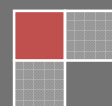


2017/  
2018

# CUADERNO DE EJERCICIOS DESPLIEGUE DE APLICACIONES WEB

Pablo Cidón Barrio  
IES Los Sauces  
2017/2018



## Contenido

UD 1 Implantación de Aplicaciones Web .....	6
Actividad 1.- Máquina anfitriona .....	6
A) Documentación.....	6
B) Habilitar respuesta ping en el Firewall de Windows.....	6
C) Comprobar la configuración .....	7
Actividad 2.- Máquina virtual en Linux.....	8
A) Documenta la instalación y configuración de una máquina virtual teniendo en cuenta las siguientes especificaciones.....	8
B) Comprobar la configuración .....	8
Actividad 3.- Servidor de aplicaciones .....	9
A) Documenta la instalación y configuración de un servidor de Aplicaciones .....	9
B) Comprobar la configuración .....	11
C) Puertos y conexiones .....	13
Actividad 4.- Estructura del entorno de desarrollo.....	13
Actividad 5.- Mantenimiento de los servidores Entorno de Desarrollo y Explotación. ....	14
Actividad 6.- Explica diferentes IDE .....	14
Notepad ++ .....	14
Netbeans .....	15
PhpStorm/WebStorm.....	17
Actividad 7.- Proceso de despliegue de una aplicación de Entorno de Desarrollo a Explotación.	18
UD 2 Sistema de Control de Versiones .....	20
Actividad 1.- Instalación de Gitlab en una máquina virtual en Ubuntu Server .....	20
Actividad 2.- Instalación de cliente git en Windows, .....	21
Actividad 3.- Trabajando con Git .....	26
Actividad 4.- Trabajando con ramas .....	28
Actividad 5.- Ramas y fusiones en local (FAST-Forward).....	28
a) Borrar el repositorio remoto DAW1718 .....	28
b) Borrar el repositorio local DAW1718 .....	29
c) Crear de nuevo el proyecto HOLAMUNDO, con un commit con el archivo README.md que contiene el nombre de la asignatura. ....	29
d) Clonación del proyecto en local.....	31

e) Crear una página web, index.html, con el mensaje “Hola Mundo” sin estilos. Una vez terminada, súbela al repositorio remoto. ....	31
d) Añade la etiqueta v1.0 al commit anterior.....	33
e) Se piden crear dos diseños con estilos(css/estilo.css) diferentes del index.html, para ello realizamos dos ramas en local, develop1 y develop2. ....	33
d) Una vez que el profesor decida que diseño es el más adecuado, se realizará la fusión en la master.....	33
e) Se sube al servidor la rama master. ....	33
d) Se puede borrar las ramas develop1 y develop2.....	33
g) Se etiqueta el nuevo commit, como v1.1. ....	33
UD 3 Servidor Web Apache .....	33
Módulo USERDIR.....	33
Creación de Alias.....	34
Control de Acceso por IP .....	35
Autenticación y autorización basic. ....	35
Control de acceso desde .htaccess .....	37
Autenticación y autorización digest.....	37
Autenticación y autorización digest por grupos .....	39
Módulo MOD-STATUS:.....	39
Desde el propio servidor: .....	40
Desde el navegador:.....	40
Módulo INFO: .....	41
Configuración de logs en Apache:.....	41
Virtualización basada en puertos: .....	41
Sitios virtuales basados en nombre .....	43
Generación de Certificados .....	45
Redireccionar de http a https:.....	48
UD 4 Servidor de Nombres de Dominio (DNS) .....	49
Actividad 01 .....	49
Instalación del Servidor DNS.....	49
Configurar reenviadores.....	50
Agregar a un dominio.....	50
Configuración de un servidor DNS master: .....	50

Creación de Alias:.....	52
Configuración de la zona inversa: .....	52
Resolución del nombre del dominio: .....	53
Actividad 02: .....	53
Actividad 03: .....	53
Actividad 04: .....	55
UD 5 FTP .....	55
Instalación y configuración de ProFTP: .....	55
Enjaular usuarios:.....	56
Asignación de los permisos: .....	57
Usuarios que tienen acceso al sistema: .....	57
Configuración del usuario anonymous ftp: .....	58
Control de acceso: .....	59
Limitar el acceso a determinados usuarios: .....	61
Usuarios y dominios virtuales.....	61
Control de Cuotas: .....	63
Recargar la configuración del servidor:.....	64
Documentación de las cuotas del disco. ....	65
Acceso seguro mediante TLS .....	65
UD 7 Documentación .....	65
NetBeans. ....	65
Línea de comandos.....	66
Doxygen. ....	67
UD 8 Servicio de directorio OpenLDAP. ....	68
Instalación y configuración de OpenLDAP.....	68
Configuración de OpenLDAP:.....	69
Configuración de la autenticación de clientes:.....	71
Configurar el demonio SLAPD:.....	71
Crear la estructura de directorio:.....	72
Añadir usuarios y grupos:.....	73
Comprobación de que está todo correcto: .....	74
Interfaz gráfica web: phpLDAPadmin. ....	75

Instalación de phpLDAPadmin:.....	75
Configuración de phpLDAPadmin:.....	75
Creación de usuarios:.....	76
Autenticación LDAP en Apache: .....	78
Usuarios:.....	79
Grupos: .....	80

## UD 1 Implantación de Aplicaciones Web

### Actividad 1.- Máquina anfitriona

#### A) Documentación

<b>Máquina Anfitriona</b>		
<b>Sistema Operativo</b>	Windows 10 64 bits	
<b>Memoria RAM</b>	4 GB	
<b>Particiones</b>	Sistema C:	Datos E:
<b>Nombre de la máquina</b>	IS32WX11	
<b>Usuarios y contraseñas</b>	admin	admin
	daw2	daw2
<b>Configuración de red</b>	<b>Puerta de Enlace:</b> 192.168.3.1	<b>DNS:</b> 192.168.20.20
<b>Software usado para el desarrollo del ciclo</b>		
Notepad++	Netbeans	FileZilla

#### B) Habilitar respuesta ping en el Firewall de Windows

Firewall de Windows con seguridad avanzada

Archivo Acción Ver Ayuda



Firewall de Windows con seguridad avanzada		Reglas de entrada tipo local			
<ul style="list-style-type: none"> <li>Reglas de entrada</li> <li>Reglas de salida</li> <li>Reglas de seguridad de con...</li> <li>Supervisión</li> </ul>		Nombre	Grupo	Perfil	Habilitado
		Enrutamiento y acceso remoto (L2TP de ...	Enrutamiento y acceso rem...	Todo	No
		Enrutamiento y acceso remoto (PPTP de ...	Enrutamiento y acceso rem...	Todo	No
		Escritorio remoto - instantánea (TCP de e...	Escritorio remoto	Todo	No
		Escritorio remoto - Modo usuario (TCP d...	Escritorio remoto	Todo	No
		Escritorio remoto - Modo usuario (UDP d...	Escritorio remoto	Todo	No
		✓ Eventos UPnP de Transmitir en dispositiv...	Funcionalidad de transmitir ...	Público	Sí
		✓ Firefox (C:\Program Files\Mozilla Firefox)		Privado	Sí
		✓ Firefox (C:\Program Files\Mozilla Firefox)		Privado	Sí
		✓ Funcionalidad de transmitir en dispositiv...	Funcionalidad de transmitir ...	Priva...	Sí
		✓ Funcionalidad de transmitir en dispositiv...	Funcionalidad de transmitir ...	Priva...	Sí
		✓ Google Chrome (tráfico mDNS entrante)	Google Chrome	Todo	Sí
		✓ Groove Música	Groove Música	Dom...	Sí
		✓ Groove Música	Groove Música	Dom...	Sí
		Grupo Hogar entrante	Grupo Hogar	Privado	No
		Grupo Hogar entrante (PNRP)	Grupo Hogar	Privado	No
		✓ ICMPv4 Abierto		Todo	Sí

## C) Comprobar la configuración

```
C:\Users\PC>ipconfig /all
```

Configuración IP de Windows

```
Nombre de host. . . . . : PC-PC
Sufijo DNS principal . . . . :
Tipo de nodo. . . . . : híbrido
Enrutamiento IP habilitado. . . : no
Proxy WINS habilitado . . . . : no
```

Adaptador de Ethernet Conexión de área local:

```
Sufijo DNS específico para la conexión. . :
Descripción . . . . . : Realtek PCIe GBE Family Controller
Dirección física. . . . . : 94-DE-80-51-45-AF
DHCP habilitado . . . . . : sí
Configuración automática habilitada . . . : sí
Vínculo: dirección IPv6 local. . . : fe80::799a:a25d:a22f:f0dd%4(Preferido)
Dirección IPv4. . . . . : 192.168.1.102(Preferido)
Máscara de subred . . . . . : 255.255.255.0
Concesión obtenida. . . . . : jueves, 01 de febrero de 2018 16:49:31
La concesión expira . . . . . : viernes, 02 de febrero de 2018 16:49:30
Puerta de enlace predeterminada . . . . : 192.168.1.1
Servidor DHCP . . . . . : 192.168.1.1
IAID DHCPv6 . . . . . : 244637312
DUID de cliente DHCPv6. . . . . : 00-01-00-01-19-E6-DC-79-94-DE-80-51-45-AF
Servidores DNS. . . . . : 8.8.8.8
                        194.179.1.100
                        212.73.32.67
NetBIOS sobre TCP/IP. . . . . : habilitado
```

Adaptador de túnel Teredo Tunneling Pseudo-Interface:

```
Sufijo DNS específico para la conexión. . :
Descripción . . . . . : Teredo Tunneling Pseudo-Interface
Dirección física. . . . . : 00-00-00-00-00-00-E0
DHCP habilitado . . . . . : no
Configuración automática habilitada . . . : sí
Dirección IPv6 . . . . . : 2001:0:9d38:6abd:1c14:7de:3f57:fe99(Preferido)
Vínculo: dirección IPv6 local. . . : fe80::1c14:7de:3f57:fe99%7(Preferido)
Puerta de enlace predeterminada . . . . : ::
IAID DHCPv6 . . . . . : 167772160
DUID de cliente DHCPv6. . . . . : 00-01-00-01-19-E6-DC-79-94-DE-80-51-45-AF
NetBIOS sobre TCP/IP. . . . . : deshabilitado
```

```
C:\Users\PC>ipconfig
```

Configuración IP de Windows

Adaptador de Ethernet Conexión de área local:

```
Sufijo DNS específico para la conexión. . :
Vínculo: dirección IPv6 local. . . : fe80::799a:a25d:a22f:f0dd%4
Dirección IPv4. . . . . : 192.168.1.102
Máscara de subred . . . . . : 255.255.255.0
Puerta de enlace predeterminada . . . . : 192.168.1.1
```

Adaptador de túnel Teredo Tunneling Pseudo-Interface:

```
Sufijo DNS específico para la conexión. . :
Dirección IPv6 . . . . . : 2001:0:9d38:6abd:1c14:7de:3f57:fe99
Vínculo: dirección IPv6 local. . . : fe80::1c14:7de:3f57:fe99%7
Puerta de enlace predeterminada . . . . : ::
```

## Actividad 2.- Máquina virtual en Linux

### A) Documenta la instalación y configuración de una máquina virtual teniendo en cuenta las siguientes especificaciones.

