNU nano 6.2                               /etc/fstab *
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options>      <dump> <pass>
# / was on /dev/sda2 during curtin installation
/dev/disk/by-uuid/e4245c52-3b5e-4368-9338-16fbea1d126b / ext4 defaults 0 1
# /var was on /dev/sda3 during curtin installation
/dev/disk/by-uuid/b2d03afa-12f5-4d97-90ef-accd26805b82 /var ext4 defaults 0 1
/swapp.img      none    swap   sw     0     0
/opt/tomcat/webapps /var/www/html/webapps none defaults,bind 0 0 ←
```

9.5 Utilizar tu propio dominio con Tomcat

Una vez más modificamos el fichero de configuracion server.xml “sudo nano /opt/tomcat/conf/server.xml” y añadimos justo debajo de lo que hay lo siguiente:

```
Note: The pattern used is equivalent to using pattern="common" -->
<Valve className="org.apache.catalina.valves.AccessLogValve" directory="logs"
       prefix="localhost_access_log" suffix=".txt"
       pattern="%h %l %u %t &"%r&" %s %b" />

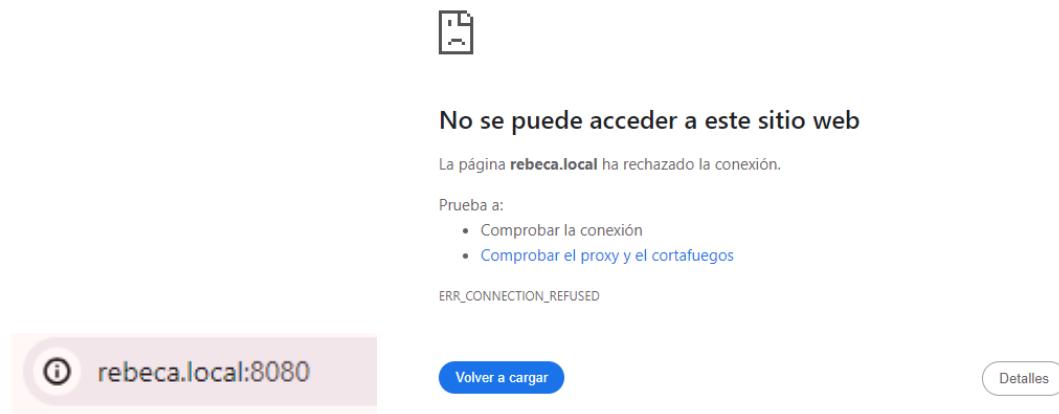
</Host>
<Host name="rebeca.local" appBase="miapp"
      unpackWARs="true" autoDeploy="true">

  <Valve className="org.apache.catalina.valves.AccessLogValve" directory="logs"
         prefix="miapp_access_log" suffix=".txt"
         pattern="%h %l %u %t &"%r&" %s %b" />
</Host>
</Engine>
</Service>
</Container>
```

Creamos una carpeta llamada miapp con “sudo mkdir /opt/tomcat/miapp” y cambiamos su propietario para que sea tomcat tomcat con el comando “sudo chown tomcat:tomcat /opt/tomcat/miapp”. Reiniciamos Tomcat

```
miadmin@rsp-used:~$ sudo mkdir /opt/tomcat/miapp
miadmin@rsp-used:~$ sudo chown tomcat:tomcat /opt/tomcat
miadmin@rsp-used:~$ sudo service tomcat restart
. . . ^
```

Comprobamos en el navegador



Ahora tendremos que montar el directorio que contiene las aplicaciones /opt/tomcat/miapp en el directorio webapps "sudo mount -bind /opt/tomcat/miapp /var/www/html/miapp". Nos tenemos que asegurar de que miapp existe en la carpeta html

```
miadmin@rsp-used:/var/www/html$ sudo mkdir miapp
miadmin@rsp-used:/var/www/html$ sudo mount --bind /opt/tomcat/miapp /var/www/html/miapp
miadmin@rsp-used:/var/www/html$
```

Si queremos montarlo de manera segura (o permanente) añadimos esta línea con "sudo nano /etc/fstab":

```
# / was on /dev/sda2 during curtin installation
/dev/disk/by-uuid/e4245c52-3b5e-4368-9338-16fbea1d126b / ext4 defaults 0 1
# /var was on /dev/sda3 during curtin installation
/dev/disk/by-uuid/b2d03afa-12f5-4d97-90ef-accd26805b82 /var ext4 defaults 0 1
/swap.img none swap sw 0 0
/opt/tomcat/webapps /var/www/html/webapps none defaults,bind 0 0
/opt/tomcat/miapp /var/www/html/miapp none defaults,bind 0 0
```

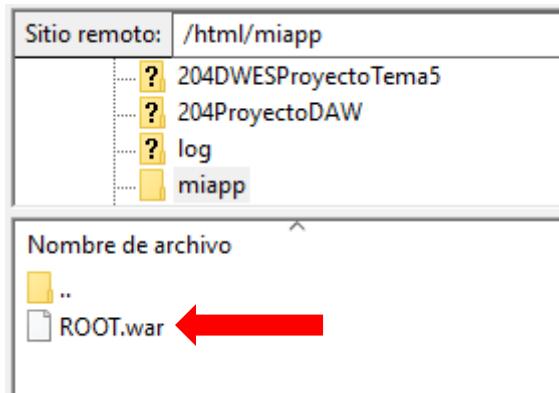
Cambiamos los permisos de miapp a 775

```
miadmin@rsp-used:/var/www/html$ sudo chmod 775 miapp/
```

Cambiamos tambien el propietario de la carpeta para que sea de operadorweb "sudo chown operadorweb:www-data /var/www/html/miapp"

```
miadmin@rsp-used:~$ sudo chown operadorweb:www-data /var/www/html/miapp
miadmin@rsp-used:~$ ls -l /var/www/html
total 72
drwxr-sr-x 10 operadorweb www-data 4096 ene 24 09:54 204DAWProyectoDAW
drwxr-sr-x 8 operadorweb www-data 4096 ene 24 09:54 204DIWProyectoDIW
drwxr-sr-x 8 operadorweb www-data 4096 ene 24 09:54 204DWECProyectoDWEC
drwxrwsr-x 15 operadorweb www-data 4096 ene 24 09:51 204DWESAplicacionFinal
drwxrwsr-x 15 operadorweb www-data 4096 ene 24 09:52 204DWESLoginLogoff
drwxrwsr-x 12 operadorweb www-data 4096 ene 25 10:10 204DWESLoginLogoffTema5
drwxr-sr-x 7 operadorweb www-data 4096 ene 24 09:59 204DWESProyectoDWES
drwxr-sr-x 9 operadorweb www-data 4096 oct 24 10:41 204DWESProyectoTema3
drwxrwsr-x 10 operadorweb www-data 4096 oct 31 10:13 204DWESProyectoTema4
drwxrwsr-x 13 operadorweb www-data 4096 ene 24 09:54 204DWESProyectoTema5
drwxrwsr-x 7 operadorweb www-data 4096 ene 24 09:54 204ProyectoDAW
-rw-r--r-- 1 operadorweb www-data 1619 oct 11 09:28 index.html
-rw-r--r-- 1 root www-data 21 oct 3 08:48 info.php
drwxr-sr-x 2 root www-data 4096 oct 2 11:29 log
drwxrwsr-x 2 operadorweb www-data 4096 ene 30 08:59 miapp
drwxr-sr-x 3 operadorweb www-data 4096 oct 3 09:36 nbproject
```

Con FileZilla nos conectamos con operadorweb y subimos una aplicación a esa carpeta con el nombre ROOT (para que sea la app por defecto) y con la extensión war



9.6 Tomcat seguro

Nos movemos al directorio `/opt/tomcat/conf` (si no nos deja porque no tenemos permisos, escribimos “`sudo su`” y volvemos hacer el `cd` como root).

Una vez ahí escribimos el comando “`keytool -genkey -keyalg RSA -alias server -keystore miapp.jks -validity 365 -keysize 2048`” y seguimos las instrucciones que aparecen en la captura:

```
miadmin@rsp-used:~$ cd /opt/tomcat/conf
-bash: cd: /opt/tomcat/conf: Permission denied
miadmin@rsp-used:~$ sudo su
[sudo] password for miadmin:
root@rsp-used:/home/miadmin# cd /opt/tomcat/conf
root@rsp-used:/opt/tomcat/conf# keytool -genkey -keyalg RSA -alias server -keystore miapp.jks -validity 365 -keysize 204
8
Introduzca la contraseña del almacén de claves:
Volver a escribir la contraseña nueva:
¿Cuáles son su nombre y su apellido?
[Unknown]: rebecca.local
¿Cuál es el nombre de su unidad de organización?
[Unknown]: tomcat
¿Cuál es el nombre de su organización?
[Unknown]: rebecca.local
¿Cuál es el nombre de su ciudad o localidad?
[Unknown]: benavente
¿Cuál es el nombre de su estado o provincia?
[Unknown]: zamora
¿Cuál es el código de país de dos letras de la unidad?
[Unknown]: es
¿Es correcto CN=rebecca.local, OU=tomcat, O=rebecca.local, L=benavente, ST=zamora, C=es?
[no]: si
```

Para confirmar los cambios escribimos “si” (está en español, no hay que poner “yes”) y hacemos “`ls -l`”

```
root@rsp-used:/opt/tomcat/conf# ls -l
total 240
drwxr-x--- 3 tomcat tomcat 4096 ene 19 09:09 Catalina
-rw-r---- 1 tomcat tomcat 12953 ene 5 09:28 catalina.policy
-rw-r---- 1 tomcat tomcat 7630 ene 5 09:28 catalina.properties
-rw-r---- 1 tomcat tomcat 1400 ene 5 09:28 context.xml
-rw-r---- 1 tomcat tomcat 1149 ene 5 09:28 jaspic-providers.xml
-rw-r---- 1 tomcat tomcat 2313 ene 5 09:28 jaspic-providers.xsd
-rw-r---- 1 tomcat tomcat 4144 ene 5 09:28 logging.properties
-rw-r--r-- 1 root root 2744 ene 30 09:12 miapp.jks
-rw-r---- 1 tomcat tomcat 8179 ene 30 08:59 server.xml
-rw-r---- 1 tomcat tomcat 2856 ene 19 09:19 tomcat-users.xml
-rw-r---- 1 tomcat tomcat 2558 ene 5 09:28 tomcat-users.xsd
-rw-r---- 1 tomcat tomcat 172951 ene 5 09:28 web.xml
```

El archivo miapp.jks pertenece a root root y tenemos que cambiarlo a tomcat tomcat con “sudo chown tomcat:tomcat miapp.jks”

```
root@rsp-used:/opt/tomcat/conf# sudo chown tomcat:tomcat miapp.jks
root@rsp-used:/opt/tomcat/conf# ls -l
total 240
drwxr-x--- 3 tomcat tomcat 4096 ene 19 09:09 Catalina
-rw-r----- 1 tomcat tomcat 12953 ene 5 09:28 catalina.policy
-rw-r----- 1 tomcat tomcat 7630 ene 5 09:28 catalina.properties
-rw-r----- 1 tomcat tomcat 1400 ene 5 09:28 context.xml
-rw-r----- 1 tomcat tomcat 1149 ene 5 09:28 jaspic-providers.xml
-rw-r----- 1 tomcat tomcat 2313 ene 5 09:28 jaspic-providers.xsd
-rw-r----- 1 tomcat tomcat 4144 ene 5 09:28 logging.properties
-rw-r--r-- 1 tomcat tomcat 2744 ene 30 09:12 miapp.jks
-rw-r----- 1 tomcat tomcat 8179 ene 30 08:59 server.xml
-rw-r----- 1 tomcat tomcat 2856 ene 19 09:19 tomcat-users.xml
-rw-r----- 1 tomcat tomcat 2558 ene 5 09:28 tomcat-users.xsd
-rw-r----- 1 tomcat tomcat 172951 ene 5 09:28 web.xml
```

Hacemos “sudo nano server.xml” y descomentamos el connector que tiene el protocolo NIO

```
-->
<!--
<Connector port="8443" protocol="org.apache.coyote.http11.Http11NioProtocol"
           maxThreads="150" SSLEnabled="true"
           maxParameterCount="1000"
           >
    <SSLHostConfig>
        <Certificate certificateKeystoreFile="conf/localhost-rsa.jks"
                      type="RSA" />
    </SSLHostConfig>
</Connector>
-->
<!-- Define an SSL/TLS HTTP/1.1 Connector on port 8443 with HTTP/2
```

Añadimos certificateKeystorePassword y certificateKeyAlias

```
<Connector port="8443" protocol="org.apache.coyote.http11.Http11NioProtocol"
           maxThreads="150" SSLEnabled="true"
           maxParameterCount="1000"
           >
    <SSLHostConfig>
        <Certificate certificateKeystoreFile="conf/miapp.jks" certificateKeystorePassword="paso1234"
                      certificateKeyAlias="server"
                      type="RSA" />
    </SSLHostConfig>
</Connector>
```

Reiniciamos el servicio de tomcat

```
root@rsp-used:/opt/tomcat/conf# sudo service tomcat restart
root@rsp-used:/opt/tomcat/conf#
```

Abrimos el puerto 8443 “sudo ufw allow 8443” y visualizamos los puertos abiertos

```
root@rsp-used:/opt/tomcat/conf# sudo ufw allow 8443
Rule added
Rule added (v6)
root@rsp-used:/opt/tomcat/conf# sudo ufw status
Status: active

To                         Action      From
--                         --          --
22                         ALLOW       Anywhere
Apache                      ALLOW       Anywhere
9003                        ALLOW      Anywhere
3306                        ALLOW      Anywhere
443                         ALLOW      Anywhere
53                          ALLOW      Anywhere
8080                        ALLOW      Anywhere
8443                        ALLOW      Anywhere
22 (v6)                     ALLOW     Anywhere (v6)
Apache (v6)                  ALLOW     Anywhere (v6)
9003 (v6)                   ALLOW     Anywhere (v6)
3306 (v6)                   ALLOW     Anywhere (v6)
443 (v6)                    ALLOW     Anywhere (v6)
53 (v6)                     ALLOW     Anywhere (v6)
8080 (v6)                   ALLOW     Anywhere (v6)
8443 (v6)                   ALLOW     Anywhere (v6)
```

Comprobamos en el navegador escribiendo “<http://192.168.3.204:8443/>”

10. LDAP

10.1 Configuracion del servidor LDAP

A partir de la maquina limpia que guardamos al principio de curso, crearemos y configuraremos un servidor LDAP que tendrá el DNS. Una vez clonada, iniciamos la maquina y cambiamos la IP de la maquina nueva “sudo nano /etc/netplan/00-installer-config.yaml” y aplicamos los cambios

```
GNU nano 6.2                               00-installer-config.yaml
# This is the network config written by 'subiquity'
network:
  ethernets:
    enp0s3:
      addresses:
        - 192.168.3.104/24
      nameservers:
        addresses:
          - 8.8.8.8
        search: []
      routes:
        - to: default
          via: 192.168.3.1
  version: 2
```

```
miadmin@rsp-uslimpia:/etc/netplan$ sudo netplan apply
```

Cambiamos el nombre de la maquina con este comando

```
miadmin@rsp-uslimpia:/etc/netplan$ sudo hostnamectl set-hostname rsp-ldap_
```

Modificamos el fichero host para cambiar el nombre

```
miadmin@rsp-uslimpia:/etc/netplan$ sudo nano /etc/hosts
```

```
GNU nano 6.2                               /etc/hosts *
127.0.0.1 localhost
127.0.1.1 rsp-ldap_

# 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
```

Reiniciamos la maquina con “sudo reboot” y comprobamos que hace ping a Google

```
miadmin@rsp-used:~$ ping google.com
PING google.com (142.250.184.14) 56(84) bytes of data.
64 bytes from mad41s10-in-f14.1e100.net (142.250.184.14): icmp_seq=1 ttl=114 time=10.9 ms
64 bytes from mad41s10-in-f14.1e100.net (142.250.184.14): icmp_seq=2 ttl=114 time=13.8 ms
64 bytes from mad41s10-in-f14.1e100.net (142.250.184.14): icmp_seq=3 ttl=114 time=11.0 ms
64 bytes from mad41s10-in-f14.1e100.net (142.250.184.14): icmp_seq=4 ttl=114 time=11.1 ms
^C
--- google.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 4282ms
rtt min/avg/max/mdev = 10.892/11.687/13.834/1.240 ms
miadmin@rsp-used:~$
```

Actualizamos los paquetes de Linux “sudo apt update” e instalamos bind9 con “sudo apt install bind9”

```
miadmin@rsp-used:~$ sudo apt install bind9
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
bind9 ya está en su versión más reciente (1:9.18.18-0ubuntu0.22.04.1).
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 28 no actualizados.
miadmin@rsp-used:~$
```

Volvemos a modificar el fichero de configuración de red y cambiamos el DNS a la IP de la maquina y el dominio a rebecca.local

```
miadmin@rsp-used:~$ sudo nano /etc/netplan/00-installer-config.yaml
```

```
# This is the network config written by 'systemctl netplan apply'
network:
  ethernets:
    enp0s3:
      addresses:
        - 192.168.3.104/24
      nameservers:
        addresses:
          - 192.168.3.104
        search: [rebecca.local]
      routes:
        - to: default
          via: 192.168.3.1
  version: 2
```

```
miadmin@rsp-ldap:~$ sudo netplan apply
```

Con las 2 maquinas encendidas hacemos el comando “scp miadmin@192.168.3.204:ruta1/ruta2” que nos permite copiar archivos de una máquina virtual a otra para copiar los 3 archivos referentes a la zona directa y la zona inversa del DNS de la máquina virtual USED

```
miadmin@rsp-ldap:~$ scp miadmin@192.168.3.204:/etc/bind/named.conf.local /home/miadmin
The authenticity of host '192.168.3.204 (192.168.3.204)' can't be established.
ED25519 key fingerprint is SHA256:uQvY2jj3SEd8GsJD9Sstnc21rWr2B1fxo5A2bm+AFpk.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.3.204' (ED25519) to the list of known hosts.
miadmin@192.168.3.204's password:
named.conf.local                                              100%   333    181.9KB/s  00:00
miadmin@rsp-ldap:~$
```

```
miadmin@rsp-ldap:~$ scp miadmin@192.168.3.204:/etc/bind/db.rebecca.local /home/miadmin
miadmin@192.168.3.204's password:
db.rebecca.local                                              100%   502    427.3KB/s  00:00
```

```
miadmin@rsp-ldap:~$ scp miadmin@192.168.3.204:/etc/bind/db.3.168.192.in-addr.arpa /home/miadmin
miadmin@192.168.3.204's password:
db.3.168.192.in-addr.arpa                                         100%   349    421.7KB/s  00:00
```

Modificamos el db.rebecca.local con la configuracion de la maquina LDAP:

```
GNU nano 6.2                                     db.rebeca.local

; BIND data for rebeca.local

$TTL    604800
@       IN      SOA    [redacted] rebecasanperi.educa.jcyl.es. (
                      1      ; Serial
                      604800    ; Refresh
                      86400     ; Retry
                     2419200   ; Expire
                      3600 )    ; Negative Cache TTL
;
@       IN      NS     [redacted]
; Registros Host
@       IN      A      192.168.3.204
rsp-used  IN      A      192.168.3.204
[redacted] IN      A      192.168.3.104
;Registro Alias
daw201   IN      CNAME  rsp-used.rebeca.local.
daw202   IN      CNAME  rsp-used.rebeca.local.
daw204   IN      CNAME  rsp-used.rebeca.local.
login    IN      CNAME  rsp-used.rebeca.local.
```

Lo movemos de carpeta y checqueamos la zona con el comando ["sudo named-checkzone rebeca.local /etc/bind/db.rebeca.local"](#)

```
miadmin@rsp-ldap:~$ sudo mv named.conf.local /etc/bind
miadmin@rsp-ldap:~$ sudo mv db* /etc/bind
miadmin@rsp-ldap:~$ sudo named-checkzone rebeca.local /etc/bind/db.rebeca.local
zone rebeca.local/IN: loaded serial 1
OK
```

Ahora cambiamos la zona inversa de la siguiente forma y después de checkear, reiniciamos el servicio:

```
miadmin@rsp-ldap:~$ sudo nano /etc/bind/db.3.168.192.in-addr.arpa

; BIND data for 3.168.192.in-addr.arpa
;
$TTL    604800
@       IN      SOA    [redacted] root.localhost. (
                      1      ; Serial
                      604800    ; Refresh
                      86400     ; Retry
                     2419200   ; Expire
                      3600 )    ; Negative Cache TTL
;
@       IN      NS     [redacted]
;Registros punteros o pointers
204    IN      PTR    rsp-used.rebeca.local.
204    IN      PTR    rebeca.local.
104    IN      PTR    [redacted]
```

```
miadmin@rsp-ldap:~$ sudo named-checkzone 3.168.192.in-addr.arpa /etc/bind/db.3.168.192.in-addr.arpa
zone 3.168.192.in-addr.arpa/IN: loaded serial 1
OK
```

```
miadmin@rsp-ldap:~$ sudo service bind9 restart
miadmin@rsp-ldap:~$
```

Comprobamos que hace ping a los dominios de ldap, de used y de login:

```
miadmin@rsp-ldap:~$ ping rsp-ldap
PING rsp-ldap (127.0.1.1) 56(84) bytes of data.
64 bytes from rsp-ldap (127.0.1.1): icmp_seq=1 ttl=64 time=0.031 ms
64 bytes from rsp-ldap (127.0.1.1): icmp_seq=2 ttl=64 time=0.091 ms
64 bytes from rsp-ldap (127.0.1.1): icmp_seq=3 ttl=64 time=0.061 ms
64 bytes from rsp-ldap (127.0.1.1): icmp_seq=4 ttl=64 time=0.082 ms
^C
--- rsp-ldap ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3019ms
rtt min/avg/max/mdev = 0.031/0.066/0.091/0.023 ms
```

```
miadmin@rsp-ldap:~$ ping rsp-used
PING rsp-used.rebeca.local (192.168.3.204) 56(84) bytes of data.
64 bytes from 192.168.3.204 (192.168.3.204): icmp_seq=1 ttl=64 time=0.808 ms
64 bytes from 192.168.3.204 (192.168.3.204): icmp_seq=2 ttl=64 time=0.969 ms
64 bytes from 192.168.3.204 (192.168.3.204): icmp_seq=3 ttl=64 time=0.927 ms
^C
--- rsp-used.rebeca.local ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 0.808/0.901/0.969/0.068 ms
miadmin@rsp-ldap:~$
```

```
miadmin@rsp-ldap:~$ ping login
PING rsp-used.rebeca.local (192.168.3.204) 56(84) bytes of data.
64 bytes from 192.168.3.204 (192.168.3.204): icmp_seq=1 ttl=64 time=0.401 ms
64 bytes from 192.168.3.204 (192.168.3.204): icmp_seq=2 ttl=64 time=1.08 ms
^C
--- rsp-used.rebeca.local ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 0.401/0.741/1.082/0.340 ms
miadmin@rsp-ldap:~$
```

Para finalizar abrimos el puerto 53 de esta maquina

```
miadmin@rsp-ldap:~$ sudo ufw status
Status: active

To                         Action      From
--                         ----       ---
22                         ALLOW      Anywhere
22 (v6)                    ALLOW      Anywhere (v6)

miadmin@rsp-ldap:~$ sudo ufw allow 53
Rule added
Rule added (v6)
miadmin@rsp-ldap:~$ sudo ufw status
Status: active

To                         Action      From
--                         ----       ---
22                         ALLOW      Anywhere
53                         ALLOW      Anywhere
22 (v6)                    ALLOW      Anywhere (v6)
53 (v6)                    ALLOW      Anywhere (v6)
```

Ahora vamos a la maquina USED y cambiamos la configuracion de red para que use el DNS de la maquina nueva

```
miadmin@rsp-used:~$ sudo nano /etc/netplan/00-installer-config.yaml
```



```
# This is the network config written by netplan
network:
  ethernets:
    enp0s3:
      addresses:
        - 192.168.3.204/24
      nameservers:
        addresses:
          - 192.168.3.104
      search: [rebecca.local]
  routes:
    - to: default
      via: 192.168.3.1
version: 2
```

```
miadmin@rsp-used:~$ sudo netplan apply
** (generate:15998): WARNING **: 10:42:34.659: Permissions for /etc/netplan/00-installer-config.yaml are too open. Netplan configuration should NOT be accessible by others.
WARNING:root:Cannot call Open vSwitch: ovsdb-server.service is not running.