**Máquina Virtual: Máquina Limpia PCB-USED**

**Sistema Operativo** Ubuntu Server

**Memoria RAM** 0,5GB

**Particiones** Disco 500GB **Sistema:**100 GB

**Datos:**298 GB

**Swap:**2 GB

**Nombre de la máquina** PCB-USED

**Usuarios y contraseñas** miadmin paso  
usuario paso

**Configuración de red** Puerta de Enlace: **DNS:**192.168.20.20 **Dirección IP:**192.168.3.111  
192.168.3.1

### B) Comprobar la configuración

Configuración de red:

```
miadmin@PCB-ED:~$ ifconfig
enp0s3    Link encap:Ethernet  direcciónHW 08:00:27:d5:66:cd
          Direc. inet:192.168.1.150  Difus.:192.168.1.255  Másc:255.255.255.0
          Dirección inet6: fe80::a00:27ff:fed5:66cd/64 Alcance:Enlace
          ACTIVO DIFUSIÓN FUNCIONANDO MULTICAST MTU:1500 Métrica:1
          Paquetes RX:131478 errores:0 perdidos:1 overruns:0 frame:0
          Paquetes TX:128329 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colaTX:1000
          Bytes RX:53695428 (53.6 MB) TX bytes:17940345 (17.9 MB)

lo        Link encap:Bucle local
          Direc. inet:127.0.0.1  Másc:255.0.0.0
          Dirección inet6: ::1/128 Alcance:Anfitrión
          ACTIVO BUCLE FUNCIONANDO MTU:65536 Métrica:1
          Paquetes RX:15585 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:15585 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colaTX:1
          Bytes RX:1158947 (1.1 MB) TX bytes:1158947 (1.1 MB)
```

Fichero /etc/network/interfaces:



```
miadmin@PCB-ED:~$ cat /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto enp0s3
iface enp0s3 inet static
address 192.168.1.150
gateway 192.168.1.1
netmask 255.255.255.0
dns-nameservers 192.168.1.150
dns-domain pablo.local
```

Fichero /etc/resolv.conf

```
miadmin@PCB-ED:~$ cat /etc/resolv.conf
# Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf(8)
#     DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN
nameserver 192.168.1.150
search pablo.local
```

Fichero /etc/hosts:

```
miadmin@PCB-ED:~$ cat /etc/hosts
127.0.0.1        localhost
127.0.1.1        PCB-ED

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

Fichero /etc/hostname:

```
miadmin@PCB-ED:~$ cat /etc/hostname
PCB-ED
```

## Actividad 3.- Servidor de aplicaciones

### A) Documenta la instalación y configuración de un servidor de Aplicaciones

En primer lugar clona la máquina anterior, y cambia los siguientes datos:

**Máquina Virtual: Máquina limpia PCB-USED**

Sistema Operativo	Ubuntu Server	
Memoria RAM	0,5GB	
Particiones	Disco 500GB	Sistema:100 GB
		Datos:298 GB

Pablo Cidón Barrio

Swap:2 GB

Nombre de la máquina PCB-USED

Usuarios y contraseñas miadmin paso  
usuario paso

Configuración de red Puerta de Enlace: DNS:192.168.20.20 Dirección IP:192.168.3.111  
192.168.3.1

### Servicios

Servicio FTP	SFTP	Usuario:operadorweb
Servicio SSH	Open SSH Server	
Servidor Web	Apache (modulo de PHP)	
SGBD	MySQL	Usuario SGBD:miadmindb/pasodb
	Directorio para los scripts de la base de datos:	Usuario de creación:
	/var/scriptsDB	usuario/paso

Comprobamos la configuración de red:

```
miadmin@PCB-ED:~$ cat /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto enp0s3
iface enp0s3 inet static
address 192.168.1.150
gateway 192.168.1.1
netmask 255.255.255.0
dns-nameservers 192.168.1.150
dns-domain pablo.local

miadmin@PCB-ED:~$ cat /etc/resolv.conf
# Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf(8)
#     DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN
nameserver 192.168.1.150
search pablo.local
```

Cambiamos el nombre del servidor:

```
miadmin@PCB-ED:~$ sudo nano /etc/hostname
```

```
miadmin@PCB-ED:~$ sudo nano /etc/hosts
```

Reiniciamos la máquina para aplicar los cambios:

```
miadmin@PCB-ED:~$ sudo reboot now
```

Actualizamos el repositorio de las aplicaciones de ubuntu:

```
miadmin@PCB-ED:~$ sudo apt-get update
```

Comenzamos con la instalación de apache:

```
miadmin@PCB-ED:~$ sudo apt-get install apache2
```

A continuación instalamos el Open SSH:

```
miadmin@PCB-ED:~$ sudo apt-get install openssh-server
```

Luego instalamos el servidor de mysql:

```
miadmin@PCB-ED:~$ sudo apt-get install mysql-server
```

Configuramos el fichero de mysql, comentando el bind address (#):

```
miadmin@PCB-ED:~$ sudo nano /etc/mysql/my.cnf
```

En caso de que no aparezca la línea tendremos que ir al siguiente fichero para configurarlo:

```
miadmin@PCB-ED:~$ sudo nano /etc/mysql/mysql.conf.d/mysqld.cnf
# bind-address            = 127.0.0.1
```

Reiniciamos el servicio de mysql:

```
miadmin@PCB-ED:~$ sudo service mysql restart
```

Instalamos phpMyAdmin:

```
miadmin@PCB-ED:~$ sudo apt-get install phpmyadmin
```

Durante la instalación señalamos la opción para que configure apache

## B) Comprobar la configuración

Utiliza el comando ping para comprobar que existe comunicación con las otras máquinas, con la puerta de enlace y con Internet.

Configuración de apache:

```
miadmin@PCB-ED:~$ cat /etc/apache2/apache2.conf
```

```

-
-
<Directory />
    Options FollowSymLinks
    AllowOverride None
    Require all denied
</Directory>

<Directory /usr/share>
    AllowOverride None
    Require all granted
</Directory>

<Directory /var/www/>
    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
</Directory>

```

### Configuración de OpenSSH:

```

SSH: SSH:
miadmin@PCB-ED:~$ cat /etc/ssh/ssh_config

```

```

Host *
# ForwardAgent no
# ForwardX11 no
# ForwardX11Trusted yes
# RhostsRSAAuthentication no
# RSAAuthentication yes
# PasswordAuthentication yes
# HostbasedAuthentication no
# GSSAPIAuthentication no
# GSSAPIDelegateCredentials no
# GSSAPIKeyExchange no
# GSSAPITrustDNS no
# BatchMode no
# CheckHostIP yes
# AddressFamily any
# ConnectTimeout 0
# StrictHostKeyChecking ask
# IdentityFile ~/.ssh/identity
# IdentityFile ~/.ssh/id_rsa
# IdentityFile ~/.ssh/id_dsa
# IdentityFile ~/.ssh/id_ecdsa
# IdentityFile ~/.ssh/id_ed25519
# Port 22
# Protocol 2
# Cipher 3des
# Ciphers aes128-ctr,aes192-ctr,aes256-ctr,arcfour256,arcfour128,aes128-cbc,3des-cbc
# MACs hmac-md5,hmac-sha1,umac-64@openssh.com,hmac-ripemd160
# EscapeChar ~
# Tunnel no
# TunnelDevice any:any
# PermitLocalCommand no
# VisualHostKey no
# ProxyCommand ssh -q -W %h:%p gateway.example.com
# RekeyLimit 1G 1h
# SendEnv LANG LC_*
# HashKnownHosts yes
# GSSAPIAuthentication yes
# GSSAPIDelegateCredentials no

```

### Configuración de mysql:

```
miadmin@PCB-ED:~$ cat /etc/mysql/my.cnf
#
# The MySQL database server configuration file.
#
# You can copy this to one of:
# - "/etc/mysql/my.cnf" to set global options,
# - "~/.my.cnf" to set user-specific options.
#
# One can use all long options that the program supports.
# Run program with --help to get a list of available options and with
# --print-defaults to see which it would actually understand and use.
#
# For explanations see
# http://dev.mysql.com/doc/mysql/en/server-system-variables.html
#
# * IMPORTANT: Additional settings that can override those from this file!
#   The files must end with '.cnf', otherwise they'll be ignored.
#

!includedir /etc/mysql/conf.d/
!includedir /etc/mysql/mysql.conf.d/
```

Configuración de phpMyAdmin:

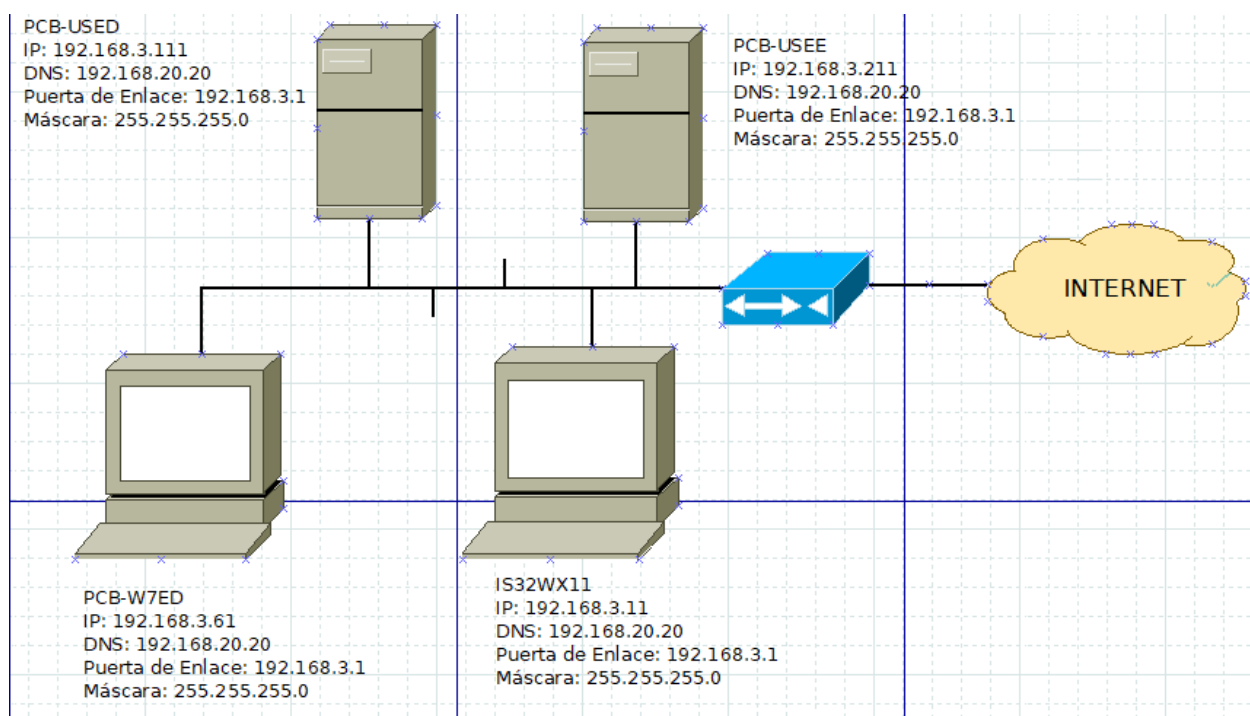
```
miadmin@PCB-ED:~$ cat /etc/phpmyadmin/config.inc.php
/* Authentication type */
// $cfg['Servers'][$i]['auth_type'] = 'cookie';
/* Server parameters */
// $cfg['Servers'][$i]['host'] = 'localhost';
// $cfg['Servers'][$i]['connect_type'] = 'tcp';
// $cfg['Servers'][$i]['compress'] = false;
/* Uncomment the following to enable logging in to passwordless accounts,
 * after taking note of the associated security risks. */
// $cfg['Servers'][$i]['AllowNoPassword'] = TRUE;
```

## C) Puertos y conexiones

Averigua los puertos TCP y UDP a la escucha en esta máquina virtual.

```
miadmin@PCB-ED:~$ netstat -ltn
Conexiones activas de Internet (solo servidores)
Proto Recib Enviad Dirección local Dirección remota Estado
tcp 0 0 192.168.1.150:53 0.0.0.0:* ESCUCHAR
tcp 0 0 127.0.0.1:53 0.0.0.0:* ESCUCHAR
tcp 0 0 0.0.0.0:22 0.0.0.0:* ESCUCHAR
tcp 0 0 127.0.0.1:953 0.0.0.0:* ESCUCHAR
tcp6 0 0 :::3306 :::* ESCUCHAR
tcp6 0 0 :::80 :::* ESCUCHAR
tcp6 0 0 :::53 :::* ESCUCHAR
tcp6 0 0 :::22 :::* ESCUCHAR
tcp6 0 0 :::1:953 :::* ESCUCHAR
```

## Actividad 4.- Estructura del entorno de desarrollo.



## Actividad 5.- Mantenimiento de los servidores Entorno de Desarrollo y Explotación.

(como controlar los servicios instalados, ficheros de configuración, modificación en los ficheros de configuración a partir de su instalación).

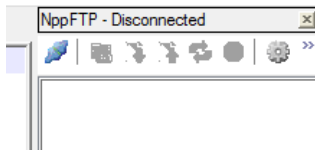
	Apache	MySQL	OpenSSH
Ficheros de configuración	/etc/apache2/apache2.conf /etc/apache2/ports.conf	/etc/mysql/my.cnf my.ini (Windows)	/etc/ssh/ssh_config
Cambios realizados		# bind-address	
Reinicio de servicios	service apache2 restart	service mysql restart	Service ssh restart

## Actividad 6.- Explica diferentes IDE

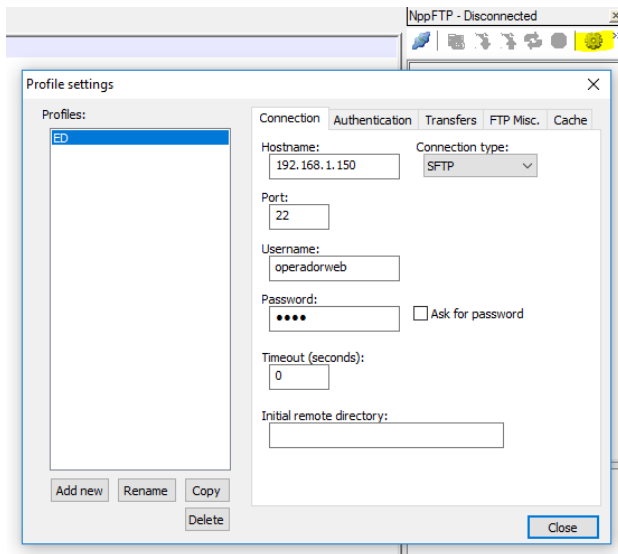
(NotePad++, Netbeans,...) como se configura para trabajar con el servidor Entorno de desarrollo.

### Notepad ++

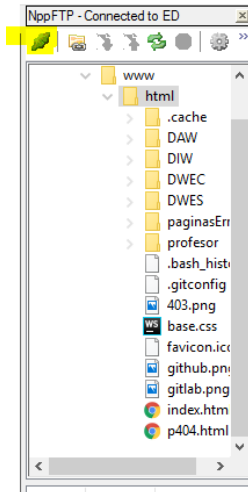
En el menú, vamos a plugins, luego a NPP\_FTP y seleccionamos la opción de mostrar la ventana FTP:



Luego vamos a settings y nos aparecerá un diálogo el que introduciremos los datos para establecer la conexión.

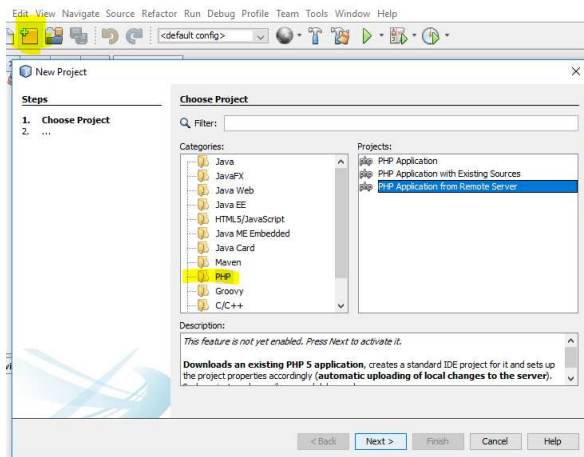


Una vez hecho probamos si funciona.

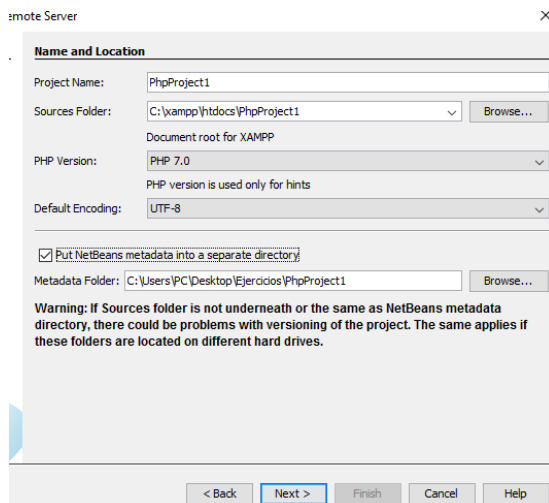


## Netbeans

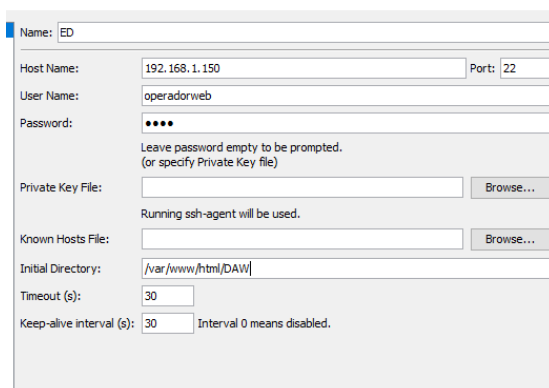
Creamos un nuevo proyecto, de PHP y en un servidor remoto.



Luego establecemos donde vamos a guardar el proyecto, los metadatos generados y la versión de PHP que se va a usar.

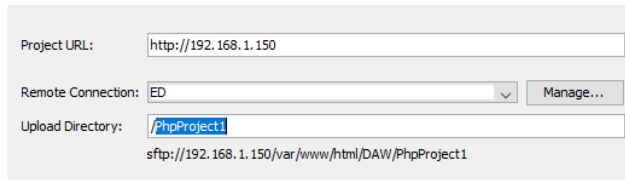


A continuación, creamos la conexión con el servidor.



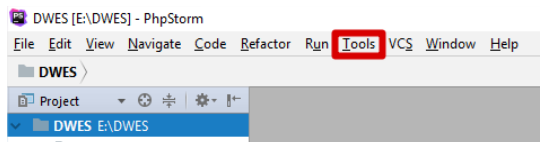
Establecemos la URL y ya lo tenemos.



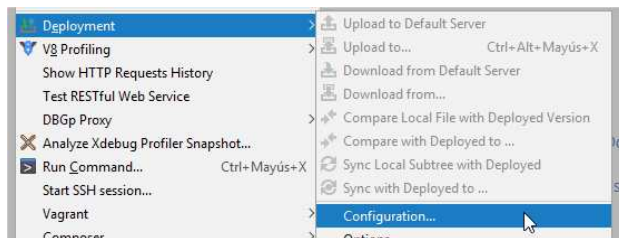


## PhpStorm/WebStorm

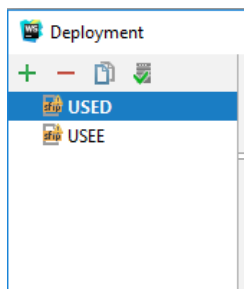
Abrimos nuestro proyecto, y luego vamos a la pestaña de tools.



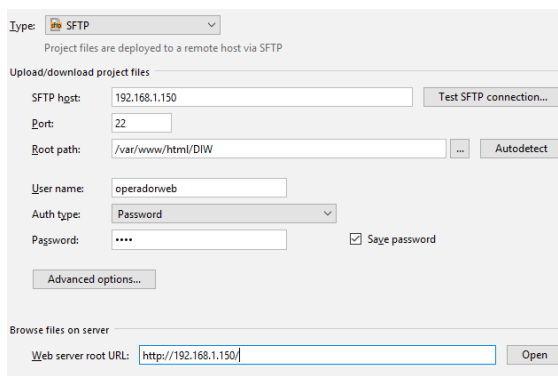
A continuación vamos a la opción de deployment y luego a configuración.



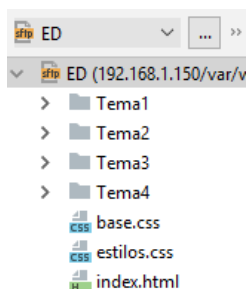
Luego nos aparecerá una ventana con las configuraciones que tenemos, para crear una nueva pulsaremos en +.



Luego completamos la conexión.

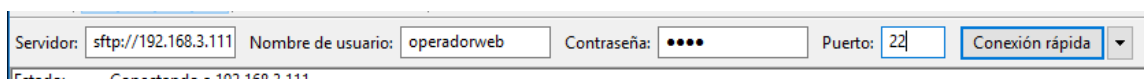


Finalmente comprobamos que se ha realizado correctamente.

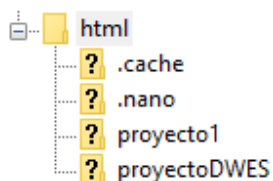


## Actividad 7.- Proceso de despliegue de una aplicación de Entorno de Desarrollo a Explotación.

En primer lugar, descargamos los archivos del entorno de desarrollo. Para ello tendremos que conectarnos con FileZilla al entorno de desarrollo.



Una vez realizada la conexión, seleccionamos la carpeta que vamos a descargar y la ubicación de la misma. Para descargarla pulsamos con el botón derecho sobre el directorio que vayamos a descargar.



Una vez descargada, nos aparecerá una carpeta con el contenido descargado, donde lo hayamos guardado.



A continuación, nos conectaremos con el entorno de explotación desde FileZilla.



Una vez conectados, seleccionamos los elementos que vamos a subir, dado que alguno de los contenidos ya se encuentran en el servidor por defecto, unicamente seleccionamos los que no estén, aunque podemos subirlos todos y los que están se reemplazarán.

proyecto1	28/09/2017 12:28	Carpeta de archivos	
proyectoDWES	28/09/2017 12:28	Carpeta de archivos	
.bash_history	28/09/2017 12:26	Archivo BASH_HIS...	2 KB
.gitconfig	28/09/2017 12:26	Archivo GITCONFIG	1 KB
base.css	28/09/2017 12:26	Documento de ho...	1 KB
daw.html	28/09/2017 12:26	Chrome HTML Do...	1 KB
diw.html	28/09/2017 12:26	Chrome HTML Do...	1 KB
dwec.html	28/09/2017 12:26	Chrome HTML Do...	1 KB
dwes.html	28/09/2017 12:26	Chrome HTML Do...	2 KB
estilos.css	28/09/2017 12:26	Documento de ho...	1 KB
favicon.ico	28/09/2017 12:26	Archivo ICO	2 KB
index.html	28/09/2017 12:26	Chrome HTML Do...	1 KB

Una vez seleccionamos, unicamente tendremos que arrastrar los archivos hasta el entorno de explotación.

Nombre de archivo	Tamaño d...	Tipo de arc...	Última modific...	Permisos	Propietario...
.nano		Carpeta de...	25/09/2017 9:4...	drwxr-xr-x	operadorw...
proyecto1		Carpeta de...	26/09/2017 10:...	drwxr-xr-x	operadorw...
proyectoDWES		Carpeta de...	27/09/2017 11:...	drwxr-xr-x	operadorw...
.bash_history	1.266	Archivo BA...	26/09/2017 10:...	-rw-----	operadorw...
.gitconfig	84	Archivo GI...	26/09/2017 10:...	-rw-r--r--	operadorw...
base.css	240	Document...	21/09/2017 13:...	-rw-r--r--	operadorw...
daw.html	778	Chrome H...	26/09/2017 10:...	-rw-r--r--	operadorw...
diw.html	792	Chrome H...	21/09/2017 12:...	-rw-r--r--	operadorw...
dwec.html	798	Chrome H...	21/09/2017 12:...	-rw-r--r--	operadorw...
dwes.html	1.603	Chrome H...	26/09/2017 12:...	-rw-r--r--	operadorw...
estilos.css	403	Document...	21/09/2017 12:...	-rw-r--r--	operadorw...
favicon.ico	1.150	Archivo ICO	20/09/2017 10:...	-rw-r--r--	operadorw...
index.html	889	Chrome H...	28/09/2017 12:...	-rw-r--r--	operadorw...

Una vez subidos, accederemos al servidor desde el navegador para comprobar que todos los documentos se han subido correctamente.

## DESARROLLO DE APLICACIONES WEB

Desarrollo Web Entorno Servidor

Desarrollo Web Entorno Cliente

Despliegue de Aplicaciones Web

Diseño de Interfaces Web

Pablo Cidon Curso 2017-2018  
Entorno de Explotación

## UD 2 Sistema de Control de Versiones

### Actividad 1.- Instalación de Gitlab en una máquina virtual en Ubuntu Server

Máquina Virtual: PCB-GIT		
<b>Sistema Operativo</b>	Ubuntu Server 16.04.1LTS x64	
<b>Memoria RAM</b>	Según los requisitos de instalación	
<b>Particiones</b>	Disco dinámico de 500GB	<b>Sistema:</b> 100GB <b>Swap:</b> 2GB <b>Datos:</b> 298GB
<b>Nombre de la máquina</b>	PCB-GIT	
<b>Usuarios y contraseñas</b>	miadmin	paso
	usuario	paso
	operadorweb	paso
<b>Configuración de red</b>	<b>IP:</b> 192.168.3.150+nº de equipo/24 <b>Puerta de enlace:</b> 192.168.3.1 <b>DNS:</b> 192.168.20.20	

Partimos de la máquina que tenemos de entorno de desarrollo.

Realizamos la configuración de red:

```
iface enp0s3 inet static
    address 192.168.3.161
    netmask 255.255.255.0
    gateway 192.168.3.1
    dns-nameservers 192.168.20.20
```

También cambiamos el nombre del servidor:

```
PCB-GIT
```

Procedemos a la instalación del repositorio Git:

Actualizamos el repositorio:

```
miadmin@PCB-USED:~$ sudo apt-get update
Des:1 http://security.ubuntu.com/ubuntu xenial-security InRelease [102 kB]
Obj:2 http://es.archive.ubuntu.com/ubuntu xenial InRelease
Des:3 http://es.archive.ubuntu.com/ubuntu xenial-updates InRelease [102 kB]
Des:4 http://es.archive.ubuntu.com/ubuntu xenial-backports InRelease [102 kB]
Descargados 306 kB en 2s (134 kB/s)
```

Descargamos el servidor openssh:

```
miadmin@PCB-GIT:~$ sudo apt-get install -y curl openssh-server ca-certificates
[sudo] password for miadmin:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
```

Luego instalaremos el Postfix:

```
miadmin@PCB-GIT:~$ sudo apt-get install -y postfix
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
```

Instalamos el repositorio de GitLab:

```
miadmin@PCB-GIT:~$ curl -sS https://packages.gitlab.com/install/repositories/gitlab/gitlab-ce/script.deb.sh | sudo bash
```

Instalamos el paquete de GitLab:

```
miadmin@PCB-GIT:~$ sudo apt-get install gitlab-ce
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
```


<https://about.gitlab.com/installation/#ubuntu>

## Actividad 2.- Instalación de cliente git en Windows,

<https://git-scm.com/downloads>

Nos dirigimos a la página de GIT, y automáticamente se iniciará la descarga automáticamente.

## Downloading Git



**Your download is starting...**

You are downloading the latest (2.14.1) 64-bit version of Git for Windows. This is the most recent **maintained build**. It was released **about 2 months ago**, on 2017-08-10.

If your download hasn't started, [click here to download manually](#).

**Other Git for Windows downloads**

Git for Windows Setup  
[32-bit Git for Windows Setup](#).  
[64-bit Git for Windows Setup](#).

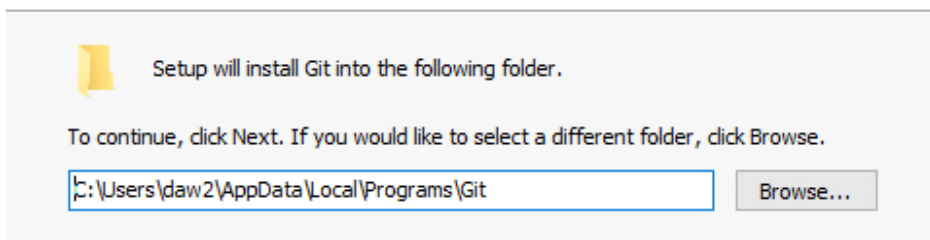
Git for Windows Portable ("thumbdrive edition")  
[32-bit Git for Windows Portable](#).  
[64-bit Git for Windows Portable](#).

The current source code release is version 2.14.2. If you want the newer version, you can build it from [the source code](#).

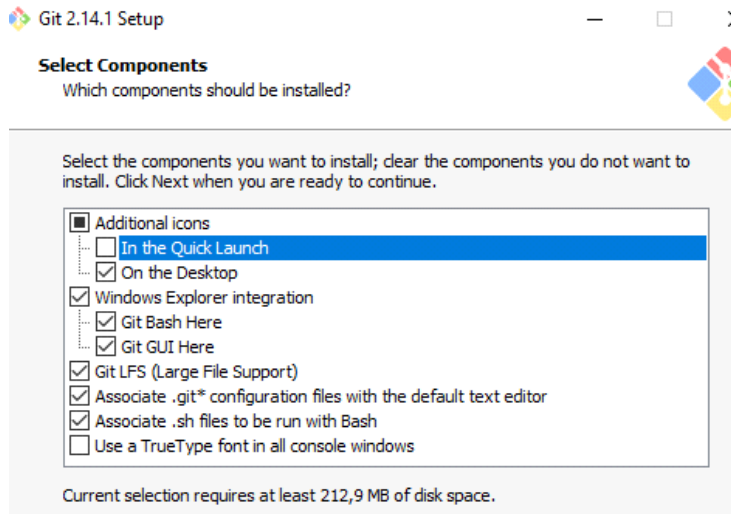
Una vez descargado el paquete lo ejecutamos y comenzaremos la instalación.



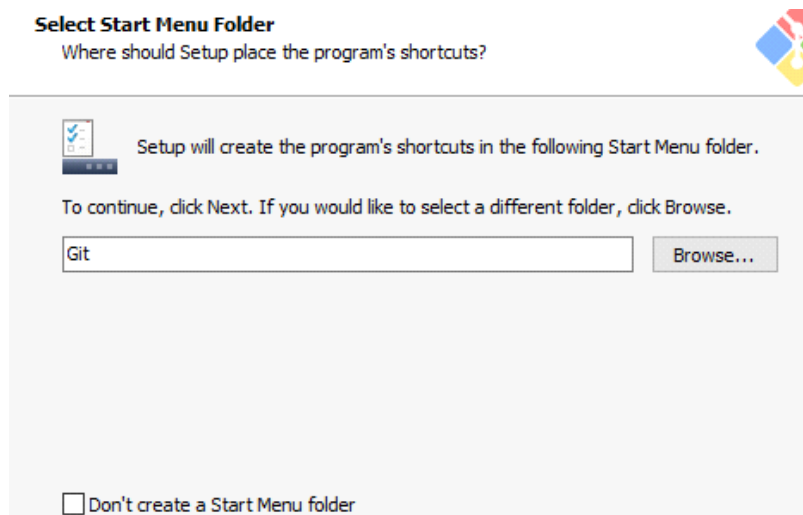
En primer lugar nos aparecerá la ubicación en la que instalaremos el programa. Aunque podremos dejar la que viene por defecto.



A continuación, elegiremos los componentes que querramos instalar:



Luego nos aparecerá la opción de crear una carpeta en el menú de inicio, si no la queremos realizar solo tendremos que marcar la opción de no crearla y ya está.



A continuación nos aparecerán las opciones para el uso del Git, en este caso dejaremos la segunda, que es la que aparece por defecto. Y nos permite usar git por consola de comandos.

**Adjusting your PATH environment**

How would you like to use Git from the command line?

☐ **Use Git from Git Bash only**

This is the safest choice as your PATH will not be modified at all. You will only be able to use the Git command line tools from Git Bash.

☒ **Use Git from the Windows Command Prompt**

This option is considered safe as it only adds some minimal Git wrappers to your PATH to avoid cluttering your environment with optional Unix tools. You will be able to use Git from both Git Bash and the Windows Command Prompt.

☐ **Use Git and optional Unix tools from the Windows Command Prompt**

Both Git and the optional Unix tools will be added to your PATH.

**Warning: This will override Windows tools like "find" and "sort". Only use this option if you understand the implications.**

[tps://git-for-windows.github.io/](https://git-for-windows.github.io/)

&lt; Back

Next &gt;

Cancel

A continuación nos aparece la opción de la instalación de las librerías, en este caso dejaremos la que viene por defecto.

☒ **Use the OpenSSL library**

Server certificates will be validated using the ca-bundle.crt file.

☐ **Use the native Windows Secure Channel library**

Server certificates will be validated using Windows Certificate Stores. This option also allows you to use your company's internal Root CA certificates distributed e.g. via Active Directory Domain Services.

Después nos aparecerá la configuración de Checkout, en este caso también dejaremos la que viene señalada por defecto.

**Configuring the line ending conversions**

How should Git treat line endings in text files?

☒ **Checkout Windows-style, commit Unix-style line endings**

Git will convert LF to CRLF when checking out text files. When committing text files, CRLF will be converted to LF. For cross-platform projects, this is the recommended setting on Windows ("core.autocrlf" is set to "true").

☐ **Checkout as-is, commit Unix-style line endings**

Git will not perform any conversion when checking out text files. When committing text files, CRLF will be converted to LF. For cross-platform projects, this is the recommended setting on Unix ("core.autocrlf" is set to "input").

☐ **Checkout as-is, commit as-is**

Git will not perform any conversions when checking out or committing text files. Choosing this option is not recommended for cross-platform projects ("core.autocrlf" is set to "false").



A continuación seleccionamos la consola que queremos utilizar, dejaremos la del programa, que viene señalada por defecto.

#### Configuring the terminal emulator to use with Git Bash

Which terminal emulator do you want to use with your Git Bash?



☒ **Use MinTTY (the default terminal of MSYS2)**

Git Bash will use MinTTY as terminal emulator, which sports a resizable window, non-rectangular selections and a Unicode font. Windows console programs (such as interactive Python) must be launched via `winpty` to work in MinTTY.

☐ **Use Windows' default console window**

Git will use the default console window of Windows ("cmd.exe"), which works well with Win32 console programs such as interactive Python or node.js, but has a very limited default scroll-back, needs to be configured to use a Unicode font in order to display non-ASCII characters correctly, and prior to Windows 10 its window was not freely resizable and it only allowed rectangular text selections.

A continuación marcamos las opciones que vienen por defecto y comenzará la instalación el programa.

#### Installing

Please wait while Setup installs Git on your computer.

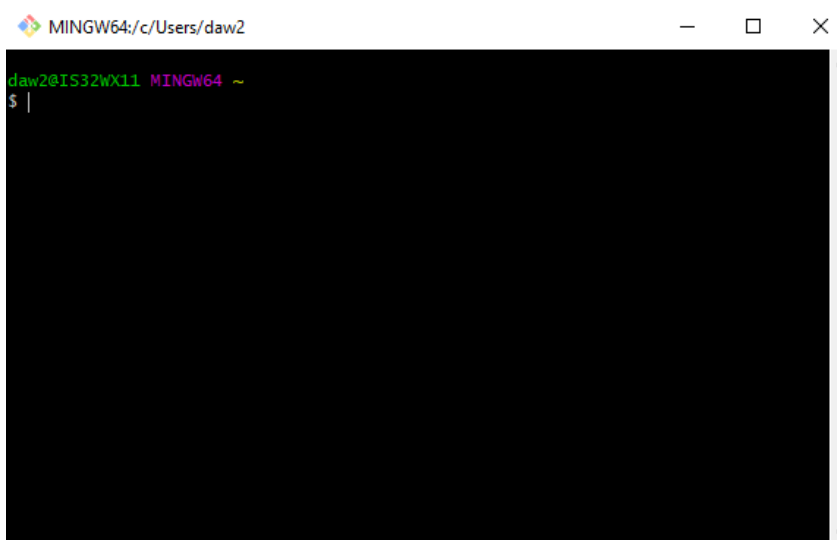


Extracting files...

C:\...\daw2\AppData\Local\Programs\Git\mingw64\lib\tcl8.6\tzdata\Australia\Victoria



Una vez acabada la instalación, se ejecutará el programa.



## Actividad 3.- Trabajando con Git

En primer lugar comprobamos que está instalado el git y la version:

```
operadorweb@PCB-USED:~$ git --version
git version 2.7.4
```

Vamos al directorio del usuario:

```
operadorweb@PCB-USED:~$ cd /var/www/html
```

Creamos el directorio del repositorio:

```
operadorweb@PCB-USED:~$ mkdir proyecto1
```

Accedemos al directorio y creamos el repositorio:

```
operadorweb@PCB-USED:~$ cd proyecto1/
operadorweb@PCB-USED:~/proyecto1$ git init
Initialized empty Git repository in /var/www/html/proyecto1/.git/
```

Hacemos un listado para ver los archivos del repositorio que están ocultos:

```
operadorweb@PCB-USED:~/proyecto1$ ls -a
.  ..  .git
```

Creamos dos archivos:

```
operadorweb@PCB-USED:~/proyecto1$ nano README.md
```

```
operadorweb@PCB-USED:~/proyecto1$ nano index.html
```

Comprobamos el status:

```
operadorweb@PCB-USED:~/proyecto1$ git status
En la rama master

Commit inicial

Archivos sin seguimiento:
  (use «git add <archivo>...» para incluir en lo que se ha de confirmar)

    README.md
    index.html
```

Cambiamos el archivo de zona: (a stage)

```
operadorweb@PCB-USED:~/proyecto1$ git add index.html
operadorweb@PCB-USED:~/proyecto1$ git status
En la rama master

Commit inicial

Cambios para hacer commit:
  (use «git rm --cached <archivo>...» para sacar del stage)

    nuevo archivo: index.html
```

Devolvemos el archivo a su zona: (a working)

```
operadorweb@PCB-USED:~/proyecto1$ git rm --cached index.html
rm 'index.html'
operadorweb@PCB-USED:~/proyecto1$ git status
En la rama master

Commit inicial

Archivos sin seguimiento:
  (use «git add <archivo>...» para incluir en lo que se ha de confirmar)

    README.md
    index.html
```

Volvemos a mover el archivo y lo editamos modificamos el fichero: (a stage)

```
operadorweb@PCB-USED:~/proyecto1$ git status
En la rama master

Commit inicial

Cambios para hacer commit:
  (use «git rm --cached <archivo>...» para sacar del stage)

    nuevo archivo: index.html

Cambios no preparados para el commit:
  (use «git add <archivo>...» para actualizar lo que se confirmará)
  (use «git checkout -- <archivo>...» para descartar cambios en el directorio de trabajo)

    modificado:    index.html
```

Si queremos volver a moverlo y está modificado: (a working)

```
operadorweb@PCB-USED:~/proyecto1$ git rm --cached -f index.html
rm 'index.html'
```

Si queremos mover todos los archivos: (a stage)

```
operadorweb@PCB-USED:~/proyecto1$ git add .
operadorweb@PCB-USED:~/proyecto1$ git status
En la rama master

Commit inicial

Cambios para hacer commit:
  (use «git rm --cached <archivo>...» para sacar del stage)

    nuevo archivo: README.md
    nuevo archivo: index.html
```

Configuración de usuarios:

```
operadorweb@PCB-USED:~/proyecto1$ git config --global user.mail "pablo.cidbar@gmail.com"
operadorweb@PCB-USED:~/proyecto1$ git config --global user.name "DAW2PCB"
```

Realizar commits:

```
operadorweb@PCB-USED:~/proyecto1$ git commit -m "mi primer commit"
```

## Actividad 4.- Trabajando con ramas

## Actividad 5.- Ramas y fusiones en local (FAST-Forward)

### a) Borrar el repositorio remoto DAW1718

En primer lugar vamos al repositorio que deseamos eliminar:



A continuación vamos a la pestaña settings:



Una vez dentro, abajo del todo nos aparece la opción de eliminar el repositorio.

## Danger Zone

Make this repository private  
Please [upgrade your plan](#) to make this repository private.

Transfer ownership  
Transfer this repository to another user or to an organization where you have the ability to create repositories.

Transfer

Delete this repository  
Once you delete a repository, there is no going back. Please be certain.

Delete this repository

Pulsamos en 'Delete this repository' y a continuación nos saldrá un diálogo de confirmación.

Delete this repository

Are you ABSOLUTELY sure?

Unexpected bad things will happen if you don't read this!

This action CANNOT be undone. This will permanently delete the **pablocidon/DAW1718** repository, wiki, issues, and comments, and remove all collaborator associations.

Please type in the name of the repository to confirm.

DAW1718

I understand the consequences, delete this repository

## b) Borrar el repositorio local DAW1718

Vamos al repositorio y con el botón derecho pinchamos en la opción de 'Eliminar'.

DAW1718	03/10/2017 10:46	Carpeta de archivos
---------	------------------	---------------------

## c) Crear de nuevo el proyecto HOLAMUNDO, con un commit con el archivo README.md que contiene el nombre de la asignatura.

Vamos a crear un nuevo repositorio para ello, vamos a 'New Repository', que se encuentra en la página principal de git



Una vez realizado, nos aparecerá una ventana en la que tendremos que introducir un nombre para el repositorio, una descripción y señalar la opción de crear el archivo README en caso de que querramos crear el archivo.

#### Create a new repository

A repository contains all the files for your project, including the revision history.

Owner: pablocidon / Repository name: HOLAMUNDO ✓

Great repository names are short and memorable. Need inspiration? How about [upgraded-spoon](#).

Description (optional): Despliegue de Aplicaciones Web

☒ Public  
Anyone can see this repository. You choose who can commit.

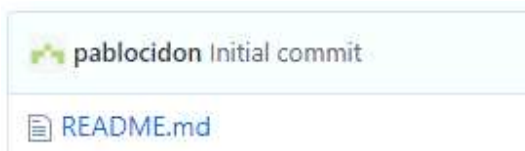
☐ Private  
You choose who can see and commit to this repository.

☒ Initialize this repository with a README  
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing.

Add .gitignore: None Add a license: None ⓘ

Create repository

Automaticamente nos creará un commit con el archivo README



Si abrimos el archivo README, veremos que en él pone el nombre del repositorio y el nombre de la asignatura que ha sido puesta como descripción.

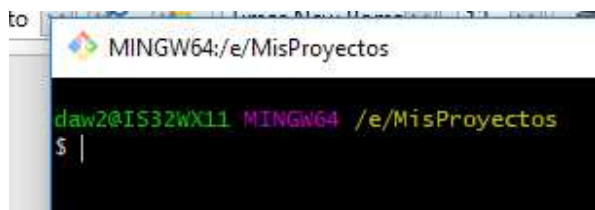
3 lines (2 sloc) | 42 Bytes

# HOLAMUNDO

Despliegue de Aplicaciones Web

## d) Clonación del proyecto en local

Una vez creado, vamos a la carpeta local donde vayamos a guardar el repositorio, y con el botón derecho seleccionamos la opción de abrir GIT Bash. Una vez hecho se abrirá la consola.

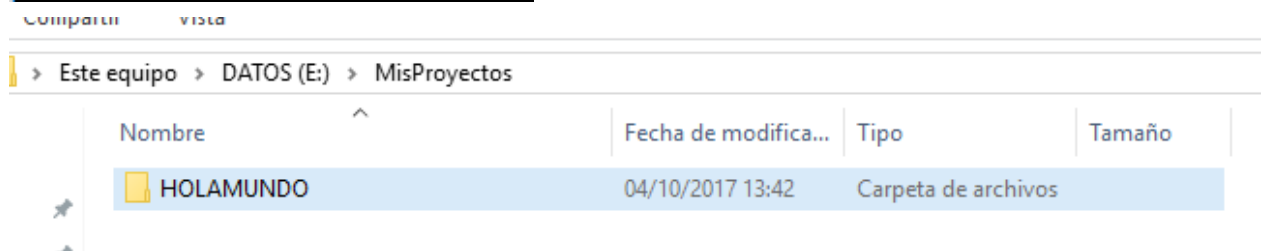


Para descargar el repositorio tendremos que introducir el siguiente comando

```
daw2@IS32WX11 MINGW64 /e/MisProyectos
$ git clone https://github.com/pablocidon/HOLAMUNDO.git
```

Listamos y comprobamos que se ha descargado

```
daw2@IS32WX11 MINGW64 /e/MisProyectos
$ ls
HOLAMUNDO/
```



## e) Crear una página web, index.html, con el mensaje “Hola Mundo” sin estilos. Una vez terminada, súbela al repositorio remoto.

En primer lugar nos dirigiremos al directorio del repositorio con el que vamos a trabajar.

Nos aparecerá entre paréntesis máster, que indica en la rama en la que vamos a trabajar.

```
daw2@IS32WX11 MINGW64 /e/MisProyectos
$ cd HOLAMUNDO/

daw2@IS32WX11 MINGW64 /e/MisProyectos/HOLAMUNDO (master)
$ |
```

A continuación creamos el archivo y listamos para comprobar que está creado

```
daw2@IS32WX11 MINGW64 /e/MisProyectos/HOLAMUNDO (master)
$ touch index.html

daw2@IS32WX11 MINGW64 /e/MisProyectos/HOLAMUNDO (master)
$ ls
index.html  README.md
```

Ahora podemos abrir el documento y realizar en él todos los cambios que deseemos.

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <meta charset="UTF-8">
    <title> Hola Mundo</title>
    <meta name="description" content="CSS"/>
    <meta name="author" content="Pablo"/>
  </head>
  <body>
    <h1>Hola Mundo</h1>
  </body>
</html>
```

Una vez realizados los cambios ya podremos realizar el commit.