** (process:15996): WARNING **: 10:42:35.376: Permissions for /etc/netplan/00-installer-config.yaml are too open. Netplan configuration should NOT be accessible by others.

** (process:15996): WARNING **: 10:42:35.543: Permissions for /etc/netplan/00-installer-config.yaml are too open. Netplan configuration should NOT be accessible by others.

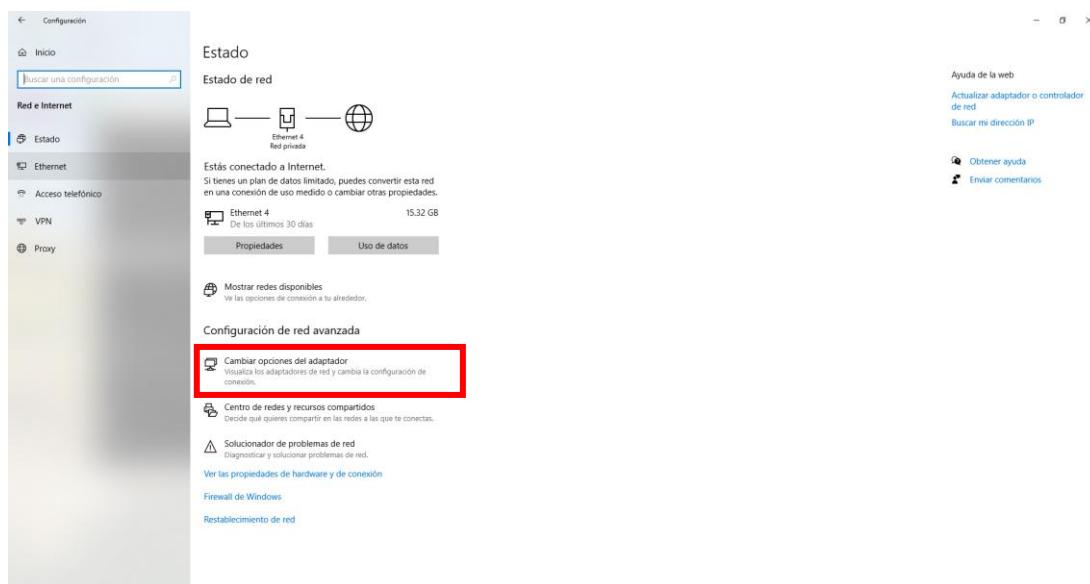
** (process:15996): WARNING **: 10:42:35.543: Permissions for /etc/netplan/00-installer-config.yaml are too open. Netplan configuration should NOT be accessible by others.
```

Desinstalamos bind9 en la maquina USED con los comandos:

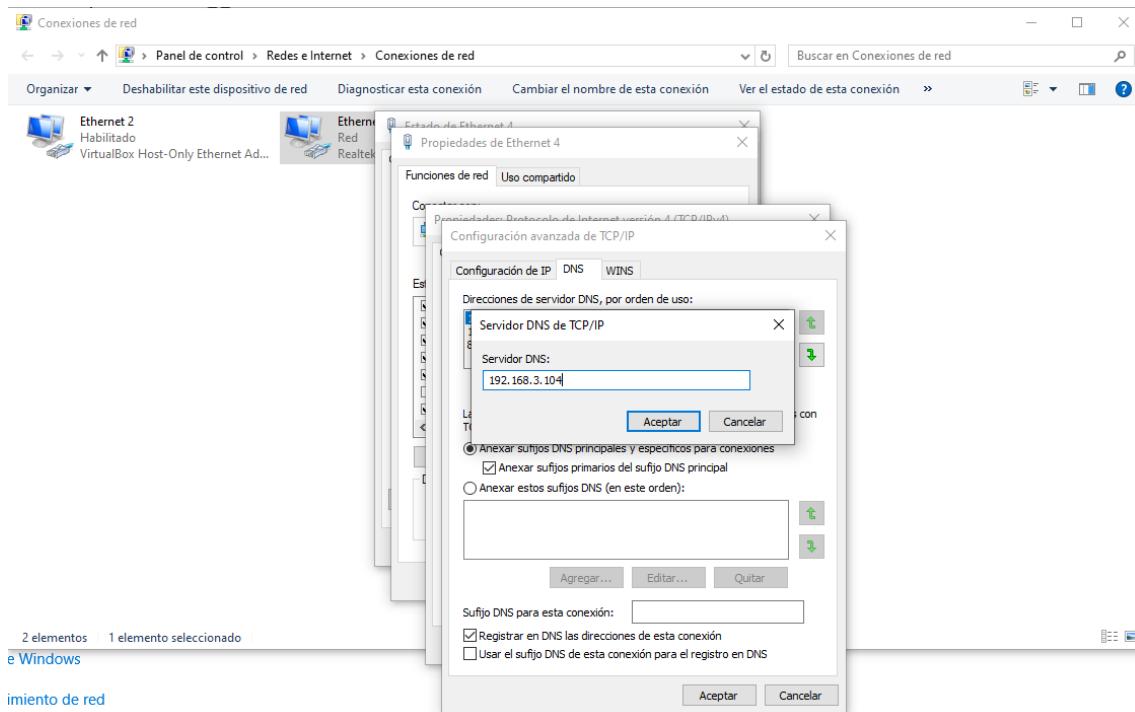
- Sudo apt remove bind9
- Sudo apt purge bind9
- Sudo apt clean bind9
- Sudo apt autoremove
- Sudo apt upgrade

10.2 Configuracion del DNS en Windows 10

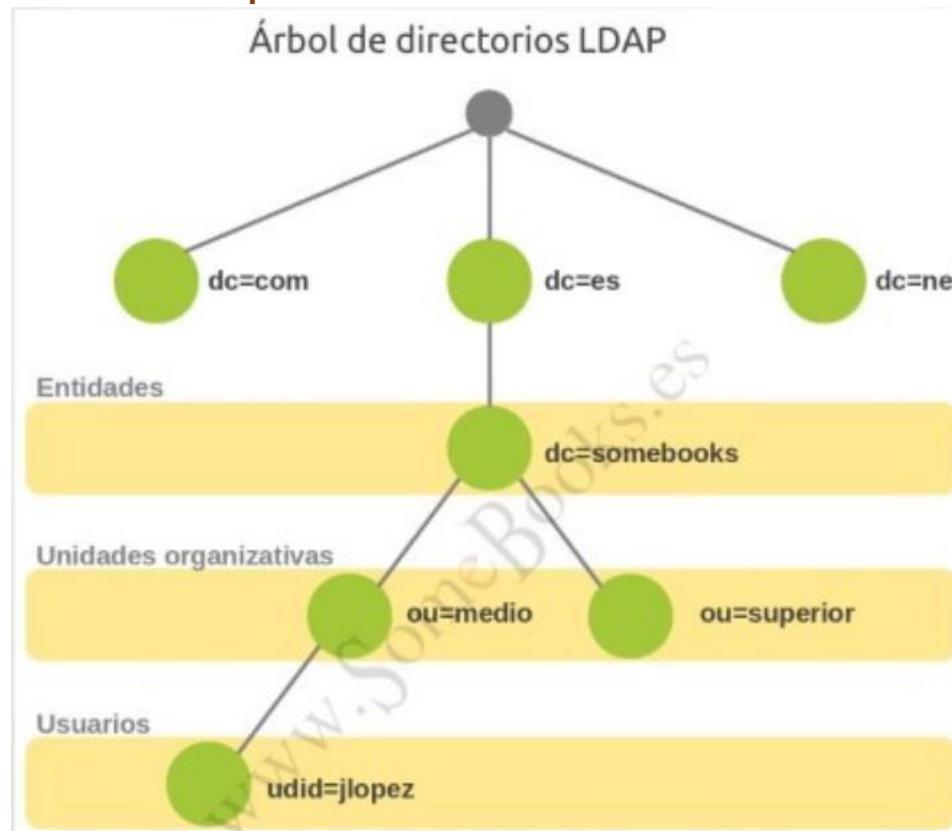
Entramos en la configuracion de red del equipo y seleccionamos “Cambiar opciones del adaptador”



Seleccionamos el Ethernet 4 “propiedades>IPv4>propiedades avanzadas>DNS” y añadimos la IP de nuestra maquina LDAP (192.168.3.104) para que tenga preferencia para usar su DNS cuando la maquina este encendida

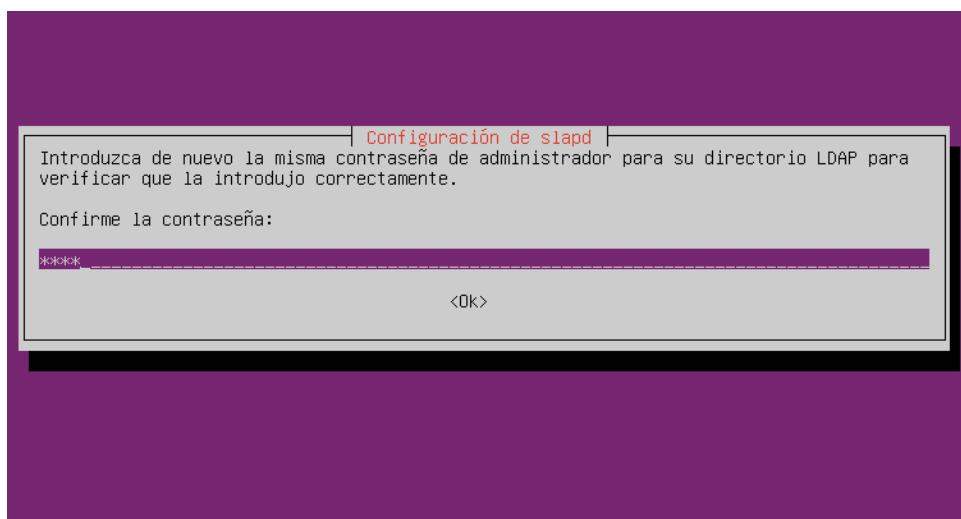
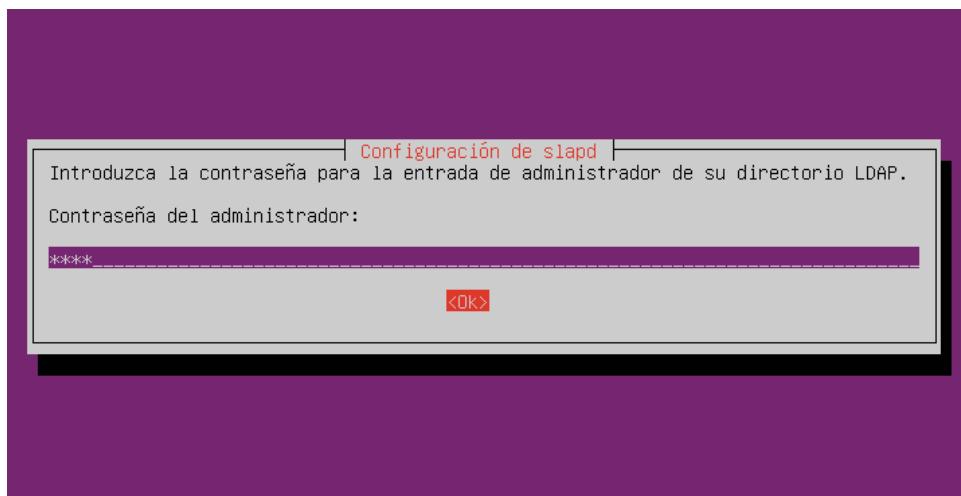


10.3 Instalar OpenLDAP



Antes de instalar Ldap hacemos un update del sistema y posteriormente usamos el comando [“sudo apt install slapd ldap-utils”](#) para instalar el servicio

Se cambiará la pantalla y nos irán pidiendo los siguientes datos:



- Contraseña: paso



Reiniciamos la maquina con el comando “sudo reboot” y después modificamos el host añadiendo la línea que se muestra en la captura

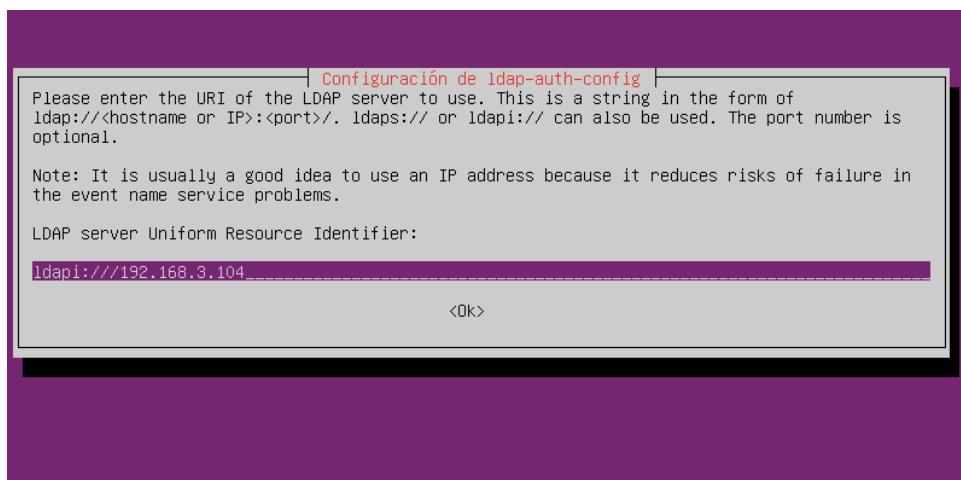
```
miadmin@rsp-ldap:~$ sudo nano /etc/hosts
```

```
GNU nano 6.2                               /etc/hosts
127.0.0.1 localhost
127.0.1.1 rsp-ldap
192.168.3.104 rsp-ldap.rebecca.local rsp-ldap ←

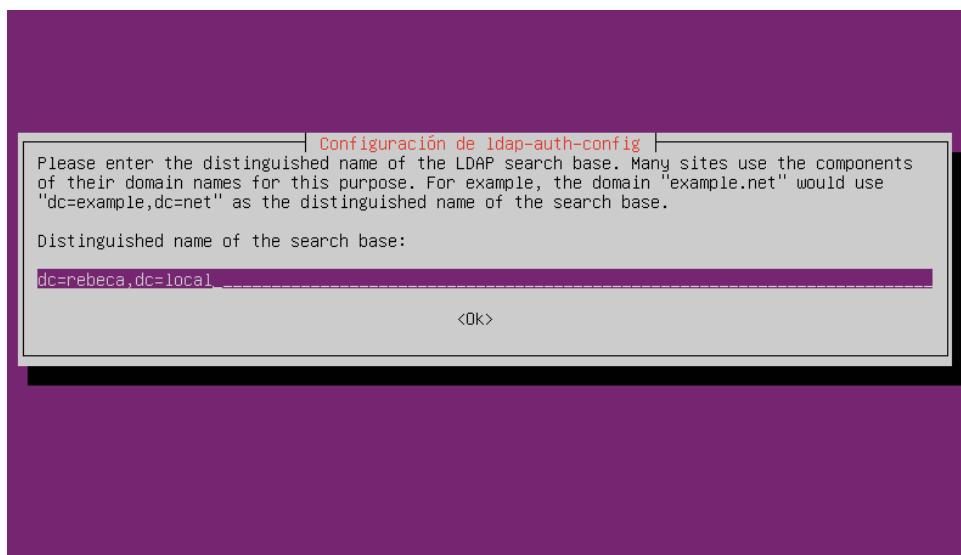
# The following lines are desirable for IPv6 capable hosts
::1      ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

volvemos a reiniciar la maquina e instalamos el paquete de Ldap con “sudo apt install libnss-ldap” y volverá a aparecer la pantalla que iremos rellenando con los datos que pide

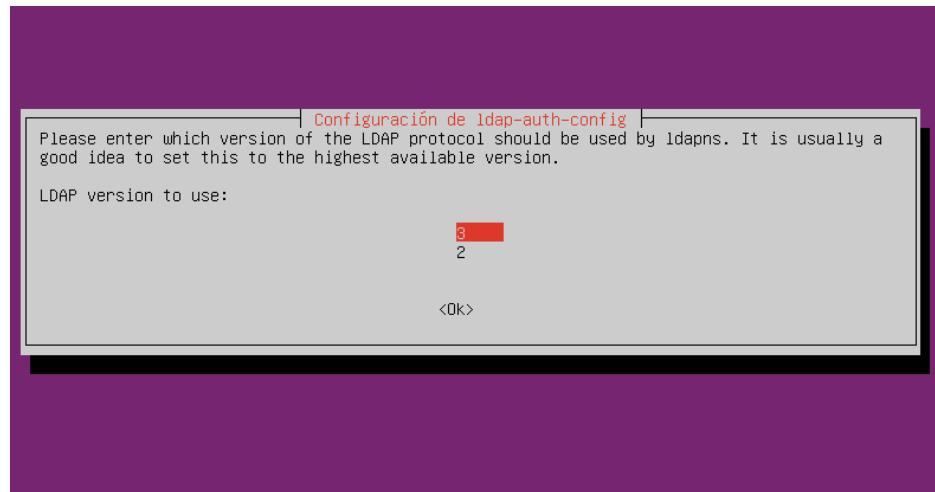
```
miadmin@rsp-ldap:~$ sudo apt install libnss-ldap
```



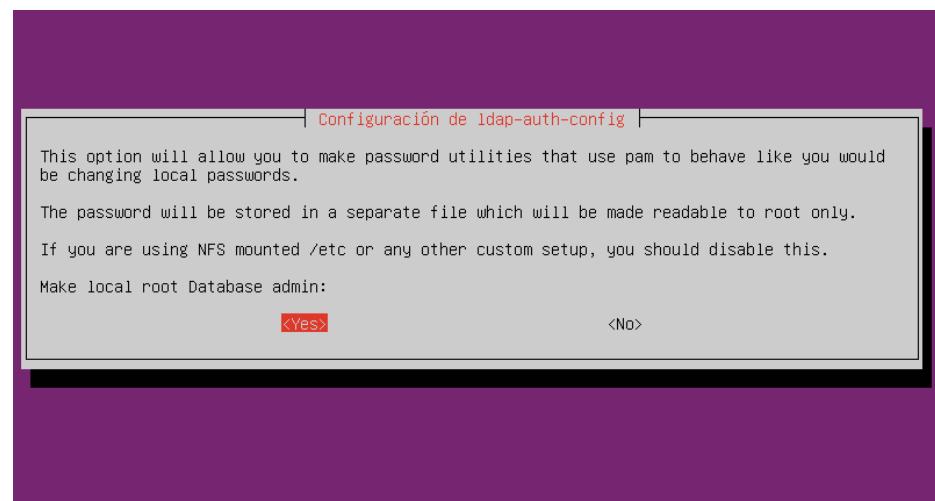
- IP del servidor ldap: 192.168.3.104



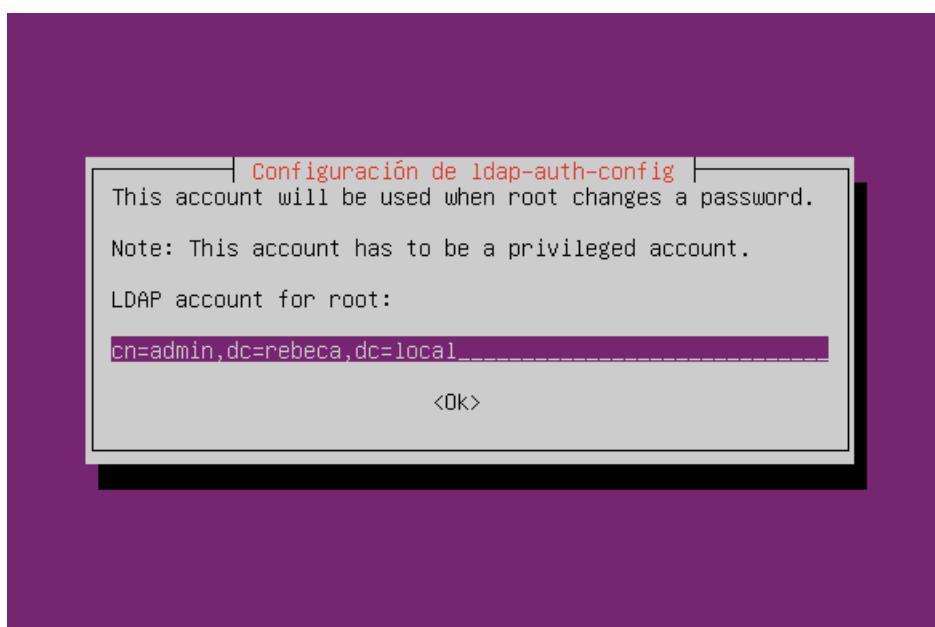
- De acuerdo con nuestro árbol: dc=rebecca,dc=local



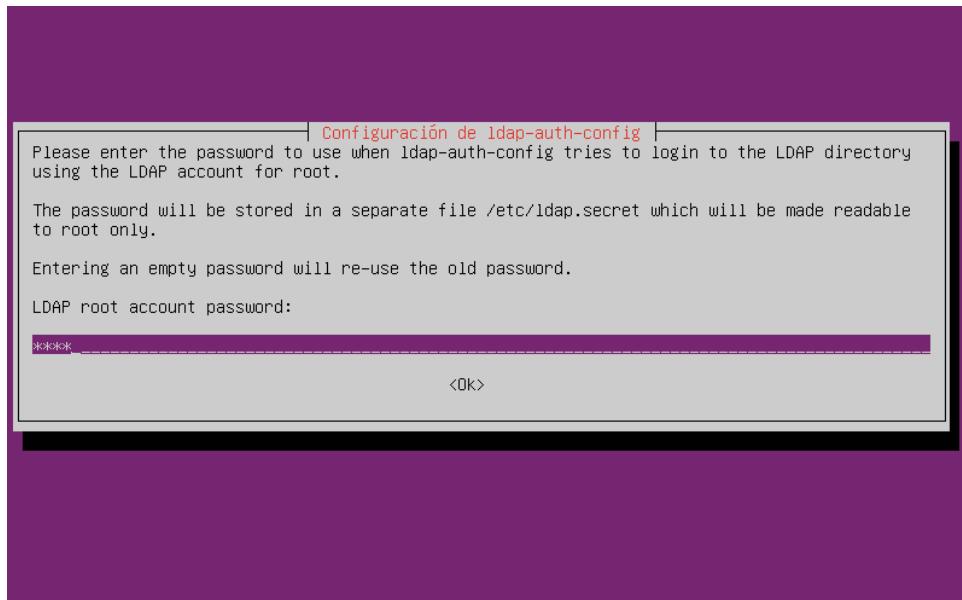
- Version de LDAP: 3



- Crear la base de datos local



- Cuenta para root: cn=admin,dc=rebeca,dc=local



Para acceder al fichero de configuracion usamos “[sudo nano /etc/ldap.conf](#)”. Ahí podemos comprobar que la configuracion al instalarlo es la correcta

```
GNU nano 6.2                               /etc/ldap.conf
# PADL Software
# http://www.padl.com
#
# Your LDAP server. Must be resolvable without using LDAP.
# Multiple hosts may be specified, each separated by a
# space. How long nss_ldap takes to failover depends on
# whether your LDAP client library supports configurable
# network or connect timeouts (see bind_timelimit).
#host 127.0.0.1

# The distinguished name of the search base.
base dc=rebeca,dc=local

# Another way to specify your LDAP server is to provide an
uri ldapi:///192.168.3.104
# Unix Domain Sockets to connect to a local LDAP Server.
#uri ldap://127.0.0.1/
#uri ldaps://127.0.0.1/
#uri ldapi://%2fvar%2frun%2fldapi_sock/
# Note: %2f encodes the '/' used as directory separator

# The LDAP version to use (defaults to 3
# if supported by client library)
ldap_version 3

# The distinguished name to bind to the server with.
# Optional: default is to bind anonymously.
binddn cn=proxyuser,dc=rebeca,dc=local

# The credentials to bind with.
# Optional: default is no credential.
bindpw paso
```