En primer lugar cambiamos el archivo de area

```
daw2@IS32WX11 MINGW64 /e/MisProyectos/HOLAMUNDO (master)
$ git status
On branch master
Your branch is up-to-date with 'origin/master'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)

    index.html
```

```
daw2@IS32WX11 MINGW64 /e/MisProyectos/HOLAMUNDO (master)
$ git add .

daw2@IS32WX11 MINGW64 /e/MisProyectos/HOLAMUNDO (master)
$ git status
On branch master
Your branch is up-to-date with 'origin/master'.

Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    new file:   index.html
```



d) Añade la etiqueta v1.0 al commit anterior.

```
daw2@IS32WX11 MINGW64 /e/MisProyectos/HOLAMUNDO (master)
$ git tag -a v1.0 -m 'Version 1.0'
```

e) Se piden crear dos diseños con estilos(css/estilo.css) diferentes del index.html, para ello realizamos dos ramas en local, develop1 y develop2.

```
daw2@IS32WX11 MINGW64 /e/MisProyectos/HOLAMUNDO (master)
$ git branch develop1

daw2@IS32WX11 MINGW64 /e/MisProyectos/HOLAMUNDO (master)
$ git branch develop2
```

d) Una vez que el profesor decida que diseño es el más adecuado, se realizará la fusión en la master.

```
daw2@IS32WX11 MINGW64 /e/MisProyectos/HOLAMUNDO (master)
$ git merge develop1
Already up-to-date.
```

e) Se sube al servidor la rama master.

```
daw2@IS32WX11 MINGW64 /e/MisProyectos/HOLAMUNDO (master)
$ git push origin master
```

d) Se puede borrar las ramas develop1 y develop2.

g) Se etiqueta el nuevo commit, como v1.1.

## UD 3 Servidor Web Apache

### Módulo USERDIR

Por defecto este modulo no está activo, por lo que lo tendremos que activar.

```
miadmin@PCB-USED:/etc/apache2/mods-enabled$ sudo a2enmod userdir
Enabling module userdir.
```

Una vez activado reiniciamos el servicio de apache.

```
miadmin@PCB-USED:/etc/apache2/mods-enabled$ sudo service apache2 restart
Restarting Apache2: httpd.
```

Ahora creamos el usuario con el que vamos a crear el directorio personal.

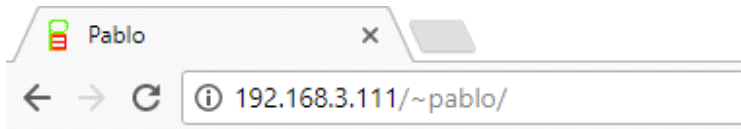
Pablo Cidón Barrio

```
miadmin@PCB-USED:~$ sudo adduser pablo
```

Accedemos a su directorio y creamos el directorio public\_html, allí crearemos un archivo html.

```
miadmin@PCB-USED:~$ cd /home/pablo
miadmin@PCB-USED:/home/pablo$ sudo mkdir public_html_
```

Luego vamos al navegador e introducimos la IP del servidor y el nombre del usuario y nos tendrá que salir el contenido del archivo creado anteriormente.



## Página de Pablo

### Creación de Alias.

Vamos al archivo userdir.conf. Para ello tendremos que seguir la siguiente ruta:

```
miadmin@PCB-USED:/etc/apache2$ cd /etc/apache2/mods-enabled/
```

```
userdir.conf
```

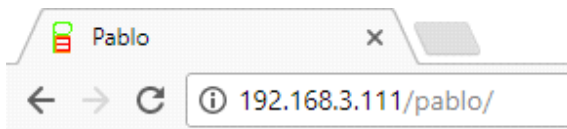
Allí creamos el alias para acceder a nuestro directorio personal.

```
Alias /pablo /home/pablo/public_html
<Directory /home/pablo/public_html>
    Options FollowSymLinks Indexes
    Require all granted
</Directory>
```

Reiniciamos el servicio para aplicar los cambios.

```
miadmin@PCB-USED:/etc/apache2/mods-enabled$ sudo service apache2 restart
miadmin@PCB-USED:/etc/apache2/mods-enabled$
```

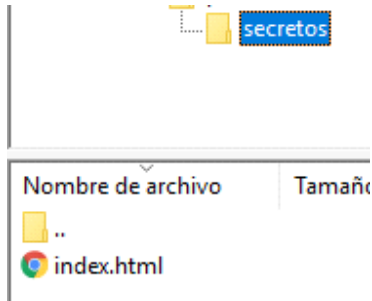
Vamos al navegador e introducimos la IP con el alias.



## Página de Pablo

## Control de Acceso por IP

Creamos el directorio al que vamos a restringir el acceso.



Realizamos la configuración de modo que solo podamos acceder desde la IP asignada.

```
miadmin@PCB-USED:~$ sudo nano /etc/apache2/mods-enabled/userdir.conf_  
<Directory /home/pablo/public_html/secretos>  
    Require ip 192.168.3.11  
</Directory>
```

Reiniciamos el servicio de apache para aplicar los cambios.

```
miadmin@PCB-USED:~$ sudo service apache2 restart_
```

Vamos al navegador y hacemos las comprobaciones.



## Autenticación y autorización basic.

Creamos los usuarios, en este caso los crearemos en el directorio de apache:

```
miadmin@PCB-USED:~$ sudo htpasswd -c /etc/apache2/passwd profesor1  
New password:  
Re-type new password:  
Adding password for user profesor1  
miadmin@PCB-USED:~$ sudo htpasswd /etc/apache2/passwd profesor2  
New password:  
Re-type new password:  
Adding password for user profesor2
```

Una vez hecho comprobamos el fichero para ver si se han creado los usuarios:

```
miadmin@PCB-USED:~$ cat /etc/apache2/passwd  
profesor1:$apr1$w3B277j.$/ca4QiAmHUm2UQpuIld.r0  
profesor2:$apr1$E1stC.Cg$0SiCTLeuDsdpTIfuGDr140
```

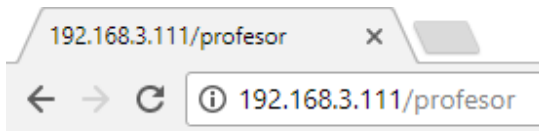
Comprobamos los módulos:

```
operadorweb@PCB-USED:/etc/apache2$ ls mods-enabled | grep authz
authz_core.load
authz_host.load
authz_user.load
```

Vamos al archivo 000-default.conf para crear las restricciones:

```
<Directory /var/www/html/profesor>
    Options FollowSymLinks Indexes
    AllowOverride None
    AuthType Basic
    AuthName "Acceso Restringido"
    AuthUserFile /etc/apache2/passwd
    Require user profesor1 profesor2
</Directory>
```

Reiniciamos el fichero y comprobamos que se han aplicado las restricciones:



### Autenticación obligatoria

http://192.168.3.111

Tu conexión con este sitio web no es privada

Nombre de usuario

Contraseña

Comprobamos como funciona si el usuario está autenticado y como lo hace si no lo está:

# Index of /profesor

<a href="#">Name</a>	<a href="#">Last modified</a>	<a href="#">Size</a>	<a href="#">Description</a>
 <a href="#">Parent Directory</a>		-	

*Apache/2.4.18 (Ubuntu) Server at 192.168.3.111 Port 80*

## Unauthorized

This server could not verify that you are authorized to access the d  
required.

*Apache/2.4.18 (Ubuntu) Server at 192.168.3.111 Port 80*

## Control de acceso desde .htaccess

Acudimos a un generador de htpasswd para genere una contraseña encriptada.

```
alumno:$apr1$/3bZa8b1$V0cidyS40wTTEris289CR1
```

Creamos el fichero y pegamos el codigo generado:

```
alumno:$apr1$/3bZa8b1$V0cidyS40wTTEris289CR1
```

Luego vamos al fichero htaccess, si no está hay que crearlo

```
1 AuthType Basic
2 AuthName "Acceso Restringido"
3 AuthUserFile .htpasswd
4 Require user alumno
```

Finalmente realizamos las comprobaciones y comprobamos que funciona correctamente.

## Autenticación y autorización digest

En primer lugar debemos de activar el módulo digest en apache.

```
miadmin@PCB-USED:/etc/apache2/mods-available$ sudo a2enmod auth_digest.load
```

Una vez hecho debemos de reiniciar el servicio de apache para que se apliquen los cambios.

Pablo Cidón Barrio

```
miadmin@PCB-USED:/etc/apache2/mods-available$ sudo service apache2 restart
```

Ahora procederemos a la creación del usuario:

```
miadmin@PCB-USED:~$ htdigest -c /var/www/.htdigest Alumnos alumno1
Could not open passwd file /var/www/.htdigest for writing: Permission denied
miadmin@PCB-USED:~$ sudo htdigest -c /var/www/.htdigest Alumnos alumno1
Adding password for alumno1 in realm Alumnos.
New password:
Re-type new password:
miadmin@PCB-USED:~$ sudo htdigest /var/www/.htdigest Alumnos alumno2
Adding user alumno2 in realm Alumnos
New password:
Re-type new password:
```

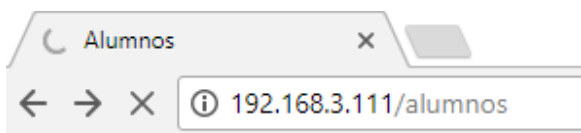
A continuación configuraremos el fichero 000-default.conf:

```
<Directory /var/www/html/alumnos>
    Options FollowSymLinks Indexes
    AuthType Digest
    AuthName "Alumnos"
    AuthDigestProvider file
    AuthUserFile /var/www/.htdigest
    Require user alumno1 alumno2
</Directory>
```

Reiniciamos el servicio de apache:

```
miadmin@PCB-USED:/etc/apache2/sites-enabled$ sudo service apache2 restart
```

Vamos al navegador y comprobamos:



Luego nos aparece el diálogo para acceder:

### Autenticación obligatoria

http://192.168.3.111

Tu conexión con este sitio web no es privada

Nombre de usuario

Contraseña

Iniciar sesión

Cancelar

Accedemos y vemos que funciona. En caso de que el usuario no esté autenticado volverá a salir el diálogo.