Para listar los objetos que hay en la base de datos usamos el comando “[sudo ldapsearch -xLLL -b dc=rebeca,dc=local](#)”

```
miadmin@rsp-ldap:~$ sudo ldapsearch -xLLL -b dc=rebeca,dc=local
No such object (32)
```

Otra forma de hacer es con el comando “**sudo slapcat**” aquí se mostrarán los objetos usuarios, grupos, unidades organizativas....

```
miadmin@rsp-ldap:~$ sudo slapcat
dn: dc=nodomain
objectClass: top
objectClass: dcObject
objectClass: organization
o: nodomain
dc: nodomain
structuralObjectClass: organization
entryUUID: 80128a52-55ed-103e-9f18-fdd3c50ba871
creatorsName: cn=admin,dc=nodomain
createTimestamp: 202402020804572
entryCSN: 20240202080457.5368992#000000#000#000000
modifiersName: cn=admin,dc=nodomain
modifyTimestamp: 202402020804572
```

10.4 Comprobación de la instalación

Comprobamos que ldap está ejecutándose con “**ps -aux | grep ldap**”

```
miadmin@rsp-ldap:~$ ps -aux | grep ldap
openldap 757 0.0 0.2 1157968 5560 ? Ssl 08:31 0:00 /usr/sbin/slappd -h ldap:///
:/:/ -g openldap -u openldap -F /etc/ldap/slappd.d
miadmin 1061 0.0 0.1 6608 2424 tty1 S+ 08:44 0:00 grep --color=auto ldap
miadmin@rsp-ldap:~$
```

A continuación, listaremos los puertos que están escuchando con “**ss -punta**” y localizamos el de ldap

```
miadmin@rsp-ldap:~$ ss -punta
Netid State Recv-Q Send-Q Local Address:Port Peer Address:Port Process
udp UNCONN 0 0 192.168.3.104:53 0.0.0.0:*
udp UNCONN 0 0 127.0.0.1:53 0.0.0.0:*
udp UNCONN 0 0 127.0.0.53%lo:53 0.0.0.0:*
udp UNCONN 0 0 [:1]:53 [:]*/
udp UNCONN 0 0 [fe80::a00:27ff:fe89:5ba6]%enp0s3:53 [:]*/
tcp LISTEN 0 10 192.168.3.104:53 0.0.0.0:*
tcp LISTEN 0 10 127.0.0.1:53 0.0.0.0:*
tcp LISTEN 0 4096 127.0.0.53%lo:53 0.0.0.0:*
tcp LISTEN 0 128 0.0.0.0:22 0.0.0.0:*
tcp LISTEN 0 5 127.0.0.1:953 0.0.0.0:*
tcp LISTEN 0 2048 0.0.0.0:889 0.0.0.0:*
tcp LISTEN 0 10 [:1]:53 [:]*/
tcp LISTEN 0 10 [fe80::a00:27ff:fe89:5ba6]%enp0s3:53 [:]*/
tcp LISTEN 0 128 [:1]:22 [:]*/
tcp LISTEN 0 5 [:1]:953 [:]*/
tcp LISTEN 0 2048 [:1]:389 [:]*/
miadmin@rsp-ldap:~$
```

Si nos resulta más fácil encontrar el servicio buscando su nombre escribimos “**ss -puta**”

```
miadmin@rsp-ldap:~$ ss -puta
Netid State Recv-Q Send-Q Local Address:Port Peer Address:Port Process
udp UNCONN 0 0 192.168.3.104:domain 0.0.0.0:*
udp UNCONN 0 0 127.0.0.1:domain 0.0.0.0:*
udp UNCONN 0 0 127.0.0.53%lo:domain 0.0.0.0:*
udp UNCONN 0 0 [:1]:domain [:]*/
udp UNCONN 0 0 [fe80::a00:27ff:fe89:5ba6]%enp0s3:domain [:]*/
tcp LISTEN 0 10 192.168.3.104:domain 0.0.0.0:*
tcp LISTEN 0 10 127.0.0.1:domain 0.0.0.0:*
tcp LISTEN 0 4096 127.0.0.53%lo:domain 0.0.0.0:*
tcp LISTEN 0 128 0.0.0.0:ssh 0.0.0.0:*
tcp LISTEN 0 5 127.0.0.1:953 0.0.0.0:*
stcp LISTEN 0 2048 0.0.0.0:ldap 0.0.0.0:*
stcp LISTEN 0 10 [:1]:domain [:]*/
tcp LISTEN 0 10 [fe80::a00:27ff:fe89:5ba6]%enp0s3:domain [:]*/
tcp LISTEN 0 128 [:1]:ssh [:]*/
tcp LISTEN 0 5 [:1]:953 [:]*/
tcp LISTEN 0 2048 [:1]:ldap [:]*/
miadmin@rsp-ldap:~$
```

Una vez sepamos el puerto lo abriremos con el comando “[sudo ufw allow 389](#)”

```
miadmin@rsp-ldap:~$ sudo ufw allow 389
[sudo] password for miadmin:
Rule added
Rule added (v6)
```

Comprobamos que el servicio está activo “[sudo service slapd status](#)”

```
miadmin@rsp-ldap:~$ sudo service slapd status
● slapd.service - LSB: OpenLDAP standalone server (Lightweight Directory Access Protocol)
  Loaded: loaded (/etc/init.d/slapd; generated)
  Drop-In: /usr/lib/systemd/system/slapd.service.d
            └─slapd-remain-after-exit.conf
    Active: active (running) since Tue 2024-02-06 08:31:47 CET; 16min ago
      Docs: man:systemd-sysv-generator(8)
   Process: 696 ExecStart=/etc/init.d/slapd start (code=exited, status=0/SUCCESS)
     Tasks: 3 (limit: 2221)
    Memory: 6.8M
      CPU: 60ms
     CGroup: /system.slice/slapd.service
             └─757 /usr/sbin/slapd -h "ldap:/// ldap:///" -g openldap -u openldap -F /etc/ldap/sla>
feb 06 08:31:42 rsp-ldap systemd[1]: Starting LSB: OpenLDAP standalone server (Lightweight Director>
feb 06 08:31:43 rsp-ldap slapd[696]: * Starting OpenLDAP slapd
feb 06 08:31:47 rsp-ldap slapd[740]: @(#) $OpenLDAP: slapd 2.5.16+dfsg-0ubuntu0.22.04.2 (Jan 25 202>
                                         Ubuntu Developers <ubuntu-devel-discuss@lists.ubuntu.c>
feb 06 08:31:47 rsp-ldap slapd[757]: slapd starting
feb 06 08:31:47 rsp-ldap slapd[696]: ...done.
feb 06 08:31:47 rsp-ldap systemd[1]: Started LSB: OpenLDAP standalone server (Lightweight Directory>
Lines: 1-20/20 (END)
```

En nuestro caso, con la configuracion inicial no se había creado el objeto correctamente, para solucionarlo hemos usado el comando “[sudo dpkg-reconfigure slapd](#)” cambiando la configuracion del objeto rebeca

```
miadmin@rsp-ldap:~$ sudo dpkg-reconfigure slapd
```

Mostramos los objetos activos almacenados en el directorio con “[sudo slapcat](#)”

```
miadmin@rsp-ldap:~$ sudo slapcat
dn: dc=rebeca,dc=local
objectClass: top
objectClass: dcObject
objectClass: organization
o: rebeca
dc: rebeca
structuralObjectClass: organization
entryUUID: a462c23e-5910-103e-9441-2fa3c3ef0e27
creatorsName: cn=admin,dc=rebeca,dc=local
createTimestamp: 20240206075404Z
entryCSN: 20240206075404.3344772#000000#000000
modifiersName: cn=admin,dc=rebeca,dc=local
modifyTimestamp: 20240206075404Z
```

También podemos comprobar que el objeto rebeca.local existe con este otro comando “[sudo ldapsearch -xLLL -b dc=rebeca,dc=local](#)”:

```
miadmin@rsp-ldap:~$ sudo ldapsearch -xLLL -b dc=rebeca,dc=local
dn: dc=rebeca,dc=local
objectClass: top
objectClass: dcObject
objectClass: organization
o: rebeca
dc: rebeca
```

10.5 Crear usuarios y unidades organizativas

Para crear usuarios y unidades organizativas lo haremos desde ficheros que ejecutaremos en la máquina del servidor LDAP. Modificamos el fichero de las unidades organizativas facilitado por Amor “add_ou.ldif” (en nuestro caso tendremos 2: superior y medio):

```

1 ######
2 #####
3 #####
4 ##### Alta de Unidad Organizativas
5 #####
6 ######
7 dn: ou=superior,dc=rebeca,dc=local
8 objectClass: organizationalUnit
9 ou: superior
10
11 dn: ou=medio,dc=rebeca,dc=local
12 objectClass: organizationalUnit
13 ou: medio
14

```

Ahora modificamos el otro archivo que contiene los usuarios “add_user-ldif” para que aparezcan las especificaciones del usuario rebeca y amor

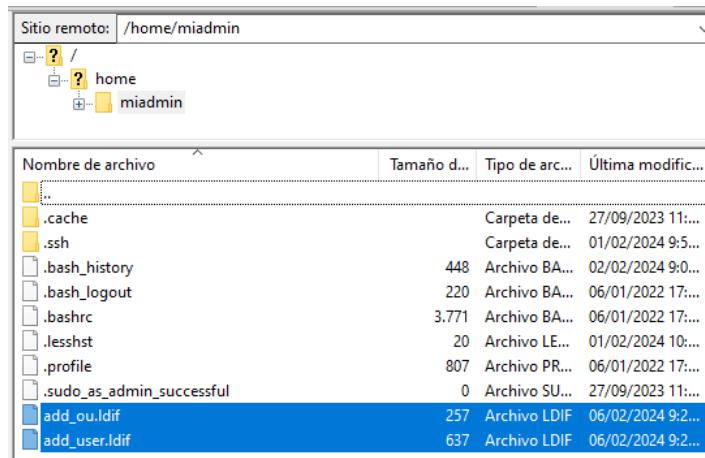
```

#####
#####
#####
##### Alta de usuarios
#####
#####
# Usuarios rebeca de la unidad organizativa superior
dn: uid=rebeca,ou=superior,dc=rebeca,dc=local
objectClass: inetOrgPerson
objectClass: organizationalPerson
objectClass: person
objectClass: top
uid: rebeca
sn: sanchez
cn: rebeca
mail: rebeca@rebeca.local
userPassword: paso

# Usuarios amor de la unidad organizativa superior
dn: uid=amor,ou=superior,dc=rebeca,dc=local
objectClass: inetOrgPerson
objectClass: organizationalPerson
objectClass: person
objectClass: top
uid: amor
sn: rodriguez
cn: amor
mail: amor@amor.local
userPassword: paso

```

Subimos ambos archivos al home de miadmin por filezilla



Usamos el comando “`sudo ldapadd -x -D cn=admin,dc=rebecca,dc=local -W -f add_ou.ldif`” para añadir los objetos de la unidad organizativa y seguido de esto repetimos el comando, pero con el fichero de los usuarios “`sudo ldapadd -x -D cn=admin,dc=rebecca,dc=local -W -f add_user.ldif`”

```
miadmin@rsp-ldap:~$ sudo ldapadd -x -D cn=admin,dc=rebecca,dc=local -W -f add_ou.ldif
[sudo] password for miadmin:
Enter LDAP Password:
adding new entry "ou=superior,dc=rebecca,dc=local"

adding new entry "ou=medio,dc=rebecca,dc=local"
```

```
miadmin@rsp-ldap:~$ sudo ldapadd -x -D cn=admin,dc=rebecca,dc=local -W -f add_user.ldif
Enter LDAP Password:
adding new entry "uid=rebecca,ou=superior,dc=rebecca,dc=local"

adding new entry "uid=amor,ou=superior,dc=rebecca,dc=local"
```

Hacemos un “`sudo slapcat |more`” para comprobar que se han añadido

<pre>dn: dc=rebecca,dc=local objectClass: top objectClass: dcObject objectClass: organization o: rebeca dc: rebeca structuralObjectClass: organization ← entryUUID: a462c23e-5910-103e-9441-2fa3c3ef0e27 creatorsName: cn=admin,dc=rebecca,dc=local createTimestamp: 202402060754042 entryCSN: 20240206075404.3344772#000000#000#000000 modifiersName: cn=admin,dc=rebecca,dc=local modifyTimestamp: 202402060754042 dn: ou=superior,dc=rebecca,dc=local ← objectClass: organizationalUnit ou: superior structuralObjectClass: organizationalUnit entryUUID: 5f4b00ee-5aa5-103e-9a60-3d4deb097684 creatorsName: cn=admin,dc=rebecca,dc=local createTimestamp: 202402080811142 entryCSN: 20240208081114.5915012#000000#000#000000 modifiersName: cn=admin,dc=rebecca,dc=local modifyTimestamp: 202402080811142 dn: ou=medio,dc=rebecca,dc=local ← objectClass: organizationalUnit ou: medio structuralObjectClass: organizationalUnit entryUUID: 5f4bcc5e-5aa5-103e-9a61-3d4deb097684 creatorsName: cn=admin,dc=rebecca,dc=local createTimestamp: 202402080811142 entryCSN: 20240208081114.5967342#000000#000#000000 modifiersName: cn=admin,dc=rebecca,dc=local modifyTimestamp: 202402080811142</pre>	<pre>dn: uid=rebecca,ou=superior,dc=rebecca,dc=local objectClass: inetOrgPerson objectClass: organizationalPerson objectClass: person objectClass: top uid: rebeca ← sn: sanchez cn: rebeca mail: rebeca@rebecca.local userPassword:: cGFzbw== structuralObjectClass: inetOrgPerson entryUUID: 75af1a32-5aa5-103e-9a62-3d4deb097684 creatorsName: cn=admin,dc=rebecca,dc=local createTimestamp: 202402080811522 entryCSN: 20240208081152.1574082#000000#000#000000 modifiersName: cn=admin,dc=rebecca,dc=local modifyTimestamp: 202402080811522 dn: uid=amor,ou=superior,dc=rebecca,dc=local objectClass: inetOrgPerson objectClass: organizationalPerson objectClass: person objectClass: top uid: amor ← sn: rodriguez cn: amor mail: amor@amor.local userPassword:: cGFzbw== structuralObjectClass: inetOrgPerson entryUUID: 75afc7b6-5aa5-103e-9a63-3d4deb097684 creatorsName: cn=admin,dc=rebecca,dc=local createTimestamp: 202402080811522 entryCSN: 20240208081152.1618512#000000#000#000000 modifiersName: cn=admin,dc=rebecca,dc=local modifyTimestamp: 202402080811522</pre>
--	---

10.6 Modificar usuarios

Creamos un fichero en el home del usuario admin llamado “modificar_datos.ldif” con el siguiente contenido (este fichero se divide en 3 bloques de líneas que tienen distintas modificaciones):

```
#añadir el móvil al usuario rebeca
dn: uid=rebeca,ou=superior,dc=rebeca,dc=local
add: mobile
Mobile: 654321987

dn: uid=amor,ou=superior,dc=rebeca,dc=local
delete: mail

dn: uid=amor,ou=superior,dc=rebeca,dc=local
changetype: modify
replace: mail
mail: amor@gmail.com
```

- El primer bloque de líneas, añade al usuario rebeca un teléfono móvil
- El segundo, elimina el correo de amor
- El tercero añade un email nuevo a amor

Para ejecutar el fichero usamos el comando “sudo ldapmodify -x -D cn=admin,dc=rebeca,dc=local -W -f modificar_datos.ldif” especificando el nombre del fichero nuevo

```
madmin@rsp-ldap:~$ sudo ldapmodify -x -D cn=admin,dc=rebeca,dc=local -W -f modificar_datos.ldif
Enter LDAP Password:
modifying entry "uid=rebeca,ou=superior,dc=rebeca,dc=local"
ldap_modify: Type or value exists (20)
        additional info: modify/add: mobile: value #0 already exists
```

Hacemos “sudo slapcat” para comprobar

```
dn: uid=rebeca,ou=superior,dc=rebeca,dc=local
objectClass: inetOrgPerson
objectClass: organizationalPerson
objectClass: person
objectClass: top
uid: rebeca
sn: sanchez
cn: rebeca
mail: rebeca@rebeca.local
userPassword:: cGFzbw==
structuralObjectClass: inetOrgPerson
entryUUID: 75af1a32-5aa5-103e-9a62-3d4deb097684
creatorsName: cn=admin,dc=rebeca,dc=local
createTimestamp: 20240208081152Z
mobile: 654321987 ←
entryCSN: 20240208083124.864449Z#000000#000#000000
modifiersName: cn=admin,dc=rebeca,dc=local
modifyTimestamp: 20240208083124Z
```

A continuación, vamos a modificar nuestros objetos para que pertenezcan a un grupo, para ello crearemos 3 ficheros (de borrado de usuarios, de creación de grupos y de creación de usuarios) y los ejecutaremos cada uno con el comando correspondiente.

Creamos un archivo “borrar_user.ldif” para borrar los usuarios rebeca y amor:

```
GNU nano 6.2                                borrar_user.ldif *
uid=rebeca,ou=superior,dc=rebeca,dc=local
uid=amor,ou=superior,dc=rebeca,dc=local
```

Ejecutamos el fichero con el comando “sudo ldapdelete -x -D cn=admin,dc=rebeca,dc=local -W -f borrar_user.ldif”

```
miadmin@rsp-ldap:~$ sudo ldapdelete -x -D cn=admin,dc=rebeca,dc=local -W -f borrar_user.ldif
Enter LDAP Password:
miadmin@rsp-ldap:~$ sudo slapcat
dn: dc=rebeca,dc=local
objectClass: top
objectClass: dcObject
objectClass: organization
o: rebeca
dc: rebeca
structuralObjectClass: organization
entryUUID: a462c23e-5910-103e-9441-2fa3c3ef0e27
creatorsName: cn=admin,dc=rebeca,dc=local
createTimestamp: 20240206075404Z
entryCSN: 20240206075404.334477Z#000000#000#000000
modifiersName: cn=admin,dc=rebeca,dc=local
modifyTimestamp: 20240206075404Z

dn: ou=superior,dc=rebeca,dc=local
objectClass: organizationalUnit
ou: superior
structuralObjectClass: organizationalUnit
entryUUID: 5f4b00ee-5aa5-103e-9a60-3d4deb097684
creatorsName: cn=admin,dc=rebeca,dc=local
createTimestamp: 20240208081114Z
entryCSN: 20240208081114.591501Z#000000#000#000000
modifiersName: cn=admin,dc=rebeca,dc=local
modifyTimestamp: 20240208081114Z

dn: ou=medio,dc=rebeca,dc=local
objectClass: organizationalUnit
ou: medio
structuralObjectClass: organizationalUnit
entryUUID: 5f4bcc5e-5aa5-103e-9a61-3d4deb097684
creatorsName: cn=admin,dc=rebeca,dc=local
createTimestamp: 20240208081114Z
entryCSN: 20240208081114.596734Z#000000#000#000000
modifiersName: cn=admin,dc=rebeca,dc=local
modifyTimestamp: 20240208081114Z
```

Creamos el fichero de creación de grupos llamado “add_group.ldif” con los grupos que queremos crear (en este caso creamos gr_alumnos y gr_profesores)

```
GNU nano 6.2                                add_group.ldif *
dn: cn=gr_alumnos,ou=superior,dc=rebeca,dc=local
objectClass: top
objectClass: posixGroup
gidNumber: 10000
cn: gr_alumnos

dn: cn=gr_profesores,ou=superior,dc=rebeca,dc=local
objectClass: top
objectClass: posixGroup
gidNumber: 10001
cn: gr_profesores
```

Ejecutamos el fichero de creación con “sudo ldapadd -x -D cn=admin,dc=rebeca,dc=local -W -f add_group.ldif”

```
miadmin@rsp-ldap:~$ sudo ldapadd -x -D cn=admin,dc=rebeca,dc=local -W -f add_group.ldif
Enter LDAP Password:
adding new entry "cn=gr_alumnos,ou=superior,dc=rebeca,dc=local"
adding new entry "cn=gr_profesores,ou=superior,dc=rebeca,dc=local"
```

```

dn: cn=gr_alumnos,ou=superior,dc=rebeca,dc=local
objectClass: top
objectClass: posixGroup
gidNumber: 10000
cn: gr_alumnos
structuralObjectClass: posixGroup
entryUUID: d868cc84-5b6f-103e-9b1d-7f45f085fe64
creatorsName: cn=admin,dc=rebeca,dc=local
createTimestamp: 20240209082036Z
entryCSN: 20240209082036.130460Z#000000#000#000000
modifiersName: cn=admin,dc=rebeca,dc=local
modifyTimestamp: 20240209082036Z

dn: cn=gr_profesores,ou=superior,dc=rebeca,dc=local
objectClass: top
objectClass: posixGroup
gidNumber: 10001
cn: gr_profesores
structuralObjectClass: posixGroup
entryUUID: d870e18a-5b6f-103e-9b1e-7f45f085fe64
creatorsName: cn=admin,dc=rebeca,dc=local
createTimestamp: 20240209082036Z
entryCSN: 20240209082036.183451Z#000000#000#000000
modifiersName: cn=admin,dc=rebeca,dc=local
modifyTimestamp: 20240209082036Z

```

Por último, creamos el fichero de creación de grupos llamado “add_usuarios_grupos.ldif” con los usuarios y los grupos a los que pertenecen. Para generar la contraseña cifrada de los usuarios usaremos el comando “[sudo slappasswd](#)” que devolverá la contraseña cifrada con SSHA que escribamos

```

miadmin@rsp-ldap:~$ sudo slappasswd
New password:
Re-enter new password:
{SSHA}ne+xp8ucSGHHAAMy1as1BGpaRganeamW
. . . . .

GNU nano 6.2                                     add_usuarios_grupos.ldif *
dn: uid=rebeca,ou=superior,dc=rebeca,dc=local
objectClass: top
objectClass: posixAccount
objectClass: inetOrgPerson
objectClass: person
cn: rebeca
uid: rebeca
ou: gr_alumnos
uidNumber: 2001                                     ←
gidNumber: 10000                                     ←
homeDirectory: /home/rebeca
loginShell: /bin/bash
userPassword: {SSHA}ne+xp8ucSGHHAAMy1as1BGpaRganeamW
sn: Sanchez
mail: rebeca@gmail.com
givenName: rebeca

dn: uid=amor,ou=superior,dc=rebeca,dc=local
objectClass: top
objectClass: posixAccount
objectClass: inetOrgPerson
objectClass: person
cn: amor
uid: amor
ou: gr_profesores
uidNumber: 2000                                     ←
gidNumber: 10001                                     ←
homeDirectory: /home/amor
loginShell: /bin/bash
userPassword: {SSHA}ne+xp8ucSGHHAAMy1as1BGpaRganeamW
sn: Rodriguez
mail: amor@gmail.com
givenName: amor

```

Ejecutamos el fichero de creación con “sudo ldapadd -x -D cn=admin,dc=rebeca,dc=local -W -f add_usuarios_grupos.ldif”

```
miadmin@rsp-ldap:~$ sudo ldapadd -x -D cn=admin,dc=rebeca,dc=local -W -f add_usuarios_grupos.ldif
Enter LDAP Password:
adding new entry "uid=rebeca,ou=superior,dc=rebeca,dc=local"
adding new entry "uid=amor,ou=superior,dc=rebeca,dc=local"

cn: rebeca
uid: rebeca
ou: gr_alumnos
uidNumber: 2001
gidNumber: 10000 ←
homeDirectory: /home/rebeca
loginShell: /bin/bash
userPassword:: e1NTSEF9bmUreHA4dWNTR0hIQUFNeTFhc2xCR3BhUmdbmVhbVc=
sn: Sanchez
mail: rebeca@gmail.com
givenName: rebeca
structuralObjectClass: inetOrgPerson
entryUUID: fa2d54c8-5b80-103e-9b1f-7f45f085fe64
creatorsName: cn=admin,dc=rebeca,dc=local
createTimestamp: 20240209102314Z
entryCSN: 20240209102314.227694Z#000000#000#000000
modifiersName: cn=admin,dc=rebeca,dc=local
modifyTimestamp: 20240209102314Z

dn: uid=amor,ou=superior,dc=rebeca,dc=local
objectClass: top
objectClass: posixAccount
objectClass: inetOrgPerson
objectClass: person
cn: amor
uid: amor
ou: gr_profesores ←
uidNumber: 2000
gidNumber: 10001
homeDirectory: /home/amor
loginShell: /bin/bash
userPassword:: e1NTSEF9bmUreHA4dWNTR0hIQUFNeTFhc2xCR3BhUmdbmVhbVc=
sn: Rodriguez
mail: amor@gmail.com
givenName: amor
structuralObjectClass: inetOrgPerson
entryUUID: fa30af9c-5b80-103e-9b20-7f45f085fe64
creatorsName: cn=admin,dc=rebeca,dc=local
createTimestamp: 20240209102314Z
entryCSN: 20240209102314.249681Z#000000#000#000000
modifiersName: cn=admin,dc=rebeca,dc=local
modifyTimestamp: 20240209102314Z
```

b. WXED – WINDOWS X

1. Nombre y configuración de red



2. Cuentas administradoras y cuenta de desarrollador

2.1 Conexión SSH desde Windows 10

Ahora pasamos a trabajar desde el Windows 10 de nuestra maquina anfitriona. Abrimos el cmd y escribimos “`ssh miadmin@192.168.3.204`” para conectarnos desde nuestro windows al terminal de nuestro entorno de desarrollo. Si al intentar conectarte te da problemas porque ya has usado la IP de la maquina nueva para otra máquina, tendremos que entrar en C: > Usuarios > TuUsuario > .ssh y eliminamos un archivo llamado “known_hosts” y volver a ejecutar el comando “`ssh miadmin@192.168.3.204`”.

Equipo > SISTEMA (C:) > Usuarios > daw2 > .ssh				
Nombre	Fecha de modificación	Tipo	Tamaño	
known_hosts	03/10/2023 10:50	Archivo	1 KB	

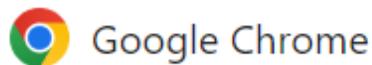
```
Active ESM Apps para recibir futuras actualizaciones
Vea https://ubuntu.com/esm o ejecute «sudo pro status»
Failed to connect to https://changelogs.ubuntu.com/meta
*** System restart required ***
Last login: Tue Oct  3 08:17:46 2023
miadmin@rsp-usedlimpia:~$
```

2.2 Cuentas administradoras del sistema operativo

```
C:\Users\daw2>net users  
Cuentas de usuario de \\IS32WX14  
-----  
admin           Administrador      bachillerato  
daw1            daw2             DefaultAccount  
eso              Invitado          smr1  
smr2            WDAGUtilityAccount  
Se ha completado el comando correctamente.  
  
C:\Users\daw2>net localgroup  
Alias para \\IS32WX14  
-----  
*Administradores  
*Administradores de Hyper-V  
*Duplicadores  
*IIS_IUSRS  
*Invitados  
*Lectores del registro de eventos  
*Operadores criptográficos  
*Operadores de asistencia de control de acceso  
*Operadores de configuración de red  
*Operadores de copia de seguridad  
*System Managed Accounts Group  
*Usuarios  
*Usuarios avanzados  
*Usuarios COM distribuidos  
*Usuarios de administración remota  
*Usuarios de escritorio remoto  
*Usuarios del monitor de sistema  
*Usuarios del registro de rendimiento  
Se ha completado el comando correctamente.
```

3. Navegadores

3.1 Google Chrome



- La actualización ya casi ha terminado. Reinicia Chrome para completar la actualización.
- Versión 120.0.6099.72 (Build oficial) (64 bits)

3.2 Microsoft Edge

Acerca de

Microsoft Edge
Versión 120.0.2210.61 (Compilación oficial) (64 bits)

Microsoft Edge está actualizado.

Descargar las actualizaciones en conexiones de uso medido

Descargar automáticamente las actualizaciones a través de redes de uso medido (por ejemplo, redes de telefonía móvil) e instalarlas al reiniciar el explorador. Pueden aplicarse cargos.

Este explorador se ha hecho gracias al proyecto de código abierto [Chromium](#) y a otro [software de código abierto](#).

Microsoft Edge
© 2023 Microsoft Corporation. Todos los derechos reservados.
[Términos de uso](#) - [Declaración de privacidad](#) - [Contrato de servicios de Microsoft](#)

3.2 Mozilla Firefox

Actualizaciones de Firefox

Mantenga Firefox actualizado para un rendimiento, estabilidad y seguridad óptimos.

Versión 120.0.1 (64-bit) [Novedades](#)

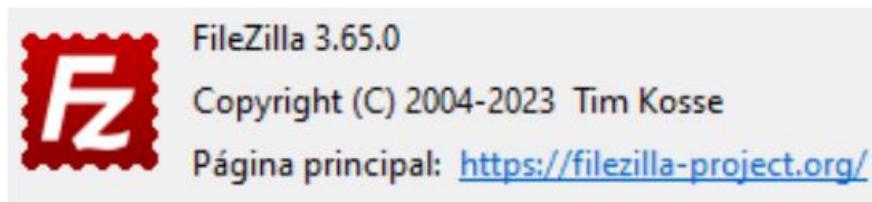
[Mostrar historial de actualizaciones...](#)

Firefox está actualizado

[Buscar actualizaciones](#)

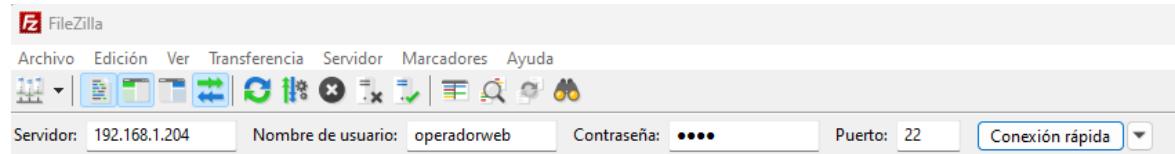
4. FileZilla

4.1 Versión



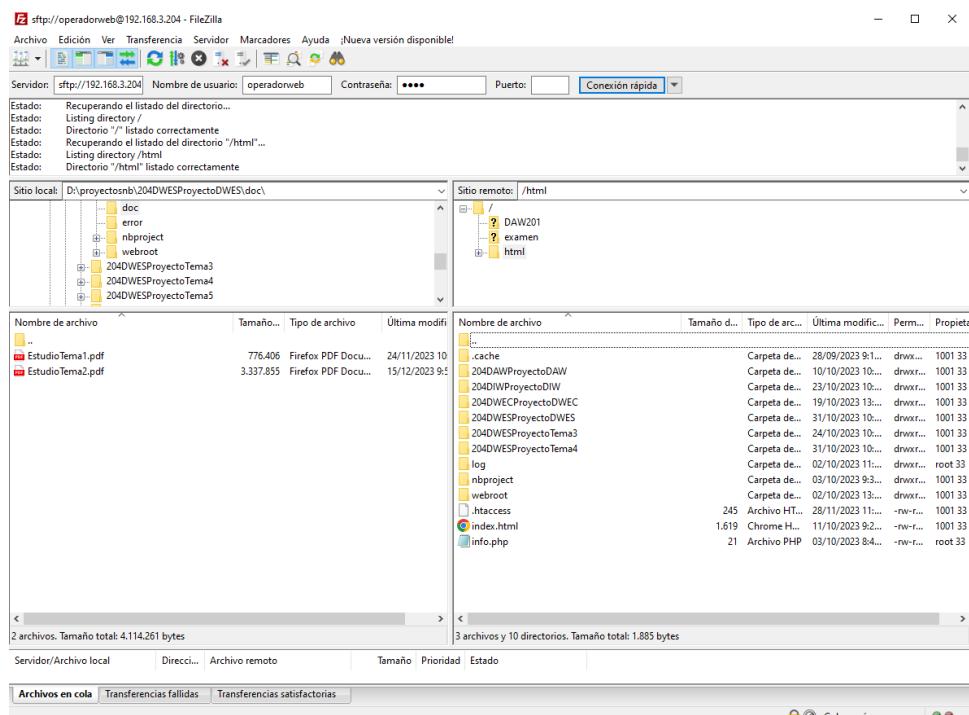
4.2 Conexión a un servidor

Para acceder a los ficheros de nuestro servidor web con un entorno gráfico, usaremos FileZilla. Para realizar una conexión tenemos que ir rellenando los campos que aparecen en la parte superior de la ventana siguiendo estas pautas:



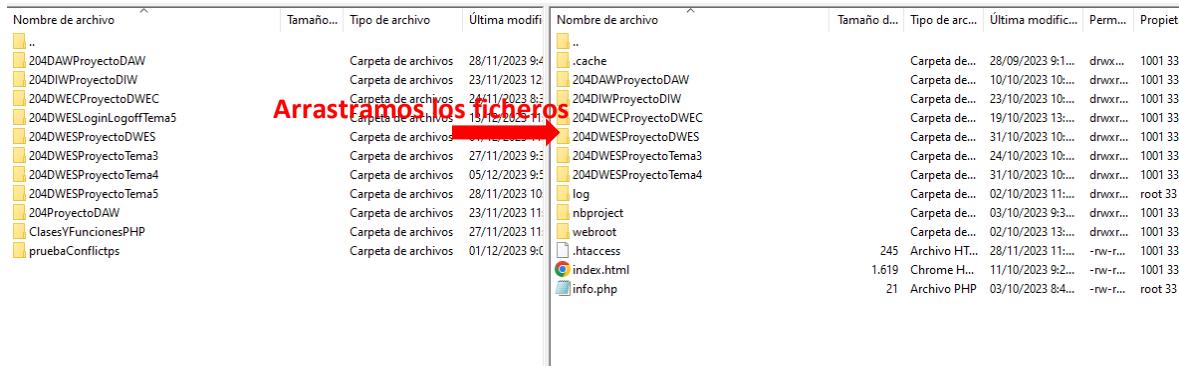
- En campo “Servidor” escribimos la IP de nuestro servidor (<192.168.1.204>)
- En “Nombre de usuario” escribimos nuestra cuenta de desarrollo y [hosting virtual \(operadorweb\)](#)
- El campo “contraseña” lo rellenamos con la contraseña del usuario ([paso](#))
- En el campo “Puerto” escribimos el puerto [22](#) que es el que utiliza SSH

Una vez, rellenados los campos, Hacemos click sobre “Conexión Rápida” y en la ventana de la derecha aparece la estructura de directorios que tiene el directorio raíz del servidor entrando al usuario operadorweb, en este caso como está enjaulado, solo podemos visualizar el interior de la carpeta public_html.



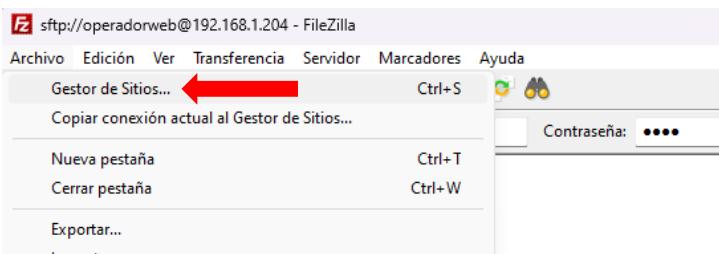
4.3 Transferencia de archivos

Para realizar transferencias de archivos tenemos que tener en la ventana de la izquierda nuestro directorio local y en la derecha el directorio correspondiente del servidor. Para subirlos o bajarlos basta con arrastrar los archivos que queramos subir (o descargar) y soltarlos en su destino

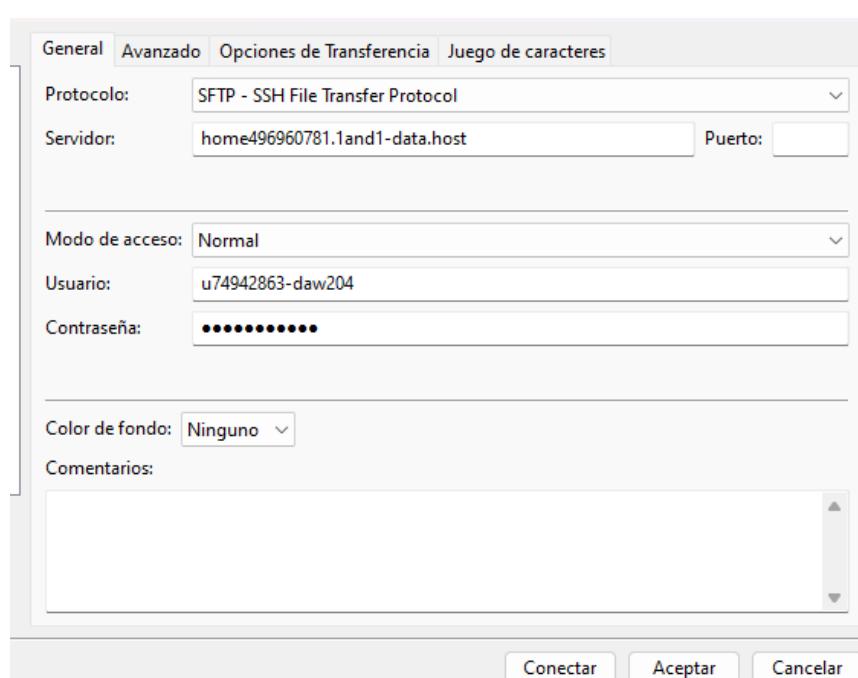


4.4 Gestión de sitios

Para hacer una conexión de FileZilla con nuestro EE tenemos que crear un sitio virtual donde se encontrará la configuración de conexión con IONOS. En primer lugar, iremos a la pestaña izquierda superior “Archivo>Gestor de sitios”



En la nueva ventana, seleccionamos el botón “Nuevo sitio”, elegimos un nombre y completamos la configuración de la siguiente manera:



Cuando pulsamos el botón “Conectar” se establecerá la conexión y la interfaz de FileZilla se adaptará al contenido de nuestro servidor de one&one

The screenshot shows the FileZilla interface with a connection established to the server sftp://192.168.1.204. The status bar at the top displays the connection details: Servidor: sftp://192.168.1.204, Nombre de usuario: operadorweb, Contraseña: ****, Puerto: 22, and Conexión rápida.

The left pane shows the local directory structure on the user's computer:

```

Síntesis local: E:\proyectosNB\204DWESELoginLogoffTema5\
├── proyectosNB
│   ├── 204DAWProyectoDAW
│   ├── 204DWESELoginLogoffTema5
│   └── 204DWESEMtoDepartamentosTema4

```

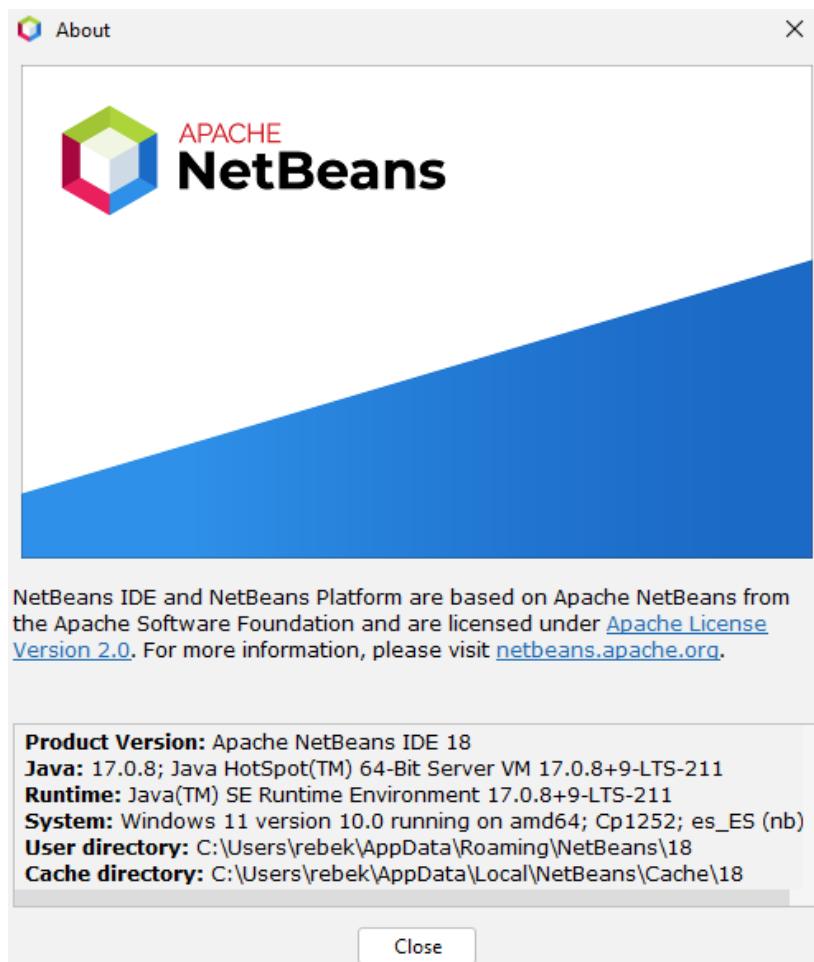
The right pane shows the remote directory structure on the server:

Nombre de archivo	Tamaño d...	Tipo de arc...	Última modificación	Permisos
..		Carpeta de...	23/11/2023 9:42:28	drwx---r-x 9
204DAWProyectoDAW		Carpeta de...	23/11/2023 9:54:22	drwx---r-x 9
204DIWProyectoDIW		Carpeta de...	23/11/2023 10:13:47	drwx---r-x 9
204DWECProyectoDWECE		Carpeta de...	13/12/2023 10:07:42	drwx---r-x 9
204DWESELoginLogoffTema5		Carpeta de...	01/12/2023 9:42:44	drwx---r-x 9
204DWESEProyectoDWESE		Carpeta de...	01/12/2023 9:47:26	drwx---r-x 9
204DWESEProyectoTema3		Carpeta de...	23/11/2023 10:35:09	drwx---r-x 9
204DWESEProyectoTema4		Carpeta de...	05/12/2023 9:58:36	drwx---r-x 9
204DWESEProyectoTema5		Carpeta de...	01/12/2023 9:47:26	drwx---r-x 9
log		Carpeta de...	23/11/2023 9:23:51	drwx---r-x 9
webroot		Carpeta de...	23/11/2023 9:23:52	drwx---r-x 9
.gitignore	12	Archivo de...	23/11/2023 9:23:50	-rw----r-- 9
.htaccess	229	Archivo HT...	23/11/2023 9:23:50	-rw----r-- 9
file.txt	0	Documento de tex...	04/12/2023 16:36:46	-rw----r-- 9
indexLoginLogoffTema5.php	1.660	Archivo de origen...	04/12/2023 16:38:08	-rw----r-- 9
LICENSE.txt	35.823	Documento de tex...	04/12/2023 16:36:46	-rw----r-- 9
README.md	667	Archivo de origen...	04/12/2023 16:36:46	-rw----r-- 9

At the bottom, there are tabs for Servidor/Archivo local, Direcci..., Archivo remoto, Tamaño, Prioridad, and Estado.

5. NetBeans

5.1 Version



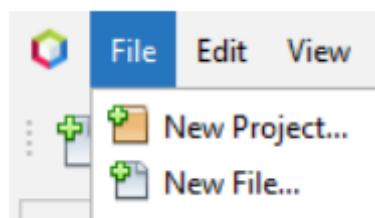
5.1 Instalación y configuración inicial (pluggings)

????????? No la hemos realizado en clase ????????

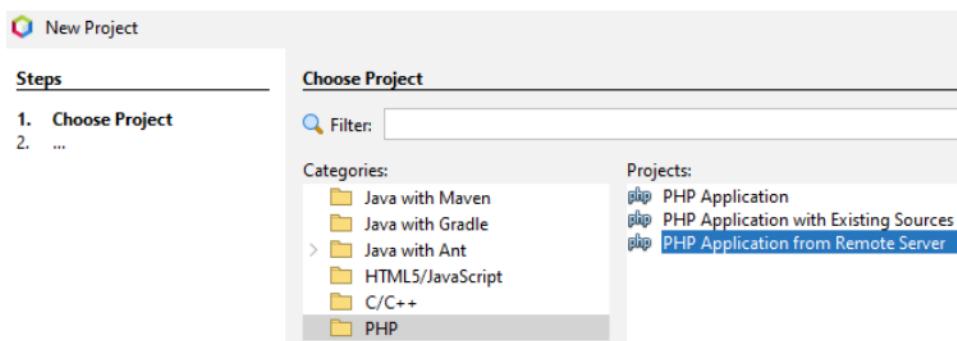
5.2 Creación de proyectos, modificación, borrado, prueba

Para crear un proyecto en NetBeans basado en una arquitectura AMP seguiremos los siguientes pasos: (capturas de pantalla por Carlos García Cachón)

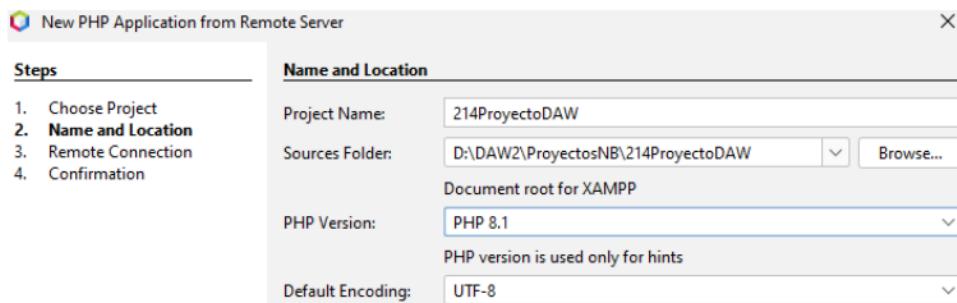
Seleccionamos "File>New Project"



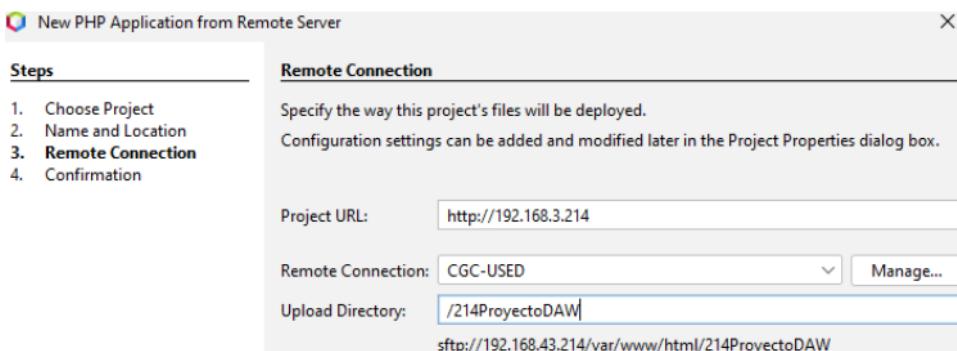
En la siguiente ventana, elegiremos PHP como lenguaje de programación y la opción "Aplicación PHP desde servidor remoto"



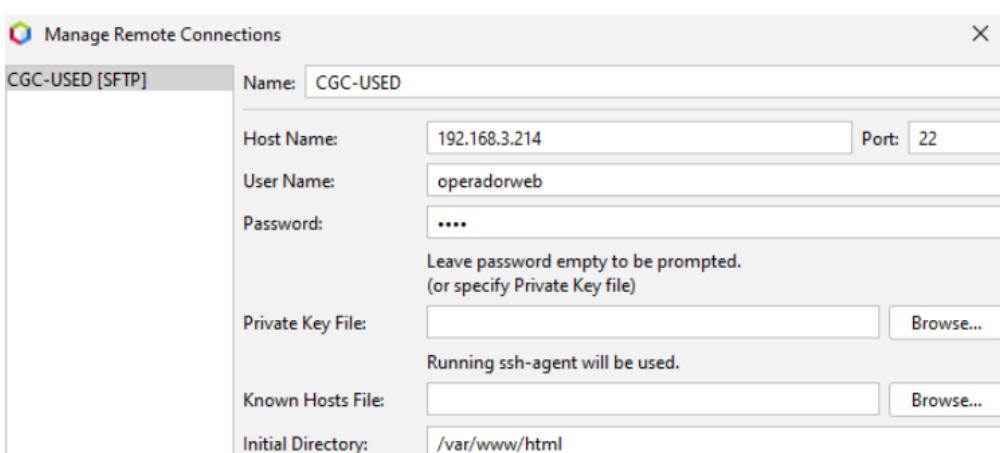
A continuación, elegiremos el nombre del proyecto, la carpeta contenedora del proyecto en nuestra maquina anfitriona y la version de php que vamos a utilizar (en nuestro caso la 8.1)



En la siguiente ventana, introducimos la IP de nuestro servidor, elegimos la carpeta donde se alojará el proyecto en el servidor (tendrá el mismo nombre que el proyecto) y creamos una nueva conexión a un servidor remoto

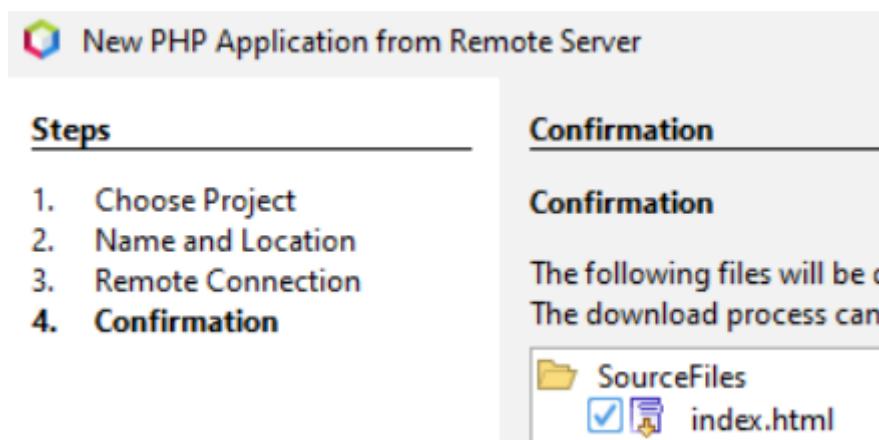


Aquí se muestran los parámetros que debe de tener nuestra conexión:

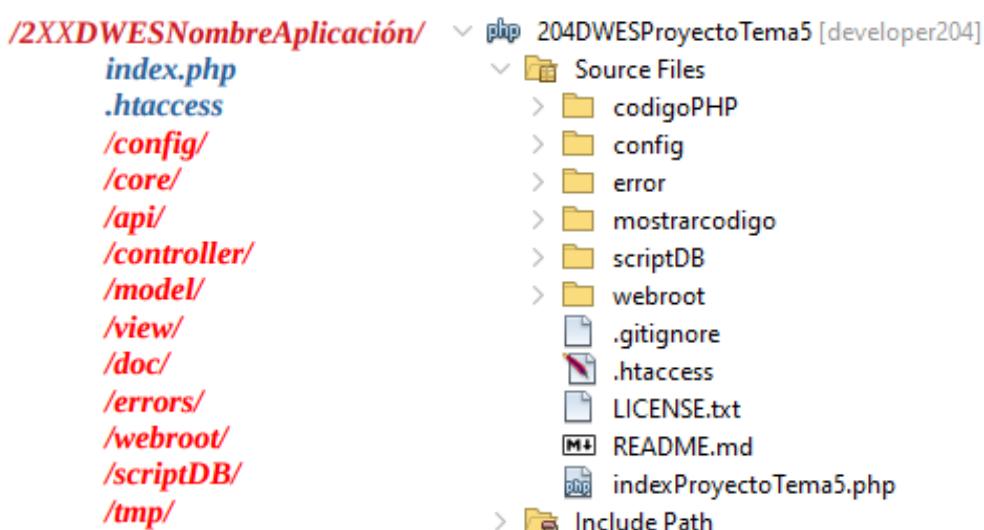


- Name: el nombre personalizado de la conexión
- Host Name: IP del servidor
- Port: 22 (por el que establece la conexión SSH)
- User Name: el nombre del usuario administrador del hosting virtual
- Password: contraseña del usuario
- Initial Directory: directorio padre

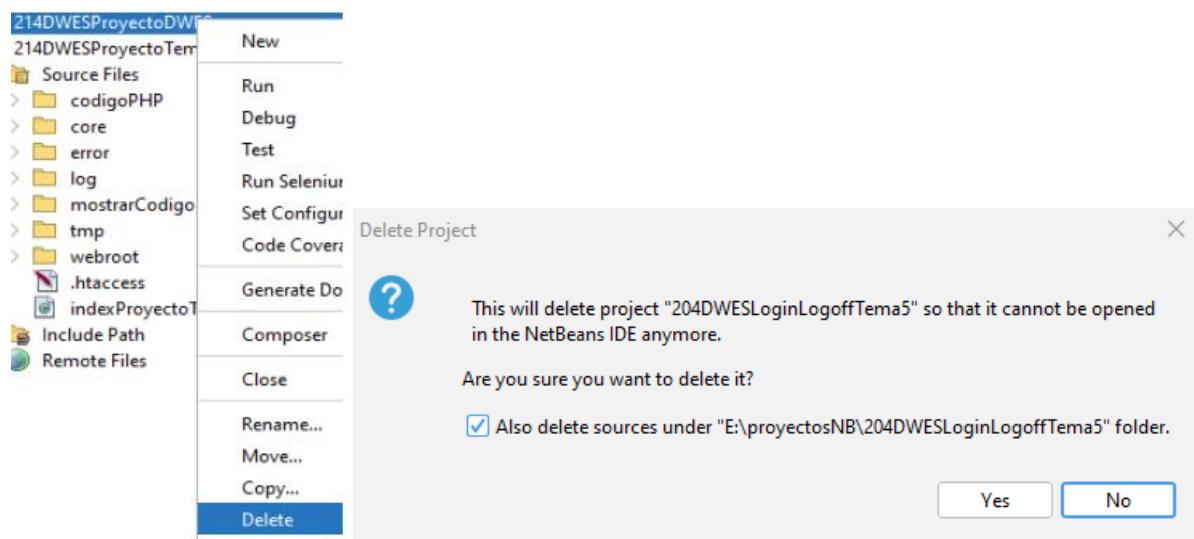
Hacemos un “Test Connection” y si todo ha salido bien, seguiremos con la creación del proyecto. En la siguiente ventana se conectará al servidor y solo te podrá dejar crear el proyecto si ese directorio (html/204DAWProyectoDAW) en el servidor contiene algún archivo. Para solucionar esto, podemos crear un fichero auxiliar (index.html o file.txt) desde FileZilla para más tarde eliminarlo



Una vez creado comenzaríamos a realizar modificaciones para que la estructura de directorios del proyecto se adapte a los estándares y en los archivos de código para darle funcionalidad al proyecto

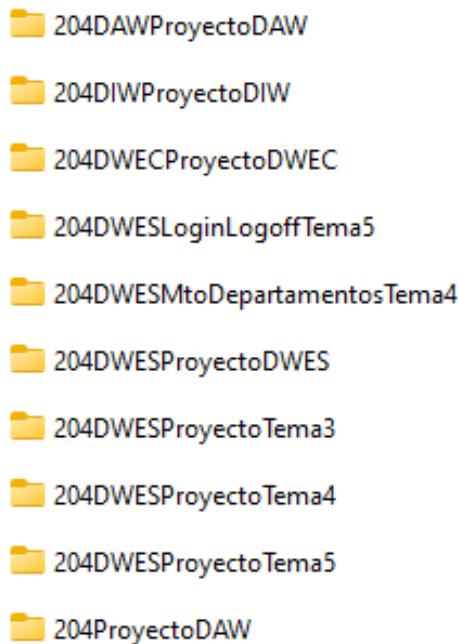


Para realizar el borrado de proyectos es tan simple como hacer click derecho encima del proyecto y seleccionar la opción “delete” (es muy importante que seleccionemos la opción de eliminado seguro para que NetBeans se asegure también de borrar todos los ficheros que se encuentran en el directorio del proyecto en nuestro equipo).

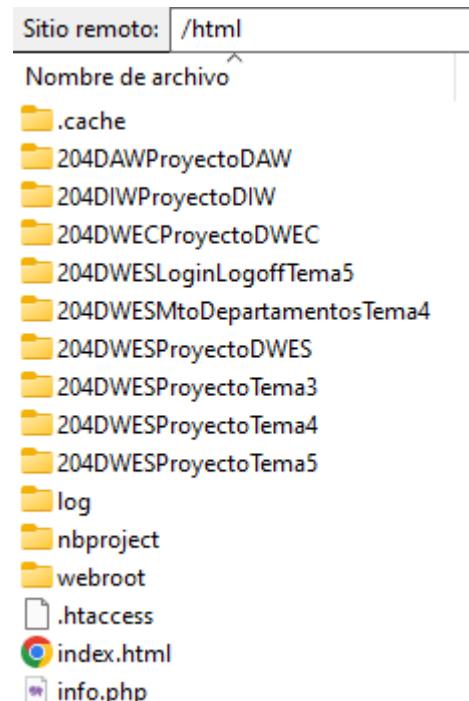


5.3 Conexión al servidor remoto SFTP. (Almacenamiento local/almacenamiento remoto)

1. Directorios en local (/proyectosNB)

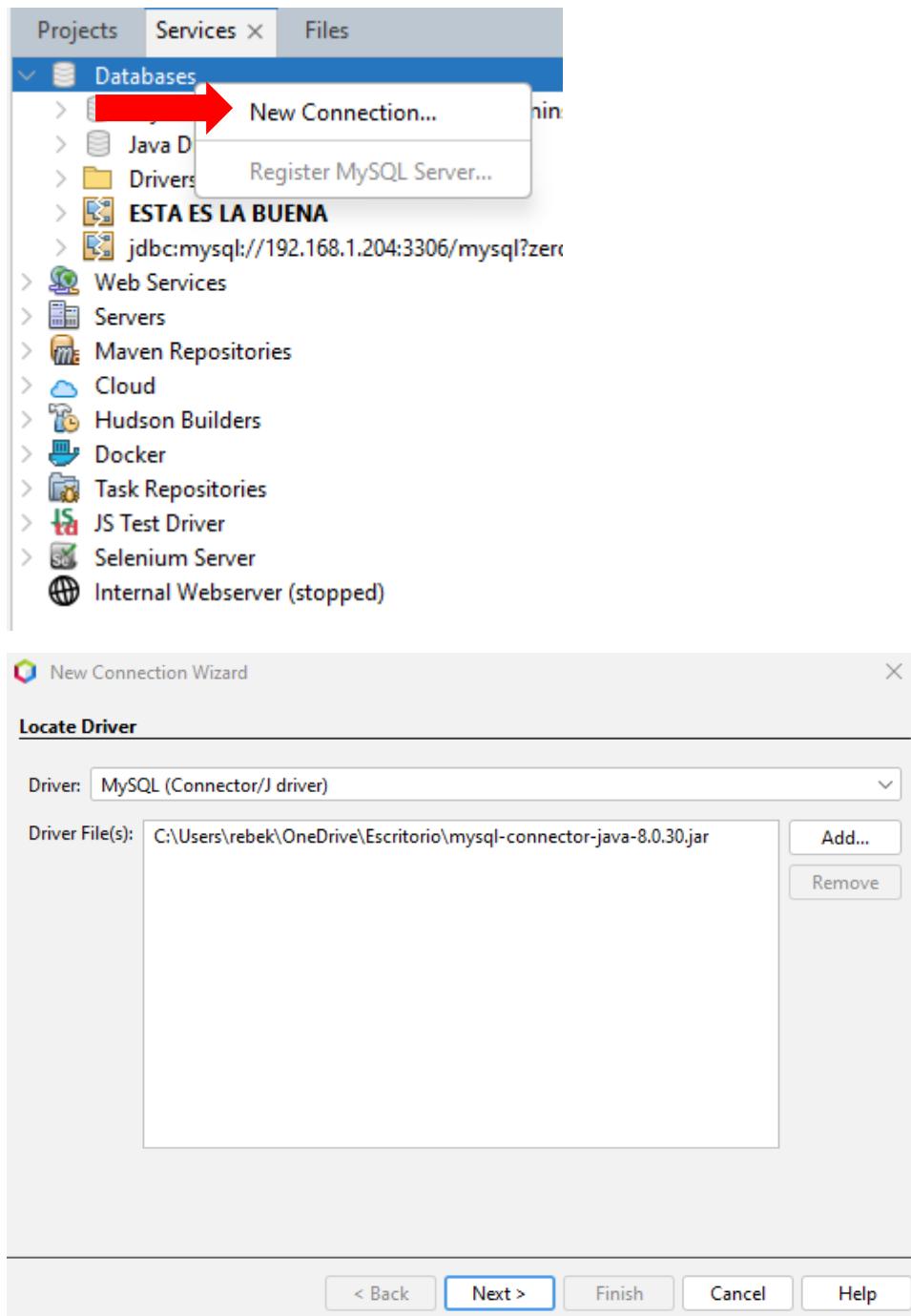


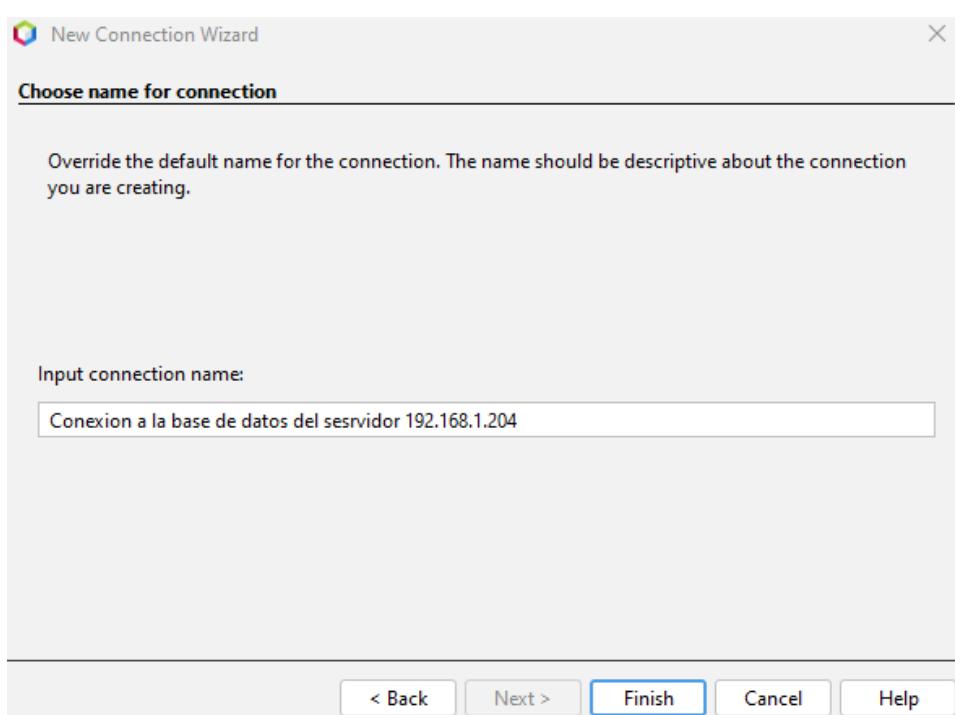
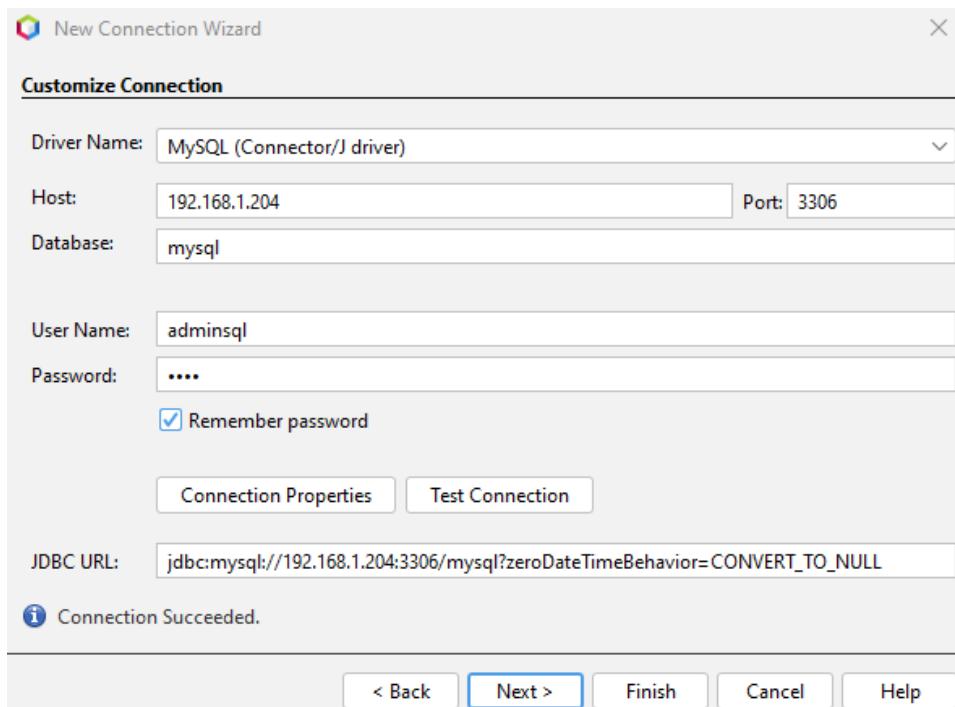
2. Directorios en remoto (/html)

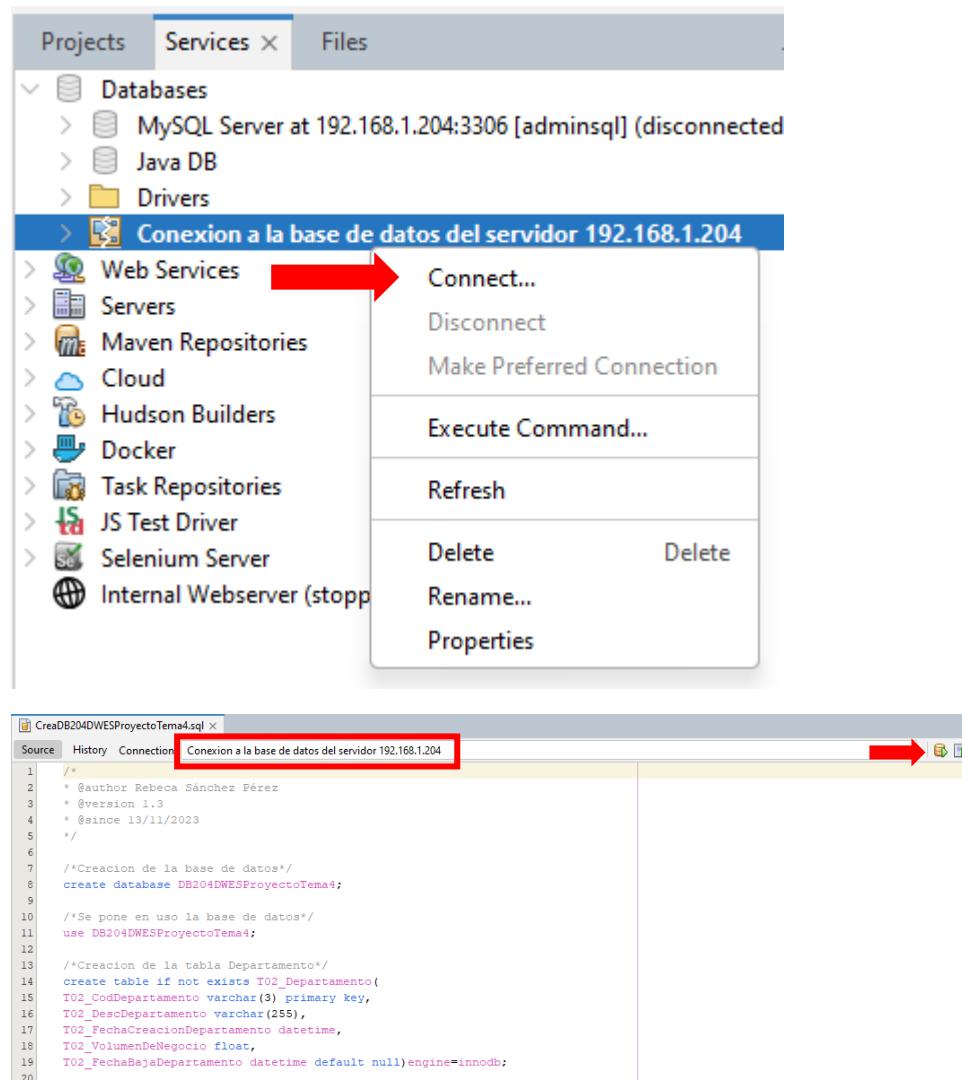


5.4 Administración de la base de datos

Dentro de NetBeans nos iremos a la pestaña de “services” del menú de la izquierda y una vez allí, podemos apreciar un apartado llamado “Databases”. Para realizar una nueva conexión haremos click derecho sobre este y seleccionamos “New connection”







5.5 Depuración - Configuración de la ejecución para la depuración

5.6 Paso a explotación

Puedes encontrar el proceso de paso a explotación en el apartado de Git de creación de una release

5.7 PHP Doc

Usage

Easy to install, and to use!

Getting started using the PHAR <p>Start by downloading our latest PHAR file and putting it in a nice and cozy place on your hard drive.</p> <pre>\$ wget https://phpdoc.org/phpDocumentor.phar</pre> <p>After you downloaded the PHAR file, don't forget to power it up by making it executable.</p> <pre>\$ chmod +x phpDocumentor.phar</pre> <p>Also, you may want to consider renaming it to phpdoc; we'll understand.</p>	Getting started using Docker <p>Use our very own Docker image, no installation needed!</p> <p>Treat our docker image like you treat all your other utility images. Just don't forget to volume mount your current directory to /data inside the container. Remember: No data, No docs.</p> <pre>\$ docker run --rm -v \${PWD}:/data phpdoc/phpdoc:3</pre> <p>Tada!</p>
---	---

> DATOS (D:) > _Software > phpdo	
Nombre	Fecha de modificación
phpDocumentor.phar	25/01/2024

En php.net en el apartado de la versión correspondiente

Old Stable PHP 8.1.27 (Changelog)

- [php-8.1.27.tar.gz \(sig\) \[19,388Kb\]](#)
sha256: 9aa5d7a29389d799885d90740932697006d5d0f55d1def67678e0c...
- [php-8.1.27.tar.bz2 \(sig\) \[15,282Kb\]](#)
sha256: a15fd73ea44f2df30b07d24786e07d1948b0ea3eed0b8b845735d5...
- [php-8.1.27.tar.xz \(sig\) \[11,636Kb\]](#)
sha256: 479e65c3f05714d4aace1370e617d78e49e996ec7a7579a5be4753...
- [Windows downloads](#) 

[GPG Keys for PHP 8.1](#)

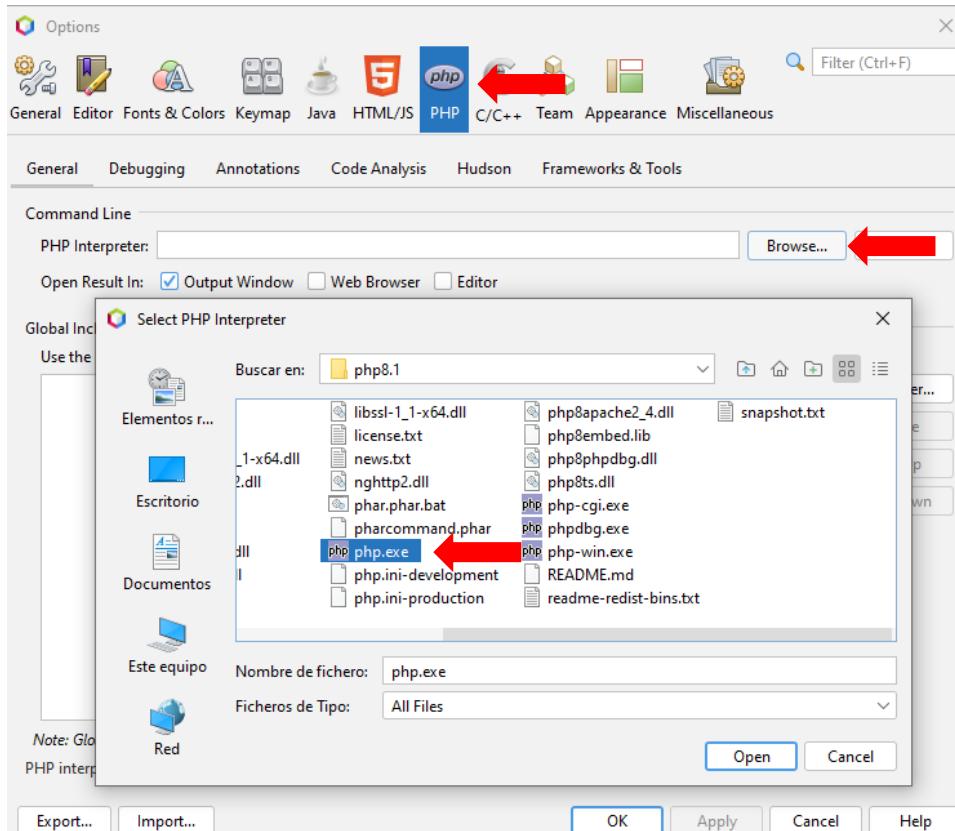
VS16 x64 Thread Safe (2023-Dec-19 23:49:00)

- [Zip \[29.43MB\]](#) 
- [Debug Pack \[24.91MB\]](#)
sha256: 20e8d445c20a1c0573d3faa2f41d9982611437ca5aad9a4e4b6ed66464a15630
- [Development package \(SDK to develop PHP extensions\) \[1.21MB\]](#)
sha256: 3344e9454066ffc49ab128c2cf6e705e9e9289fa55505848e2d0b25d44526dee

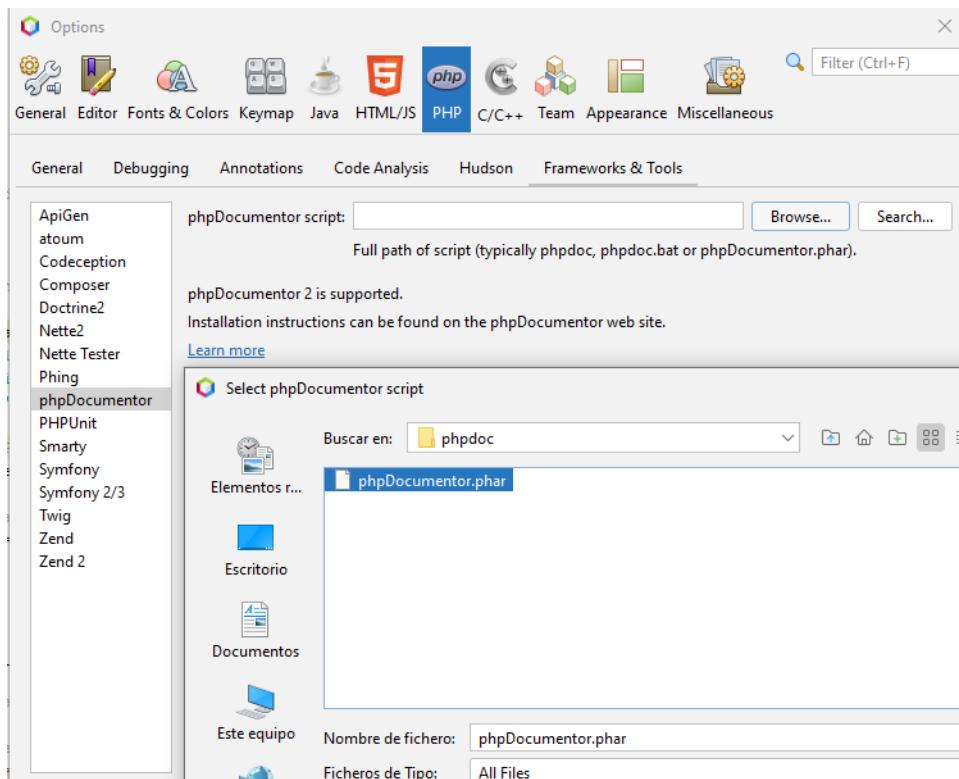
Descomprimimos el zip descargado en la carpeta software dentro de un directorio llamado "php8.1" (Datos(D:)>_Software>php8.1)

> DATOS (D:) > _Software > php8.1	
Nombre	Fecha de modificación
dev	25/01/2024 9:09
ext	25/01/2024 9:09
extras	25/01/2024 9:09
lib	25/01/2024 9:09
deplister.exe	25/01/2024 9:09
glib-2.dll	25/01/2024 9:09
gmodule-2.dll	25/01/2024 9:09
icudt70.dll	25/01/2024 9:09

En NetBeans accedemos al apartado de opciones en “Tools>Options” y seleccionamos la pestaña de PHP para seleccionar el intérprete de PHP que hemos instalado, para ello hacemos click en “Browse...” y seleccionamos el archivo “php.exe” que se encuentra en la ruta de la captura anterior (Datos(D:)>_Software>php8.1)

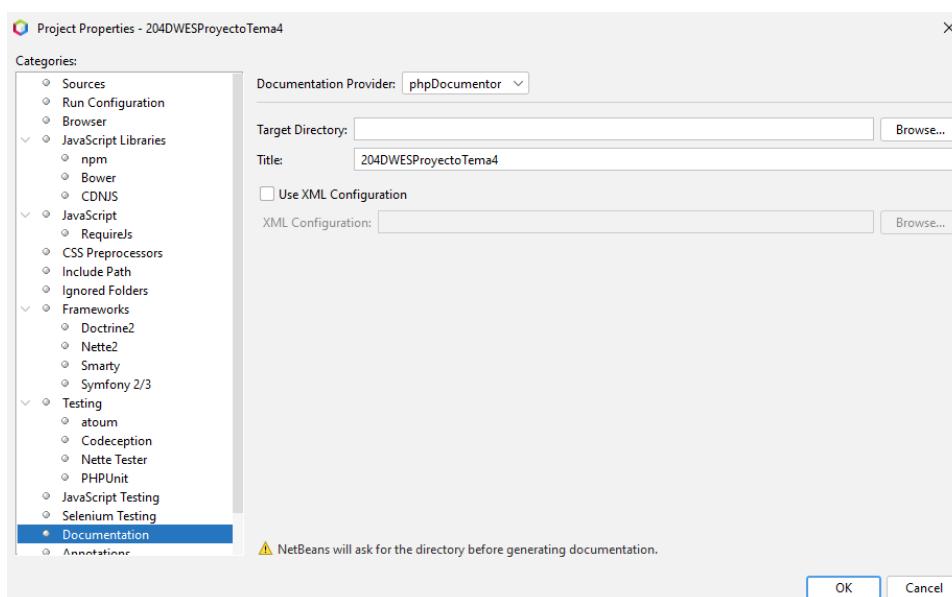
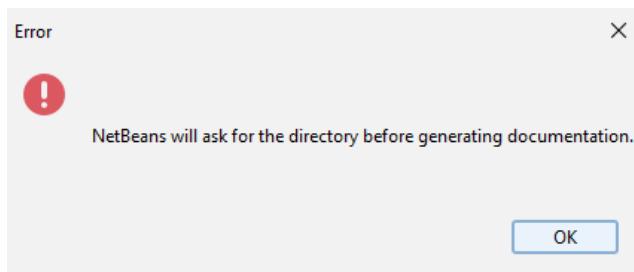


A continuación, cambiamos la configuración del apartado phpDocumentor seleccionando el archivo descargado anteriormente en



Apply y ok

Encima de un proyecto click derecho "Generate Documentation>phpDocumentor"



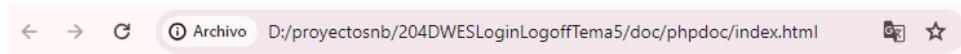
Abir CMD de windows copia pega ese comando

```
C:\Users\daw2>"D:\_Software\php8.1\php.exe" "D:\_Software\phpdoc\phpDocumentor.phar" "run" "--ansi" "--directory" "D:/proyectosnb/204DWESLoginLogoffTema5" "--target" "D:/proyectosnb/204DWESLoginLogoffTema5/doc/phpdoc" "--title" "204DWESLoginLogoffTema5"
phpDocumentor v3.4.3

Parsing files
13/13 [=====] 100%
Applying transformations (can take a while)

All done in 1 seconds!
```

> DATOS (D:) > proyectosnb > 204DWESLoginLogoffTema5 > doc > phpdoc			
Nombre	Fecha de modificación	Tipo	Tamaño
classes	25/01/2024 9:19	Carpetas de archivos	
css	25/01/2024 9:19	Carpetas de archivos	
files	25/01/2024 9:19	Carpetas de archivos	
graphs	25/01/2024 9:19	Carpetas de archivos	
indices	25/01/2024 9:19	Carpetas de archivos	
js	25/01/2024 9:19	Carpetas de archivos	
namespaces	25/01/2024 9:19	Carpetas de archivos	
packages	25/01/2024 9:19	Carpetas de archivos	
reports	25/01/2024 9:19	Carpetas de archivos	
index.html	25/01/2024 9:19	Chrome HTML Documento	29 KB



204DWESLoginLogoffTema5

Search (Press "/" to focus)

Packages

- Application
- Validacion

Reports

- Deprecated
- Errors
- Markers

Indices

- Files

Documentation

Table of Contents

Packages

- P Application
- P Validacion

Classes

- C validacionFormularios

EN UBUNTU

```
miadmin@rsp-used:~$ wget https://phpdoc.org/phpDocumentor.phar
```

```
miadmin@rsp-used:~$ sudo mv phpDocumentor.phar /usr/local/bin/phpdoc
[sudo] password for miadmin:

sudo chmod +x /usr/local/bin/phpdoc_
```

```
miadmin@rsp-used:/var/www/html/204DWESLoginLogoffTema5/doc$ ls -l /usr/local/bin
total 14064
-rwxrwxr-x 1 miadmin miadmin 14399113 oct  4 20:35 phpdoc
miadmin@rsp-used:/var/www/html/204DWESLoginLogoffTema5/doc$
```

```
miadmin@rsp-used:~$ phpdoc --version
Xdebug: [Step Debug] Could not connect to debugger
xdebug.client_host/xdebug.client_port) :-(

phpDocumentor vv3.4.3
```

```
miadmin@rsp-used:/var/www/html/204DWESLoginLogoffTema5/doc$ sudo phpdoc run -d /var/www/html/204DWES
LoginLogoffTema5/dc/phpdoc
Xdebug: [Step Debug] Could not connect to debugging client. Tried: localhost:9003 (fallback through
xdebug.client_host/xdebug.client_port) :-(

[warning] Your documentation set seems to be empty!
Parsing files

Applying transformations (can take a while)

All done in 0 seconds!
Xdebug: [Step Debug] Could not connect to debugging client. Tried: localhost:9003 (fallback through
xdebug.client_host/xdebug.client_port) :-(

Xdebug: [Step Debug] Could not connect to debugging client. Tried: localhost:9003 (fallback through
xdebug.client_host/xdebug.client_port) :-(

Xdebug: [Step Debug] Could not connect to debugging client. Tried: localhost:9003 (fallback through
xdebug.client_host/xdebug.client_port) :-(

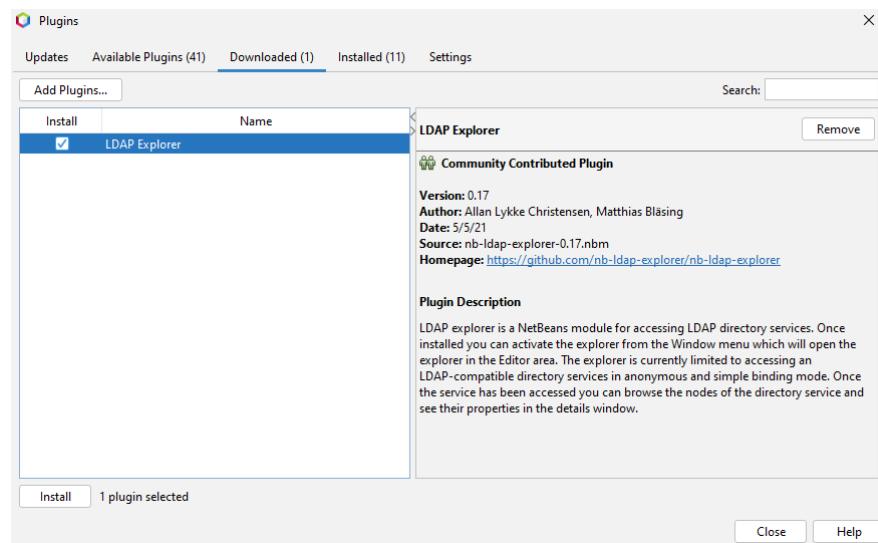
miadmin@rsp-used:/var/www/html/204DWESLoginLogoffTema5/doc$
```

5.8 CSS / JS / AJAX / XML / JSON

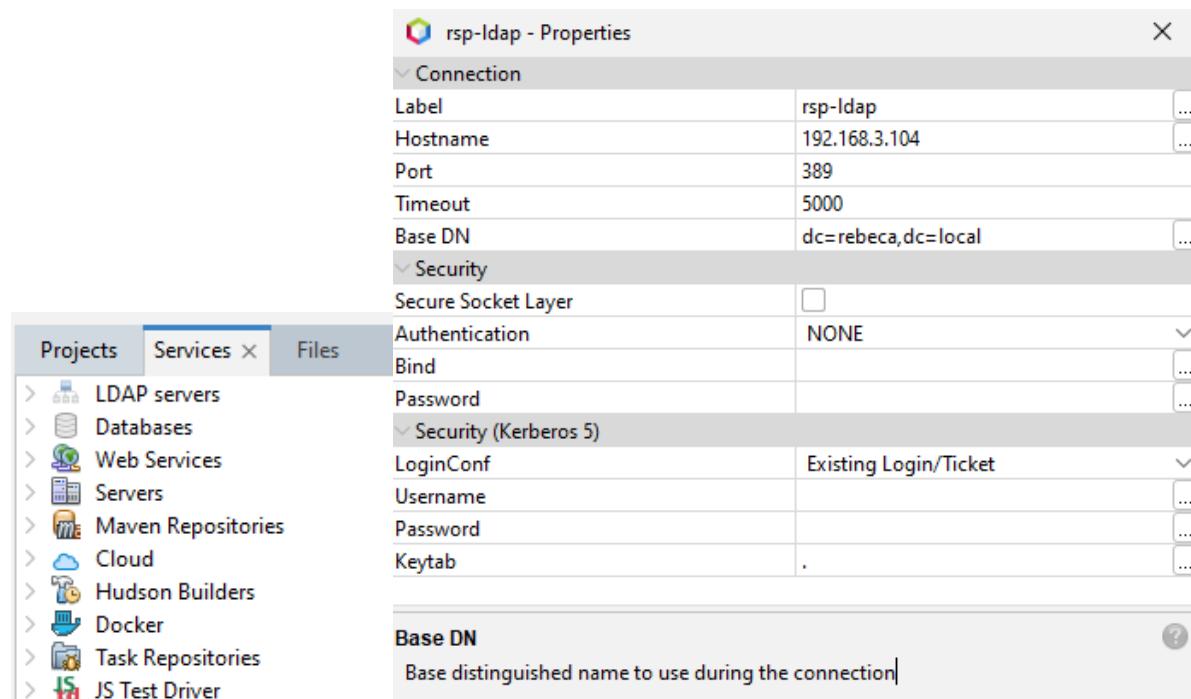
?????? No la hemos realizado en clase ???????

5.9 LDAP

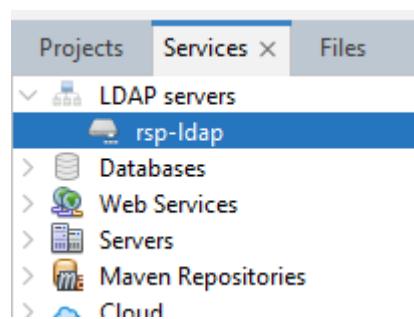
Instalaremos en nuestro NetBeans un lector del servicio LDAP, para ello nos descargaremos el plugin, vamos a la pestaña “plugins” y lo añadimos



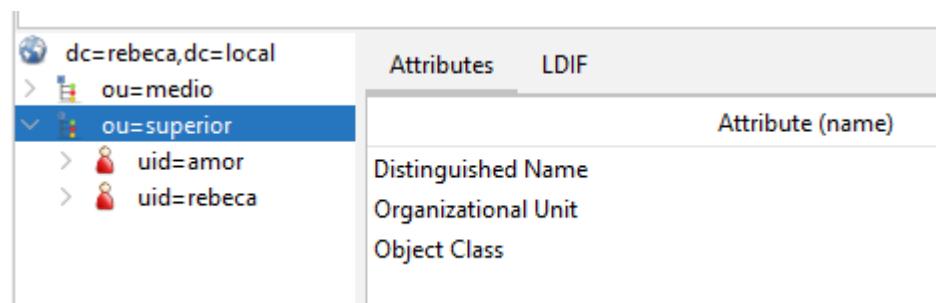
Aparecerá en la ventada de Services de la izquierda con las siguientes propiedades:



Cuando todo este correcto, aparecerá una nueva conexión



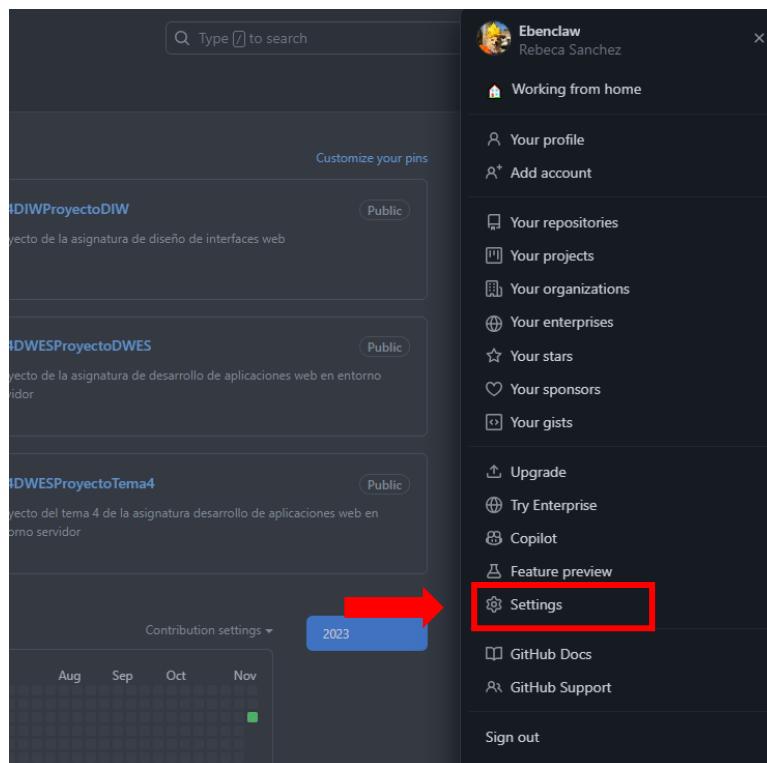
Hacemos click derecho “Establish New Connection” y se encontrara nuestro servidor LDAP con las unidades organizativas y los usuarios que tenemos en nuestra máquina virtual



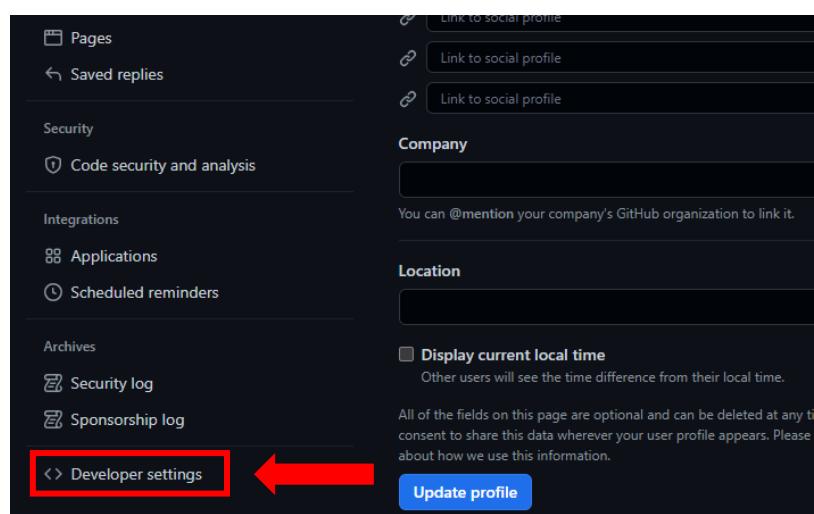
c. GITHUB – INTERNET

1. Creación de token desde GitHub

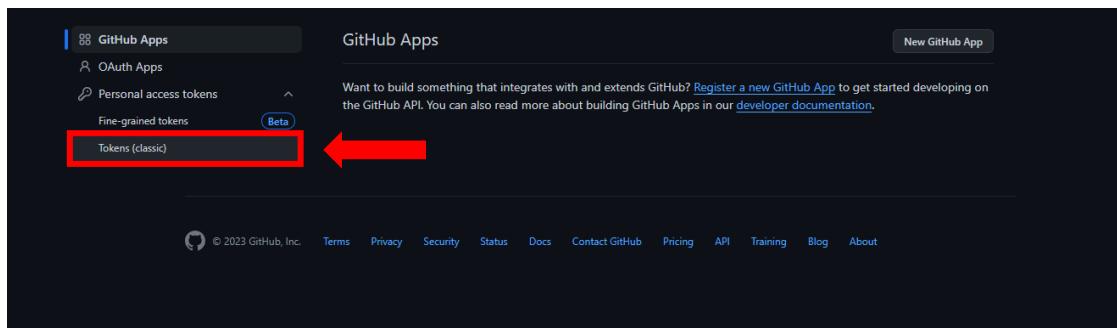
Para comenzar a trabajar en GitHub, necesitamos tener una cuenta registrada y una vez la tengamos, comenzaremos creando un token. Un token es una contraseña de autenticación usada para manejar la API de GitHub, para crearlo nos dirigiremos a la derecha de la parte superior de la página donde aparece nuestro perfil y cuando le hagamos clic, se desplegará un menú con distintas opciones. Nos dirigiremos a la opción de “settings”



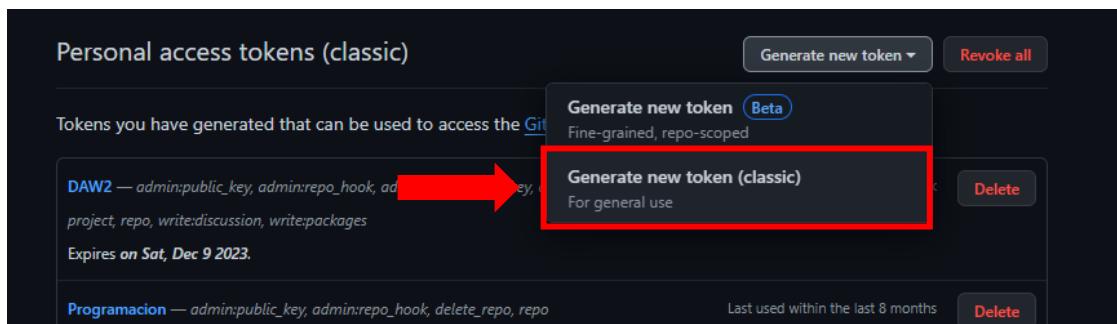
En la siguiente ventana, nos desplazaremos hasta la parte inferior y seleccionaremos la opción “developer settings” del menu de la izquierda



Nos dirigimos a “Personal access tokens>Tokens (clasic)”

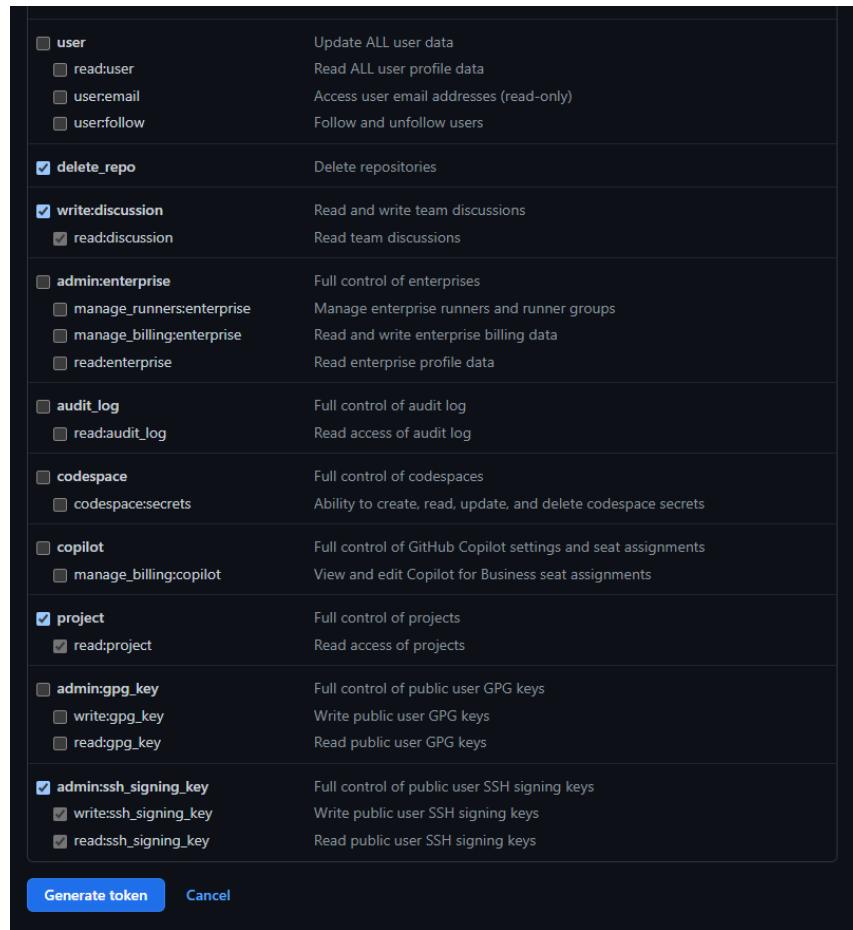


A continuación, nos aparecerá la lista de nuestros tokens. Para crear uno nuevo, desplegamos el botón “Generate new token” y seleccionamos la opción de “Generate new token (classic)”



Aparecerá una ventada en la elegiremos el nombre del token, el tiempo de expiración y las especificaciones. Seguiremos las siguientes especificaciones:

The screenshot shows the 'Select scopes' dialog box. It lists various OAuth scopes with checkboxes. Some scopes have sub-options. The checked scopes are: repo (repo, repos:status, repos:deployment, public_repo, repos:invite, security_events), workflow (workflow), write:packages (write:packages, read:packages), delete:packages (delete:packages), admin:org (admin:org, write:org, read:org, manage:runners:org), admin:public_key (admin:public_key, write:public_key, read:public_key), admin:repo_hook (admin:repo_hook, writer:repo_hook, read:repo_hook), admin:org_hook (admin:org_hook), gist (gist), notifications (notifications), user (user), and read:user (read:user). Descriptions for each scope are provided to the right.



Una vez seleccionadas las características, generamos el token y nos aparecerá la ventana anterior con la clave del nuevo token. Esta clave será la contraseña que nos pedirán siempre que intentemos realizar cambios en el repositorio remoto, por lo cual, es muy importante copiarla en un fichero y guardarla para no perderla.

Personal access tokens (classic)

Tokens you have generated that can be used to access the [GitHub API](#).

Make sure to copy your personal access token now. You won't be able to see it again!

✓ ghp_QSFM6ZL31nupYYJDjtuUtk7xySQ410JB4eK		Delete
---	--	---------------

Generate new token ▾ Revoke all

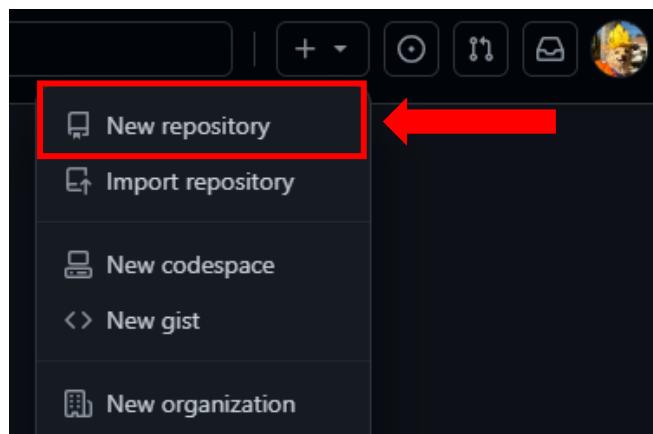
2. Comandos básicos de Git

A continuacion, se muestra una lista con los comandos básicos que podemos usar en git bash o en NetBeans:

- git init creará un nuevo repositorio local GIT.
- git clone se usa para clonar un repositorio en local
- git add se usa para agregar archivos al área de preparación o staging
- git commit creará una instantánea de los cambios y la guardará en el directorio git
- git status muestra la lista de los archivos que se han cambiado junto con los archivos que están por ser preparados o confirmados
- git push se usa para enviar confirmaciones locales a las ramas del repositorio remoto.
- git checkout cambia el área de trabajo a la rama especificada.
- git branch se usa para listar, crear o borrar ramas
- git pull fusiona todos los cambios que se han hecho en el repositorio remoto con el directorio de trabajo local
- git merge se usa para fusionar una rama con otra rama activa
- git diff se usa para hacer una lista de conflictos
- git log se usa para ver el historial del repositorio listando ciertos detalles de la confirmación
- git fetch le permite al usuario buscar todos los objetos de un repositorio remoto que actualmente no se encuentran en el directorio de trabajo local, es decir, muestra los cambios existentes entre el repositorio remoto y el local

3. Creación de un repositorio desde NetBeans

En primer lugar, crearemos un repositorio vacío en GitHub desde nuestro navegador con el nombre que va a tener nuestro proyecto. Para ello hacemos click en el icono de + en la parte superior derecha de la página y seleccionamos “New repository” donde elegiremos el nombre del proyecto y una breve descripción de este. También podemos especificar la privacidad de este, si queremos que sea público o privado, en este caso elegiremos público. Después se nos generara una URL que tendremos que asocia a nuestro repositorio local.



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk (*).

Owner *	Repository name *
Ebendlaw	/ 204DAWProyectoDAW2
<input checked="" type="checkbox"/> 204DAWProyectoDAW2 is available.	

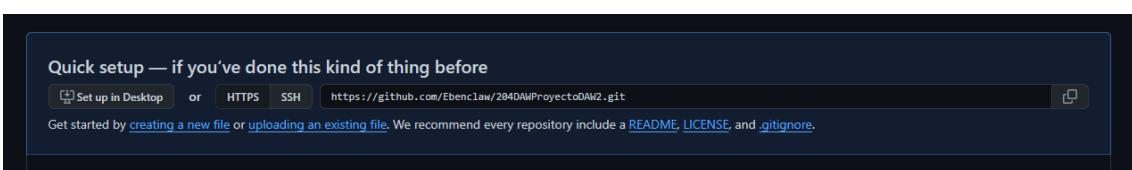
Great repository names are short and memorable. Need inspiration? How about [effective-palm-tree](#) ?

Description (optional)

Proyecto de desarrollo de aplicaciones web

Public
 Anyone on the internet can see this repository. You choose who can commit.

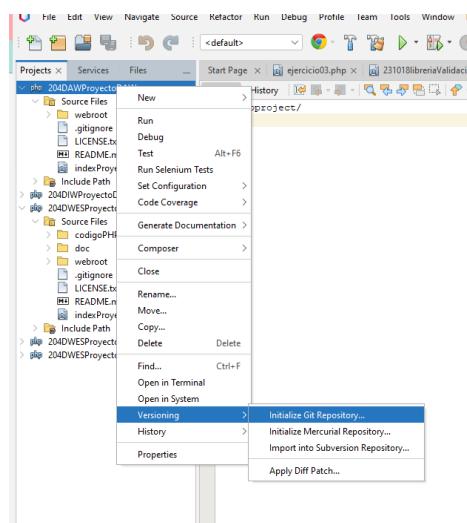
Private
 You choose who can see and commit to this repository.



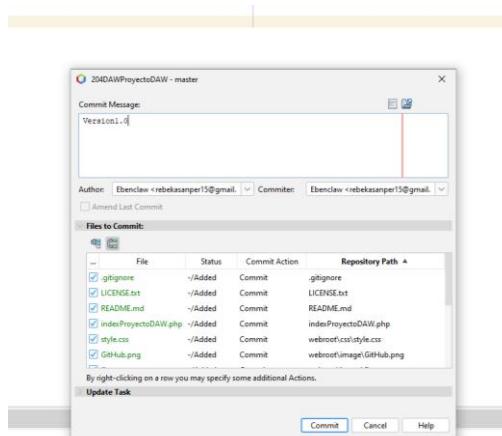
A continuación, desde NetBeans o en la carpeta de local, añadimos un README.md y una LICENCE.txt en cada proyecto y un “.gitignore” con el contenido “/nbproject/” para que no suba esa carpeta (que se genera automáticamente cuando creas un proyecto en NetBeans) al repositorio remoto

codigoPHP	24/11/2023 10:03	Carpeta de archivos
doc	23/11/2023 10:17	Carpeta de archivos
error	24/11/2023 9:59	Carpeta de archivos
nbproject	11/10/2023 9:33	Carpeta de archivos
webroot	11/10/2023 9:33	Carpeta de archivos
.gitignore	14/11/2023 9:07	Documento de te... 1 KB
.htaccess	24/11/2023 9:59	Archivo HTACCESS 1 KB
indexProyectoDWES.php	24/11/2023 10:03	Archivo PHP 3 KB
LICENSE.txt	14/11/2023 8:45	Documento de te... 35 KB
README.md	23/11/2023 10:17	Archivo MD 1 KB

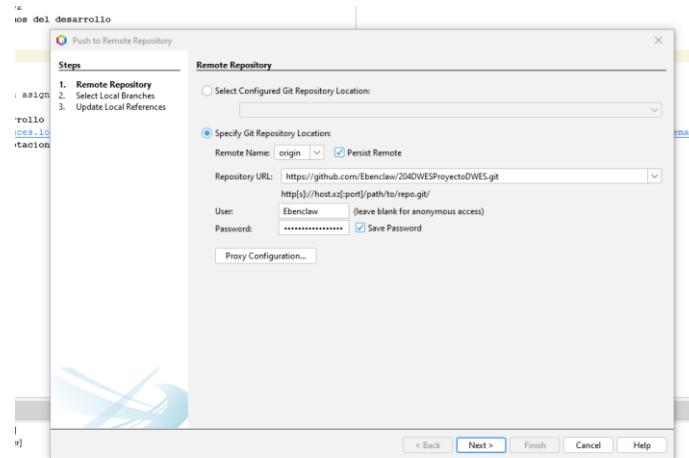
Desde NetBeans, abrimos el proyecto que deseamos versionar y le hacemos click derecho>"versioning">"Initialize Git Repository"



Hacemos el primer commit en master con la Version1.0 con click derecho>Git>Commit. Podemos escribir un pequeño texto en el que aparezca el contenido de la primera versión.



Una vez hecho el commit, hacemos un push al repositorio de GitHub indicando la URL del proyecto que habíamos creado vacío, el usuario y la password que será nuestro token



Comprobamos que el proyecto se ha subido correctamente a GitHub

File	Version	Updated
codigoPHP	Version1.0	17 minutes ago
doc	Version1.0	17 minutes ago
webroot	Version1.0	17 minutes ago
.gitignore	Version1.0	17 minutes ago
LICENSE.txt	Version1.0	17 minutes ago
README.md	Version1.0	17 minutes ago
indexProyectoDWES.php	Version1.0	17 minutes ago

Creamos una rama a partir de la versión 1.0 de la master en local llamada "developer204" con git>branch/tag>create branch. Esta rama la usaremos para publicar todos los commit que realicemos en el desarrollo. Para movernos entre ramas usaremos la opción checkout

The screenshot shows the 'Create Branch' dialog box in SourceTree. The 'Branch Name' field contains 'developer'. The 'Revision' dropdown is set to 'master'. Below it, the commit details are shown: Commit ID 'master (18664ec994)', Author 'Ebenclaw <rebekasanper15@gmail.com>', Date '17 nov 2023, 8:47:48', and Message 'Version1.0'. A checkbox 'Checkout Created Branch' is checked. At the bottom are 'Create', 'Cancel', and 'Help' buttons.

Below the dialog, the SourceTree file tree shows two repositories:

- > 204DWESProyectoDWES [developer]
- > 204DWESProyectoTema3 [developer]
 - > Branches
 - > Local
 - > developer - 18664ec994
 - > master (in sync with "origin/master") - 18664ec994
 - > Remote
 - > Tags
 - > Stashes
 - > Remotes

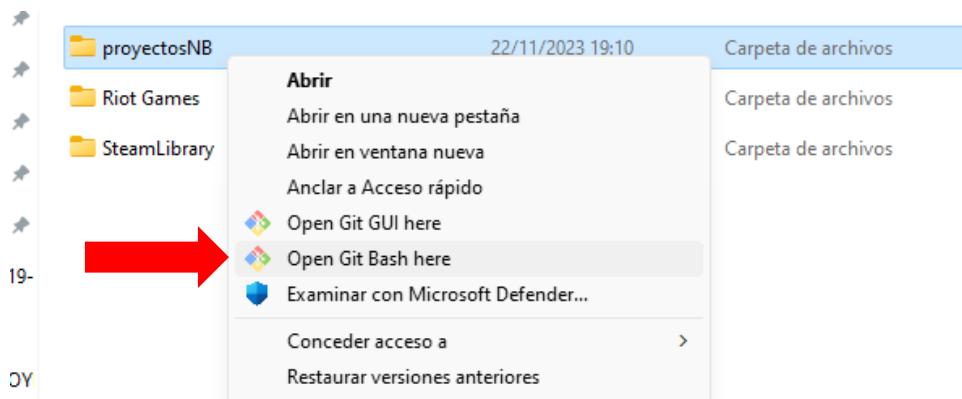
Dentro de esa rama realizamos cambios, cuando el proyecto este actualizado hacemos commit y después hacemos un push para actualizar el repositorio remoto con la rama nueva

The screenshot shows a GitHub repository interface. On the left, a sidebar displays the current branch 'master' and other repository statistics: '2 Branches' and '1 Tags'. A 'Switch branches/tags' dropdown menu is open, showing a search bar and a list of branches. The branch 'developer204' is highlighted with a red rectangle. Other branches listed include 'master' and 'default'. Below the sidebar, the main repository area shows a list of commits:

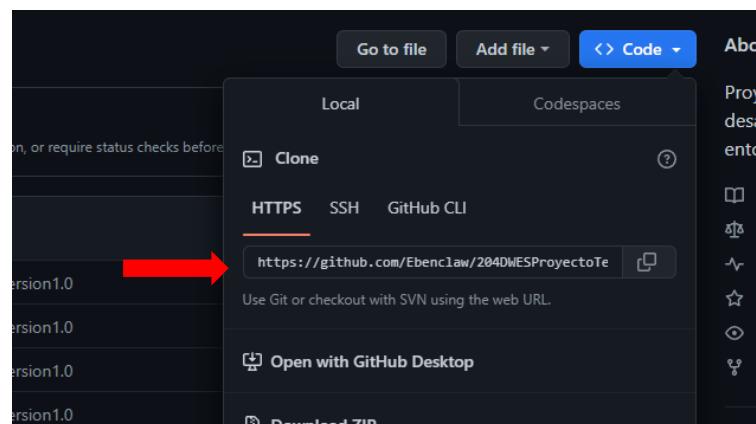
Commit Message	Date
Cambios en el incide de ejercicios	last month
Cambios en el incide de ejercicios	last month
Cambios en el incide de ejercicios	last month
Version1.0	last month
cambios en el README y en el .htaccess	3 weeks ago
Version1.0	last month
cambios en el README y en el .htaccess	3 weeks ago
link al proyecto del tema 5	3 weeks ago

4. Clonar un repositorio

Si queremos clonar un repositorio remoto a nuestro repositorio local, en primer lugar, haremos click derecho sobre la carpeta padre en la que queremos que esté situado nuestro proyecto y seleccionamos “open Git Bash here”.

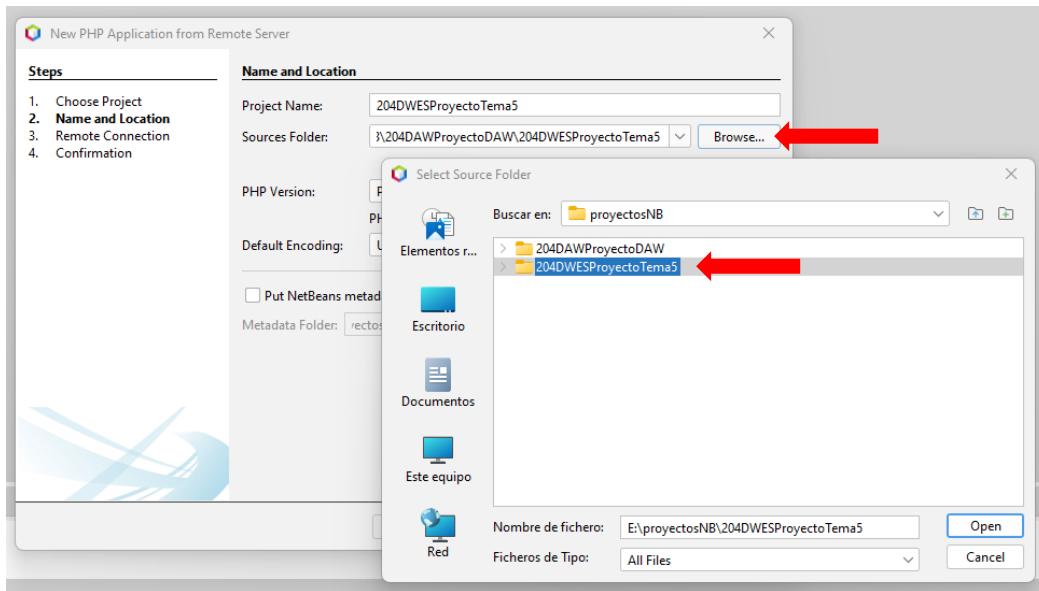


A continuación, se abrirá el cmd de Windows y escribiremos “git clone <<url de nuestro repositorio remoto>>”. La url de nuestro repositorio ponemos copiarla desde la barra de navegación de nuestro navegador o desde la ventana “code” del proyecto en GitHub.

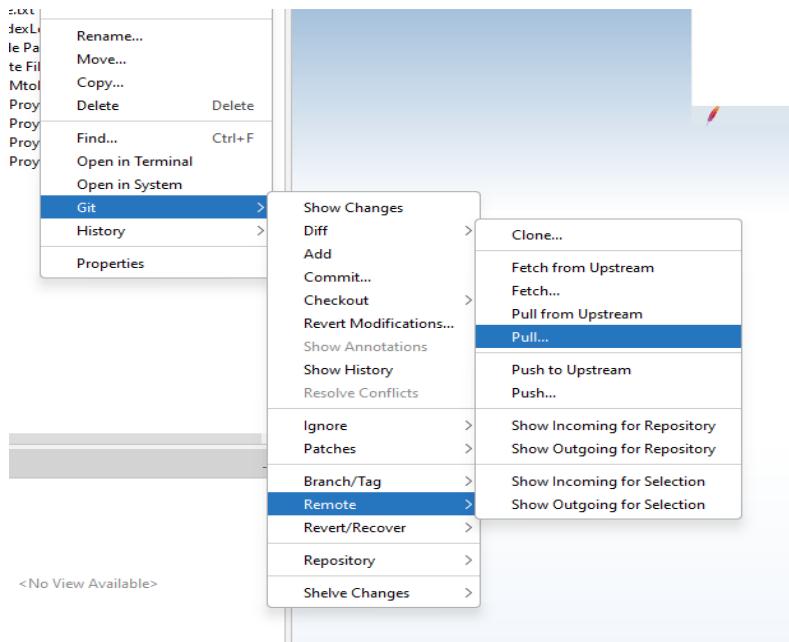


```
rebek@Pc_de_Rebeca MINGW64 /e/proyectosNB
$ git clone https://github.com/Ebenclaw/204DWESProyectoTema5.git
Cloning into '204DWESProyectoTema5'...
remote: Enumerating objects: 45, done.
remote: Counting objects: 100% (45/45), done.
remote: Compressing objects: 100% (31/31), done.
remote: Total 45 (delta 12), reused 42 (delta 9), pack-reused 0
Receiving objects: 100% (45/45), 2.48 MiB | 1.65 MiB/s, done.
Resolving deltas: 100% (12/12), done.
```

Una vez clonado, iremos a NetBeans y crearemos un nuevo proyecto como hasta ahora, pero seleccionaremos el repositorio clonado cuando nos pida la localización del proyecto



Con el proyecto en NetBeans abierto, haremos click derecho “versioning>remote>fetch” para comparar los commit que hay en el repositorio remoto. Para descargar las ramas remotas en las ramas locales en las que queremos hacer cambios, usamos el comando `git pull` sobre la rama a desarrollar



5. Crear una versión estable en la rama master

Cuando hemos alcanzado una versión estable de nuestra aplicación, es el momento de pasarla al entorno de explotación, pero no subiremos un commit cualquiera, sino aquel que ha sido mergeado a la rama master e identificado como una versión estable. Lo realizaremos de la siguiente manera:

Desde git bash cambiamos a la rama master con “git checkout master” (nos colocamos en la rama destino en la que se va a mergear la versión) y realizamos el comando “git merge –no-ff developer204” (el comando –no-ff significa que cuando realice el merge a su vez haga un commit)

```
daw2@IS32WX04 MINGW64 /d/proyectosnb/204DWESProyectoDWES (developer204)
$ git branch
* developer204
  master

daw2@IS32WX04 MINGW64 /d/proyectosnb/204DWESProyectoDWES (developer204)
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.

daw2@IS32WX04 MINGW64 /d/proyectosnb/204DWESProyectoDWES (master)
$ git merge --no-ff developer204
Merge made by the 'ort' strategy.
 .htaccess      |  5 +++++
 README.md      |  3 ++
 codigоТРН/indexTema1.php |  6 +++---
 codigоТРН/indexTema2.php |  7 +++++-
 doc/EstudioTema2.pdf | Bin 930044 -> 1671719 bytes
 indexProyectoDWES.php | 18 ++++++-----+---+
 webroot/css/style.css | 24 ++++++-----+---+
 7 files changed, 46 insertions(+), 17 deletions(-)
 create mode 100644 .htaccess
```

A continuación, cambiamos a la rama developer con un checkout y seguido de esto hacemos un “git merge –ff master” para colocar el cursor sobre la versión estable de la rama master. Esta es la forma más idónea de mover el puntero gracias al comando –ff que anula el autocommit cuando hacemos un merge. Podemos escribir “git log –graph --all” para mostrar una línea temporal muy gráfica en la que se distinguen las ramas y todas las versiones

```
daw2@IS32WX04 MINGW64 /d/proyectosnb/204DWESProyectoDWES (master)
$ git branch
  developer204
* master

daw2@IS32WX04 MINGW64 /d/proyectosnb/204DWESProyectoDWES (master)
$ git checkout developer204
Switched to branch 'developer204'
Your branch is ahead of 'origin/developer' by 2 commits.
 (use "git push" to publish your local commits)

daw2@IS32WX04 MINGW64 /d/proyectosnb/204DWESProyectoDWES (developer204)
$ git merge --ff master
Updating ee0a19f..8c26ab8
Fast-forward

daw2@IS32WX04 MINGW64 /d/proyectosnb/204DWESProyectoDWES (developer204)
$ git log --graph --all
*   commit 8c26ab8dd6c9a1d7b4394dec736c0618a47b2bea (HEAD -> developer204, master)
|\ Merge: 1dcf8c9 ee0a19f
| | Author: Ebenclaw <rebekasanper15@gmail.com>
| | Date:   Thu Nov 23 10:17:50 2023 +0100
| |
| | Merge branch 'developer204'
```

Para terminar, subiremos los cambios al repositorio de GitHub con “git push origin -u --all”

```
* commit f418c80450e71bf8caa609abc0169288d342e7dd
  Author: Ebenclaw <rebekasanper15@gmail.com>
  Date:   Fri Nov 17 09:09:40 2023 +0100

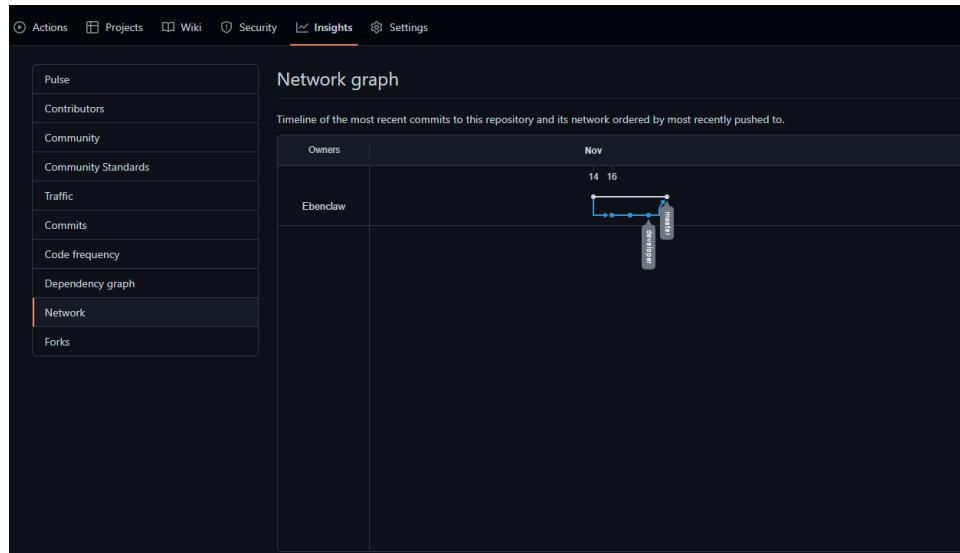
    Primer commit en developer

* commit 1dcf8c9752f6780cd6a3182fcfef77daf86553bd (origin/master)
  Author: Ebenclaw <rebekasanper15@gmail.com>
  Date:   Fri Nov 17 08:38:35 2023 +0100

    Version1.0

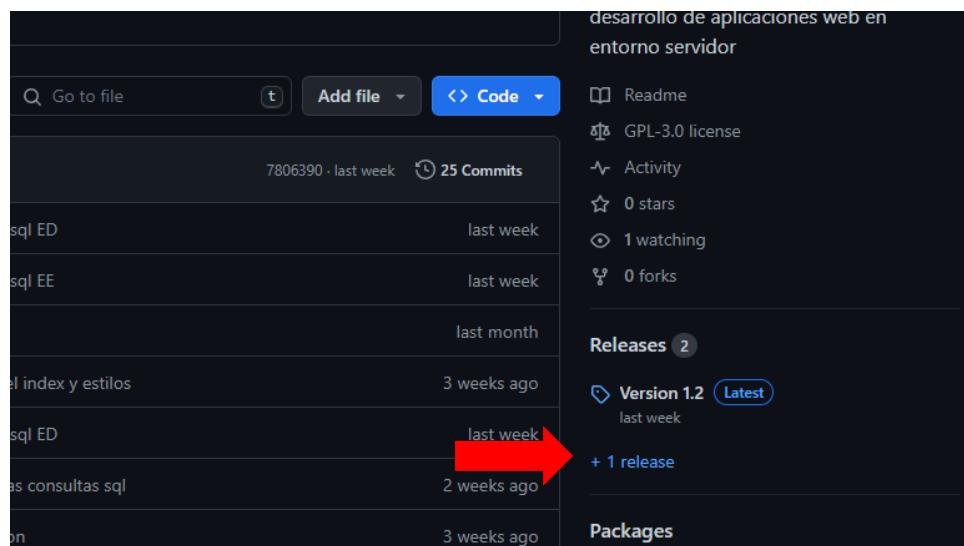
daw2@IS32WX04 MINGW64 /d/proyectosnb/204DWESProyectoDWES (developer204)
$ git push origin -u --all
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 227 bytes | 227.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Ebenclaw/204DWESProyectoDWES.git
  ee0a19f..8c26ab8  developer204 -> developer204
  1dcf8c9..8c26ab8  master -> master
branch 'developer204' set up to track 'origin/developer204'.
branch 'master' set up to track 'origin/master'.
```

Comprobamos que se ha subido a GitHub correctamente, para ello existe la herramienta “Network” en las estadísticas de los proyectos que nos muestra un diagrama de flujo muy visual

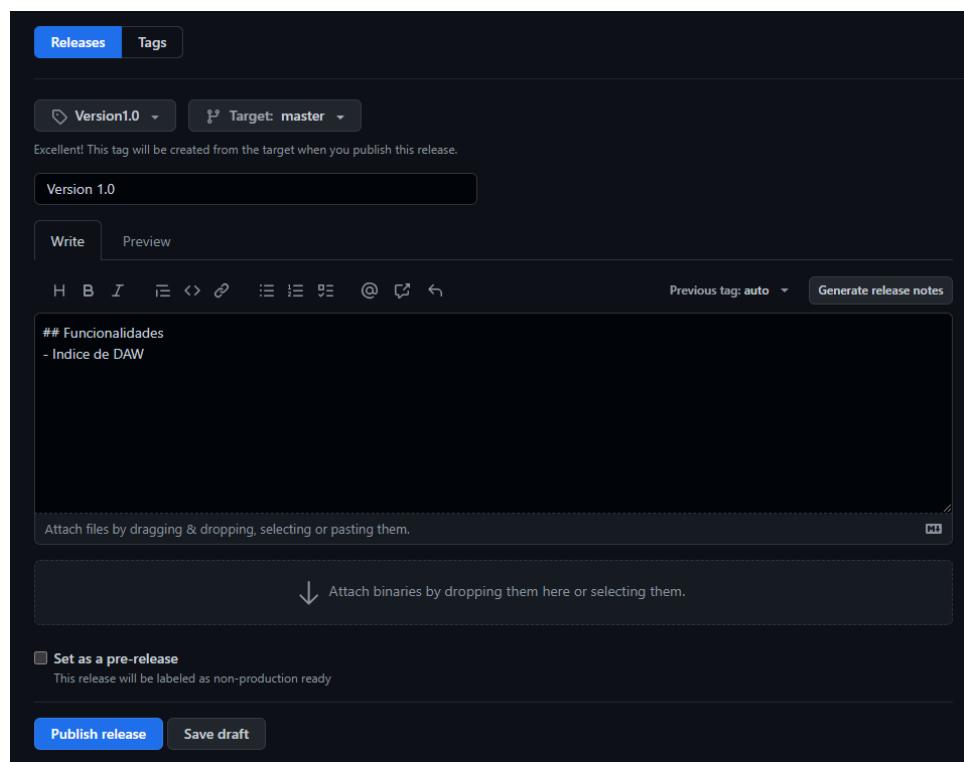


6. Crear una release

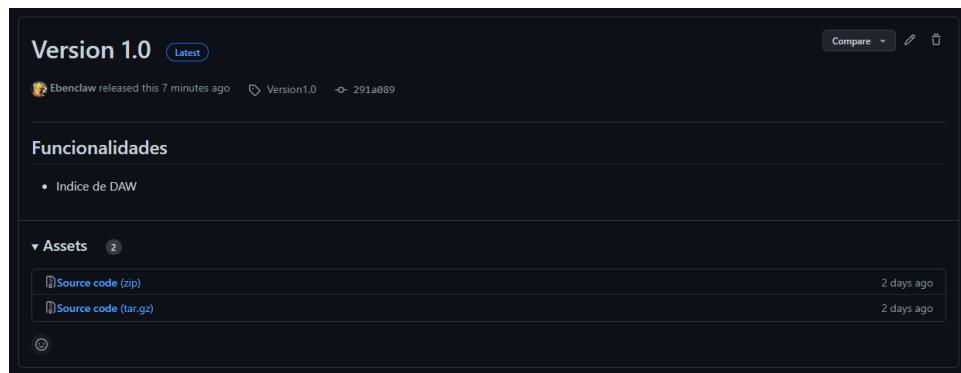
Para crear una release de una versión estable de nuestra app, iremos a nuestro proyecto de GitHub a la parte de la derecha donde aparece el título Release y “create a new release”



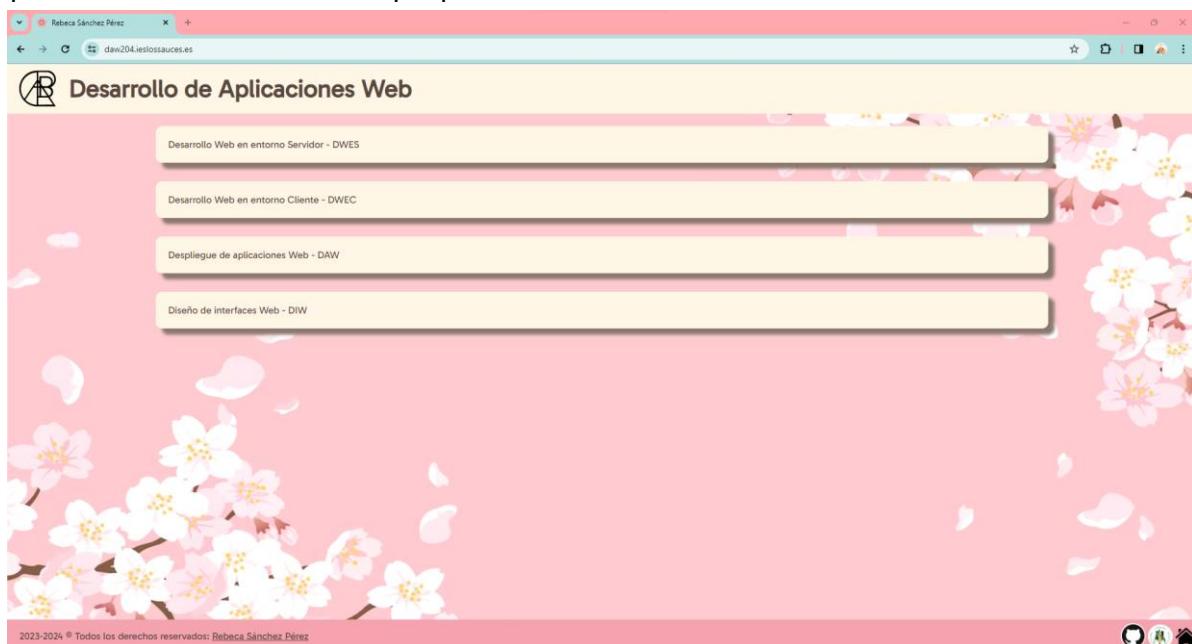
En la siguiente ventana seleccionamos una Tag (si es nuestra primera release, tendremos que crearla), elegiremos un nombre para nuestra release (en nuestro caso la llamaremos como el nombre de la versión) y detallaremos una breve descripción a cerca de los cambios que se ha llevado a cabo en esa versión del código



Cuando este creada, se generarán 2 ficheros, uno con extensión zip y otro tar.gz. Descargamos el zip y lo descomprimimos en nuestro repositorio local. Esta carpeta corresponde a las líneas de código del último commit que ha sido mergeado en la rama master. Esta carpeta es la que desplegaremos en el entorno de explotación mediante filezilla



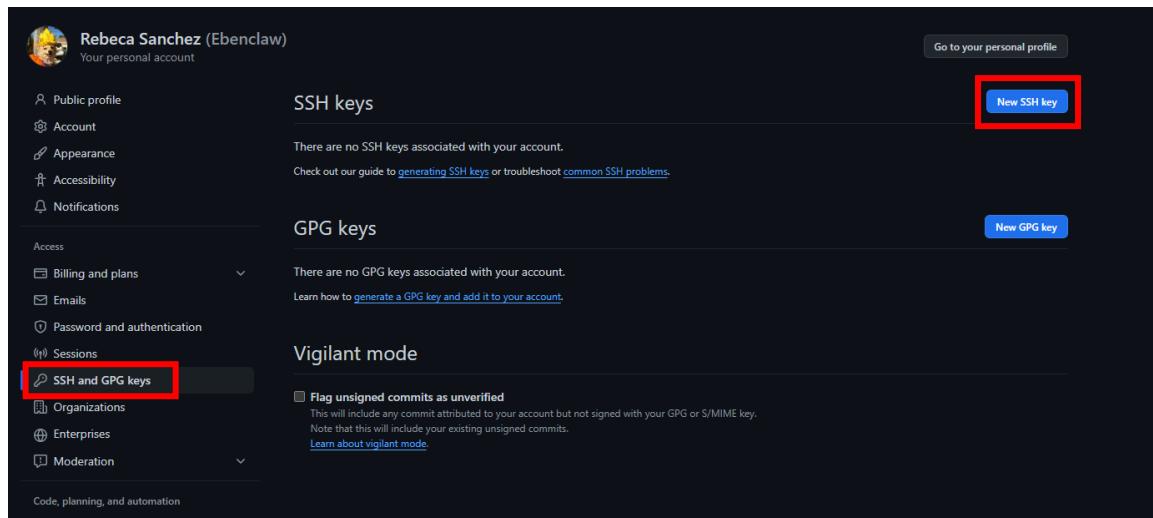
Para comprobar que la release se ha subido correctamente al entorno de explotación, escribiríamos en el navegador "daw204.ieslossauces.es" y desde ese dominio comprobaríás que esa es la versión estable que pretendías subir



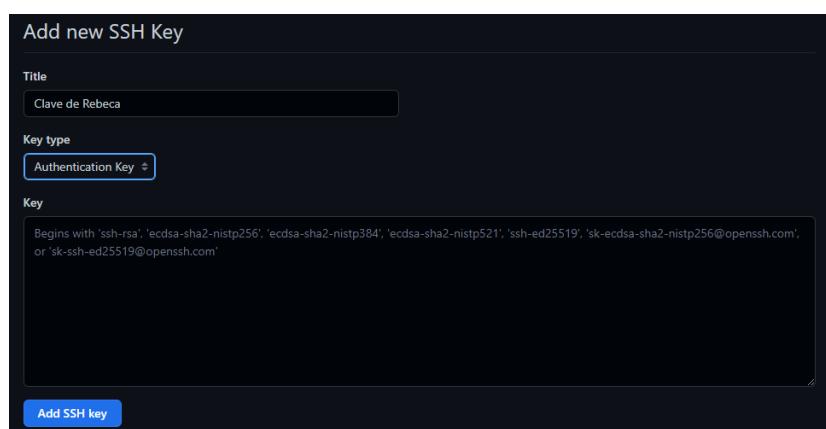
7. Conexión con SSH

7.1 Generar clave SSH

La conexión SSH nos permite vincular nuestros repositorios de manera segura utilizando SSH mediante una clave pública y una clave privada. Para generarla, abriremos el menú de “Settings”, la opción “SSH and GPG keys” y seleccionamos “New SSH key”



Nombraremos a nuestra clave con el título que queramos y seleccionaremos el tipo de clave de autenticación. En el recuadro de abajo tenemos que escribir nuestra clave privada pero como en la máquina anfitriona no tenemos, la generaremos con el comando `“ssh-keygen -t ed25519 -C tucorreoelectronico@hotmail.com”` desde el cmd de Windows y se generarán 2 claves (pública y privada) en el directorio que te especifica la sinopsis.



```
daw2@IS32WX04 MINGW64 ~
$ ssh-keygen -t ed25519 -C "rebeca.sanper.1@educa.jcyl.com"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/c/Users/daw2/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /c/Users/daw2/.ssh/id_ed25519
Your public key has been saved in /c/Users/daw2/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:t0u97Sn84uRernAKixJ2m5nHPRvyjRda9ul7g3XnS5I rebeca.sanper.1@educa.jcyl.com
The key's randomart image is:
+--[ED25519 256]--+
| |
| |
| |
| |
| |
| |
| |
| |
| |
+---[SHA256]---
```

Abriremos el fichero que contiene la clave pública y la copiaremos en el recuadro de antes de GitHub

id_ed25519	01/12/2023 9:20	Archivo	1 KB
id_ed25519.pub	01/12/2023 9:20	Documento de Mi...	1 KB
known_hosts	26/10/2023 8:56	Archivo	1 KB

Add new SSH Key

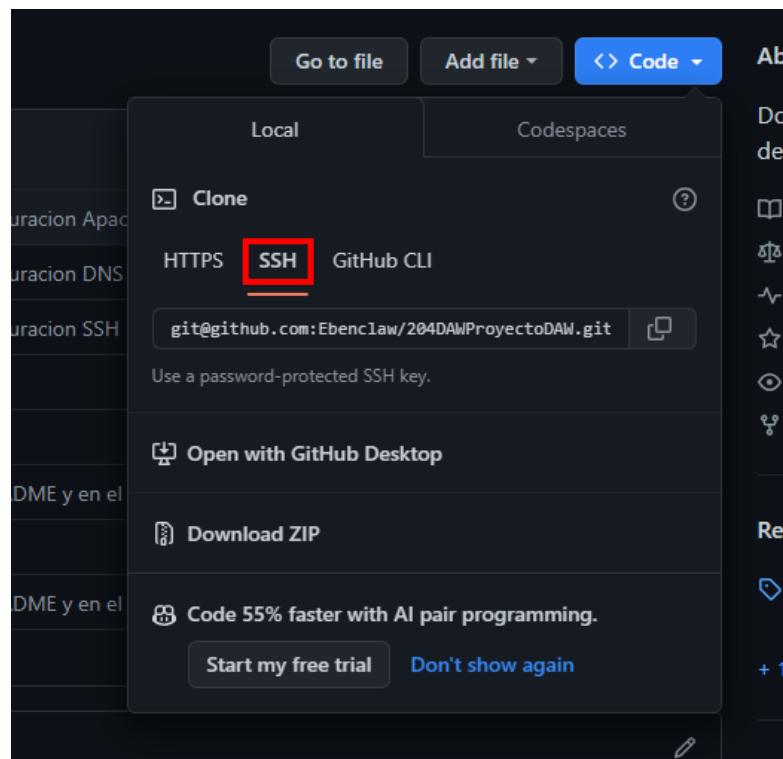
Title

Key type

Key

7.2 Conexión SSH desde NetBeans

Copiamos el enlace SSH de nuestro repositorio remoto



A la hora de establecer una conexión en NetBeans del repositorio local con el remoto, elegiremos la opción “Specify Git Repository Location” y en la URL del directorio pegamos la del paso anterior de GitHub. Usaremos la clave privada en “Private Key Files” y se establecerá la conexión SSH