# CONTENIDOS PARA EL ALUMADO

## Autenticación y autorización digest por grupos

En primer lugar debemos activar el módulo de apache para la autenticación por grupos. Una vez hecho reiniciamos el servicio.

```
miadmin@PCB-USED:/etc/apache2$ sudo a2enmod authz_groupfile.load
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:
  service apache2 restart
miadmin@PCB-USED:/etc/apache2$ sudo service apache2 restart
```

Creamos un fichero que va a contener los grupos de usuarios:

```
miadmin@PCB-USED:~$ cat /var/www/.htgroup
profesores: profesor1 profesor2
alumnos: alumno1 alumno2
```

Vamos al fichero de configuración y cambiamos las directivas:

```
<Directory /var/www/html/alumnos>
    Options FollowSymLinks Indexes
    AuthType Digest
    AuthName "Alumnos"
    AuthDigestProvider file
    AuthUserFile /var/www/.htdigest
    AuthGroupFile /var/www/.htgroup
    Require group alumnos
</Directory>
```

Reiniciamos el servicio de apache y comprobamos:



# CONTENIDOS PARA EL ALUMADO

## Módulo MOD-STATUS:

Activamos el módulo:

```
miadmin@PCB-USED:/etc/apache2$ sudo a2enmod status.load
```

Reiniciamos el servicio y comprobamos:

## Desde el propio servidor:

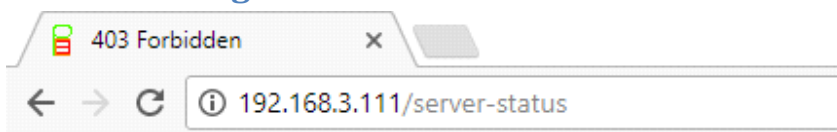
```
miadmin@PCB-USED:/etc/apache2$ sudo apache2ctl status
Apache Server Status for localhost (via ::1)
```

```
Server Version: Apache/2.4.18 (Ubuntu)
Server MPM: prefork
Server Built: 2017-07-27T14:34:01
```

```
Current Time: Monday, 13-Nov-2017 09:25:46 CET
Restart Time: Monday, 13-Nov-2017 09:25:23 CET
Parent Server Config. Generation: 1
Parent Server MPM Generation: 0
Server uptime: 23 seconds
Server load: 0.37 0.13 0.04
Total accesses: 1 - Total Traffic: 0 kB
CPU Usage: u0 s0 cu0 cs0
.0435 requests/sec - 0 B/second - 0 B/request
1 requests currently being processed, 5 idle workers
```

```
Scoreboard Key:
" " Waiting for Connection, "S" Starting up, "R" Reading Request,
"W" Sending Reply, "K" Keepalive (read), "D" DNS Lookup,
"C" Closing connection, "L" Logging, "G" Gracefully finishing,
"I" Idle cleanup of worker, "." Open slot with no current process
```

## Desde el navegador:



# Forbidden

You don't have permission to access /server-status on this server.

*Apache/2.4.18 (Ubuntu) Server at 192.168.3.111 Port 80*

Como no permite acceder debemos modificar el fichero mod\_status.conf:

```
miadmin@PCB-USED:/etc/apache2$ sudo nano mods-enabled/status.conf
```

```
<Location /server-status>
    SetHandler server-status
    Require local
    Require ip 192.168.3.11
</Location>
```

Reiniciamos el servicio y comprobamos que funciona:



## Apache Server Status for 192.168.3.111 (via 192.168.3.111)

Server Version: Apache/2.4.18 (Ubuntu)  
 Server MPM: prefork  
 Server Built: 2017-07-27T14:34:01

### Módulo INFO:

Activamos el módulo y reiniciamos el servicio de apache:

```
miadmin@PCB-USED:/etc/apache2$ sudo a2enmod info.load
Enabling module info.
To activate the new configuration, you need to run:
  service apache2 restart
miadmin@PCB-USED:/etc/apache2$ sudo service apache2 restart
```

Realizamos comprobaciones:

Desde el servidor:

### Configuración de logs en Apache:

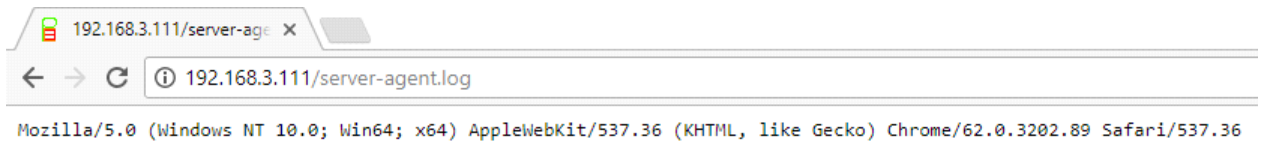
Vamos observamos el fichero de configuración de apache los formatos de log:

```
LogFormat "%v:%p %h %l %u %t \"%r\" %>s %D \"%{Referer}i\" \"%{User-Agent}i\"" vhost_combined
LogFormat "%h %l %u %t \"%r\" %>s %D \"%{Referer}i\" \"%{User-Agent}i\"" combined
LogFormat "%h %l %u %t \"%r\" %>s %D" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent
```

Luego vamos al fichero 000-default.conf y lo configuramos:

```
CustomLog /var/www/html/server-agent.log agent
```

Reiniciamos el servicio y comprobamos en el navegador:



### Virtualización basada en puertos:

Vamos al fichero de configuración de los puertos:

```
miadmin@PCB-USED:~$ sudo nano /etc/apache2/ports.conf
```

```
Listen 80
Listen 90
```

Reiniciamos el servicio y comprobamos:

```
miadmin@PCB-USED:~$ netstat -ltn
Conexiones activas de Internet (solo servidores)
Proto Recib Enviad Dirección local Dirección remota Estado
tcp 0 0 0.0.0.0:3306 0.0.0.0:* ESCUCHAR
tcp 0 0 0.0.0.0:22 0.0.0.0:* ESCUCHAR
tcp6 0 0 :::80 :::* ESCUCHAR
tcp6 0 0 :::81 :::* ESCUCHAR
tcp6 0 0 :::22 :::* ESCUCHAR
miadmin@PCB-USED:~$
```

Comprobamos en el navegador:



Como nos lleva al mismo sitio tendremos que configurar un fichero para que no suceda esto.

Vamos a realizar el archivo de configuración para el puerto 80:

Establecemos el puerto 80, que viene por defecto:

```
<VirtualHost *:80>
```

Y la ruta del documento:

```
DocumentRoot /var/www/puerto80
```

Vamos a realizar el archivo de configuración para el puerto 90:

Establecemos el puerto 90:

```
<VirtualHost *:90>
```

Y la ruta del documento:

```
DocumentRoot /var/www/puerto90
```

Activamos los puertos y reiniciamos el servicio:

```
miadmin@PCB-USED:/etc/apache2/sites-available$ sudo a2ensite puerto80.conf
Site puerto80 already enabled
miadmin@PCB-USED:/etc/apache2/sites-available$ sudo a2ensite puerto90.conf
Site puerto90 already enabled
miadmin@PCB-USED:/etc/apache2/sites-available$
```

También tendremos que desactivar el anterior:

```
miadmin@PCB-USED:/etc/apache2/sites-available$ sudo a2dissite 000-default.conf
```

Cambiamos el propietario de los directorios para que nos permita editarlos:

```
miadmin@PCB-USED:/etc/apache2/sites-available$ sudo chown operadorweb:www-data -R /var/www
```

Comprobamos que funciona:

Puerto 80:



Puerto 90:



## Sitios virtuales basados en nombre

En primer lugar vamos a desactivar los puertos:

```
miadmin@PCB-USED:/etc/apache2/sites-available$ sudo a2dissite puerto80.conf
Site puerto80 disabled.
To activate the new configuration, you need to run:
  service apache2 reload
miadmin@PCB-USED:/etc/apache2/sites-available$ sudo a2dissite puerto90.conf
Site puerto90 disabled.
To activate the new configuration, you need to run:
  service apache2 reload
miadmin@PCB-USED:/etc/apache2/sites-available$ sudo service apache2 reload
```

Ahora vamos a realizar la configuración de hosts en el servidor:

```
192.168.3.111 www.pablo.es
192.168.3.111 www.pablo.com
192.168.3.111 pablo.com
192.168.3.111 pablo.es_
```

Luego realizamos la configuración de los hosts en el cliente:

```
192.168.3.111 www.pablo.com
192.168.3.111 www.pablo.es
192.168.3.111 pablo.com
192.168.3.111 pablo.es|
```

Creacion de los sitios, para ello realizamos la copia de uno de los documentos:

```
miadmin@PCB-USED:/etc/apache2/sites-available$ cp puerto80.conf pablo-com.conf_
```

```
miadmin@PCB-USED:/etc/apache2/sites-available$ cp puerto80.conf pablo-es.conf
```

En el punto com:

```
DocumentRoot /var/www/pablo.com_
```

```
<Directory /var/www/html/pablo.com>
    Options -Indexes
```

```
ServerName www.pablo.com
ServerAlias pablo.com
```

```
ErrorLog ${APACHE_LOG_DIR}/error-pablo-com.log
ErrorLog /var/www/pablo.com/error.log
CustomLog ${APACHE_LOG_DIR}/access-pablo-com.log combined
CustomLog /var/www/pablo.com/access.log combined
```

En el punto es:

```
DocumentRoot /var/www/pablo.es
```

```
<Directory /var/www/html/pablo.es>
    Options -Indexes
    AllowOverride All
</Directory>
```

```
ServerName www.pablo.es
ServerAlias pablo.es
```

```
ErrorLog ${APACHE_LOG_DIR}/error-pablo-es.log
ErrorLog /var/www/pablo.es/error.log
CustomLog ${APACHE_LOG_DIR}/access-pablo-es.log combined
CustomLog /var/www/pablo.es/access.log combined
```

Desactivamos el sitio anterior:

```
miadmin@PCB-USED:/etc/apache2/sites-available$ sudo a2dissite 000-default.conf
Site 000-default disabled.
```

Activamos los sitios creados:

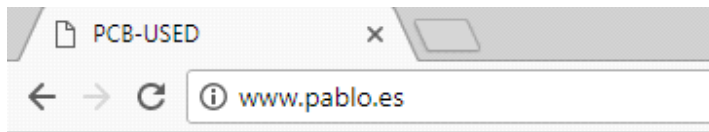
```
miadmin@PCB-USED:/etc/apache2/sites-available$ sudo a2ensite pablo-com.conf
Enabling site pablo-com.
```

```
miadmin@PCB-USED:/etc/apache2/sites-available$ sudo a2ensite pablo-es.conf
Enabling site pablo-es.
```

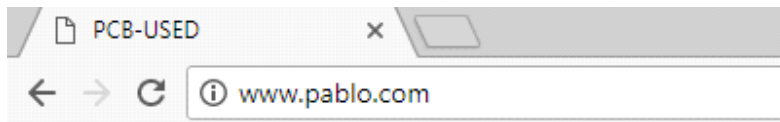
Reiniciamos el servicio:

```
miadmin@PCB-USED:/etc/apache2/sites-available$ sudo service apache2 reload
```

Comprobamos que funciona:



## PÁGINA DE PABLO.ES



## PÁGINA DE PABLO.COM

### Generación de Certificados

En primer lugar generamos la clave privada:

```
miadmin@PCB-USED:~$ openssl genrsa 2048 > clavePrivada.key
Generating RSA private key, 2048 bit long modulus
.....+++
.....+++
e is 65537 (0x10001)
```

```
miadmin@PCB-USED:~$ ls
clavePrivada.key
```

```
miadmin@PCB-USED:~$ cat clavePrivada.key
-----BEGIN RSA PRIVATE KEY-----
MIIEpQIBAAKCAQEAs/N+VdUn1mBI1HVoX8CoyZERZB4EUWktckgLAGXA9iZRpvdo
Q+EsuUzd14Uj1A9YrFXWau6SNB0jru42stt15CS4nXpdELz9oRp4JPB6SSqn8+qn
8hHESugb/rB/WRqtKUYbopv0jwxhZu5GOL+Jkf8maPyGwbSONUMdwx8P/7q/P1jL
cHN7XkPa11bQIdWtYg6iafeJS15Tp56UisUpjuTIPWe8A+n3s0SE2RPHM8/j/kvfA
uKDwad2RfG/TH9fRU+Ctuy9WetUsOLp2jP++4P1Sm3oOP2uv1XM0paTVzsanQrBI
QdLiqqxvQ2xi/v0uCapVsRViRUFcXJux12pTgQIDAQABaoIBAQCfYH4ED2kRMqB
QPACpko/n9hHT76Mts2ffSchRv3MBMw0kNMrg3ciVf28gRLmcnEa4TsWha2/sj80
Y3UnT+HCWj0Y2hOfC40m+AAI00D9gaYXBjGQ1+N+dM9M141yoYnUHouDTXN4hoKN
42MWR63UPKMMQPwo+30oBwFku2w0FPeDUa3uLws0vvuJrD2fxwbi4QIO7ActFI19
U5iqN0D951VUNp30XC4ua6eo4ODPMkmDsv7ZUx9Spcna5nr70mAFU7fp23p23jrm
OpHJYiOwgeEqkHCzhD+SyCpo04jyQC0rCYKAjGrsU7JyK0eppxIH7o68+/ixYs8S
u18tsFXxaoGBA06vdDdNpiXDEAbGifL79AgcEj3Q4b,jic30g2YjFVLF0cEb6skq
UjWo4wtEmP91hcJ+UgkfQBxldXxs2XWxoy4nULYe+EcG/Gq7+HMqU9zPn9duU27p
+Mei1JSQGpIm0qZN+BMJhuV60h3AZu2vjuJPk+YGj1AggC/pMDzsyx1HAoGBAMEB
TcaoDP8g1qai14SoHB9N0Shy80Qv2+PrwB12T5oT200Su4h50w8XDThxKevI2Xy9
tVdcmmjyugSGMq/qfGzWs3B8HBWrc1fs+KQUrteYmkgWv+Jxfr+pguAfW6ZECvDM
rbRI3gNiisdkhYMP0YijbJ2AL0iKIGHeDt3QEz3AoGBA175SDSY7bJtmcoFOXLO
m3wUJJUrLeXp/I0DL4aJ82buPyc+qsgvdQ6UINMiFyfG5QX2GX5DXSBU1fGfXi+S
5bUq1X0SvHrtydvB1xSdQTjeodew/+yveFdU0v02yMPNP5dg0v3MUika5pftvXoZ
E1+0///k5AnY04r3eu6zxQNXAoGA1W0XgggUwkmHqumLkArHgAKREYhDF5oTTtx
dPbUsIXe9QVhEK6FggcDctGju/x0KPNTgEG2BJGTxQcdnuWns+82ADJ7agTTYITS
dRSLvujXJDsApVrjuh55/20uUoaMFgcjsCFjU1SFpPDhaI+UuwAgJs7hIdvYF2Srm
xvxxU0ECgYEAE5H2rsdrhTxquDehxUoSAYU+sz+eXBIGHEi0QEMT16T16QRHst7RK
ElwGp6e,jYHdp1KLv9USS1xFyWZdnzyD00MCtkn5SmpaKob2C0SRrQTPEX91CS4Y
4MXz/jHvk0yzsLDJrC66HpxmT/b0BYdDRYau1s4AUuUOIjYJW3Md/x4=
-----END RSA PRIVATE KEY-----
```

A continuación generamos la solicitud de certificado:

Pablo Cidón Barrio

```
miadmin@PCB-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.
```

```
miadmin@PCB-USED:~$ cat certificado.csr
-----BEGIN CERTIFICATE REQUEST-----
MIICHzCCAW8CAQAwQjELMAkGA1UEBhMCRCVMDzANBgNVBAGMB1phbW9yYTESMBAG
A1UEBwwJQmUuYXZlbnRlMQ4wDAYDVQQDDAVQYWJsbnzCCASlwdQYJKoZIhvcNAQEB
BQADggEPADCCAQoCggEBAMCSUT2r5d0WH/ymsPR+okobkYaysBI7cEnP5rvy+IIN
0wlmLEBNPJaNWXDcuDQL90qbNIIYHEu503F/JICgye1x/CSKHu/9jUba30HysTKN
b7BLoYXjQa3mP4b0H4XY0VXCvhrYf06hKkaImpGkVfsJ80cVFlss/Y34R/WVZ8FJ
9FomKZKTdBYE3+6a9B3EPkAYoYS1JStaKsCjagudVzFaXufLQY10CMxVREoKFOD5
27rzVKLWISysXiJzQupAnLth5o86ELj50dQp71TY/ZoVBuxeL/Y8tvKHLi+Ccypw
HCzFxLOhJQj2Sdogz+vo4IS9PmME/C8lalasreStkrUCAwEAaAAMA0GCSqGSIb3
DQEBcwUAA4IBAQBFGMKIN17rJU2dCtjpu0Y4UDLcVQH3jcwRwzWE/fbx51tjeJpD
z+dy+7twg3JcygkhiX4v4m9EH9KCDcgoKGGqFNZdKef7IIQKZ0o22o260lxwLmkP
Mq6pg3CMbcXw/XHXFULQW7RWIdJ+kWl9ugaeNdv4gc5TUoT1EvmXJlIBZ8ZFT1SN
v0FwWswNcgufq1pTvfXTvPcG1iMR2FxD4YbSoDsWv8kA+AHU3WWFgYBeYyA2qe06
ZDjAdfjYRQFRAtDM//n2chmM/9nG0qrxRw9oDqD6Nqg/wKCcdBFGW3jzw6waaStF
DlaB05vhqenMHv/LnPyhcdbrXIHU7eqGc0e
-----END CERTIFICATE REQUEST-----
```

Una vez hecho, creamos el certificado autofirmado usando la clave privada:

```
miadmin@PCB-USED:~$ openssl x509 -req -days 365 -in certificado.csr -signkey clavePrivada.key > cert
ificadoFirmado.crt
Signature ok
```

```
miadmin@PCB-USED:~$ cat certificadoFirmado.crt
-----BEGIN CERTIFICATE-----
MIIDADCCAegCCQDTfG+MptHA0jANBgkqhkiG9w0BAQsFAADBCMQswCQYDVQQGEwJF
UzEPMA0GA1UECAwGdFb3JhMRlweAYDVQQHDA1CZW5hdmUudGUxZjAMBgNVBAMM
BUBhYmUuYXZlbnRlMQ4wDAYDVQQDDAVQYWJsbnzCCASlwdQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAMCSUT2r
5d0WH/ymsPR+okobkYaysBI7cEnP5rvy+IIN0wlmLEBNPJaNWXDcuDQL90qbNIIY
HEu503F/JICgye1x/CSKHu/9jUba30HysTKNb7BLoYXjQa3mP4b0H4XY0VXCvhrY
f06hKkaImpGkVfsJ80cVFlss/Y34R/WVZ8FJ9FomKZKTdBYE3+6a9B3EPkAYoYS1
JStaKsCjagudVzFaXufLQY10CMxVREoKFOD527rzVKLWISysXiJzQupAnLth5o86
ELj50dQp71TY/ZoVBuxeL/Y8tvKHLi+CcypwHCzFxLOhJQj2Sdogz+vo4IS9PmME
/C8lalasreStkrUCAwEAATANBgkqhkiG9w0BAQsFAAOCAQEAtCDxDUlNyBxNqi62
qQvpUZKf4Pkekf1la22j31a2NFAyxdAq7nbmraY7y0eeNIZluSVkV0DzUK57Qa81
TcLr10aiI7GKMtdVcrJrJESBpfTw977t1GL4cbBGz7RVTC3Nk39jJf0PMYlvUae/
Zx01XIE+JlQSKKEgLN7hmm5jeOCiUvN6Gwer5wUUQZcrryqhniqI6iCbvWf7vxAY
30dFf4q99q98Qu2UxvWhaofOLjhsrF15x10YRkI2E8sIfGANF0uu1KP9ICA9uAuB
3/h2KvQ6GIXvHJ1Ac9LdqDhxbDe2xzIydao6KDO6VPtOzjVHnuc00etytweegffM
KmOgw==
-----END CERTIFICATE-----
```

Ahora cambiamos los permisos del fichero.key y el propietario:

```
miadmin@PCB-USED:~$ sudo chown root:ssl-cert clavePrivada.key
miadmin@PCB-USED:~$ sudo chmod 640 clavePrivada.key
miadmin@PCB-USED:~$ sudo chown root:root certificadoFirmado.crt
miadmin@PCB-USED:~$ sudo mv clavePrivada.key /etc/ssl/private
miadmin@PCB-USED:~$ sudo mv certificadoFirmado.crt /etc/ssl/certs/
```

Copiamos el fichero de configuración SSL por defecto para la configuración de los sitios:

```
miadmin@PCB-USED:/etc/apache2/sites-available$ sudo cp default-ssl.conf pablo-com-ssl.conf
miadmin@PCB-USED:/etc/apache2/sites-available$ sudo cp default-ssl.conf pablo-es-ssl.conf
```

Una vez hecho, configuramos los sitios:

En la dirección.com

```
ServerAdmin webmaster@localhost
ServerName www.pablo.com
ServerAlias pablo.com
DocumentRoot /var/www/pablo.com
```

```
SSLCertificateFile /etc/ssl/certs/certificadoFirmado.crt
SSLCertificateKeyFile /etc/ssl/private/clavePrivada.key
```

En la dirección.es:

Copiamos el fichero.com:

```
miadmin@PCB-USED:/etc/apache2/sites-available$ sudo cp pablo-com-ssl.conf pablo-es-ssl.conf
```

Configuramos el fichero:

```
ServerAdmin webmaster@localhost
ServerName www.pablo.es
ServerAlias pablo.es
DocumentRoot /var/www/pablo.es/public_html
```

Y activamos los sitios:

```
miadmin@PCB-USED:/etc/apache2/sites-available$ sudo a2ensite pablo-com-ssl.conf
```

```
miadmin@PCB-USED:/etc/apache2/sites-available$ sudo a2ensite pablo-es-ssl.conf
Enabling site pablo-es-ssl.
```

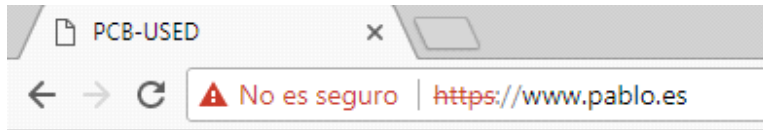
Comprobamos que funciona:

En la dirección.com:



# PÁGINA DE PABLO.COM

En la dirección.es

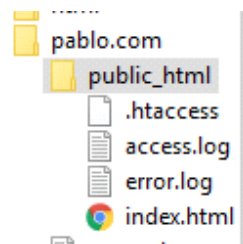


## PÁGINA DE PABLO.ES

### Redireccionar de http a https:

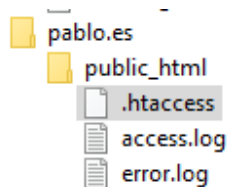
Creamos el fichero .htaccess dentro del document root y añadimos lo siguiente:

Dirección.com:



```
RewriteEngine On
RewriteCond %{SERVER_PORT}80
RewriteRule ^(.*)$ https://www.pablo.com/$1 [R,L]
```

Dirección.es:



```
RewriteEngine On
RewriteCond %{SERVER_PORT}80
RewriteRule ^(.*)$ https://www.pablo.es/$1 [R,L]
```

En caso de que el módulo rewrite no esté activo, tendremos que activarlo:

```
miadmin@PCB-USED:/etc/apache2/mods-available$ sudo a2enmod rewrite.load
Module rewrite already enabled
```

Comprobamos que funciona:



## UD 4 Servidor de Nombres de Dominio (DNS)

### Actividad 01

#### Instalación del Servidor DNS

En primer lugar realizamos la actualización del repositorio:

```
miadmin@PCB-ED:~$ sudo apt-get update
[sudo] password for miadmin:
```

Luego instalamos el servidor DNS

```
miadmin@PCB-ED:~$ sudo apt-get install bind9
```

Una vez instalado, realizamos la configuración de red, en la que añadiremos como servidor DNS nuestra propia IP.

```
miadmin@PCB-ED:~$ sudo nano /etc/network/interfaces
```

```
iface enp0s3 inet static
address 192.168.1.150
gateway 192.168.1.1
netmask 255.255.255.0
dns-nameservers 192.168.1.150
dns-domain pablo.local
```

Iniciamos el servicio bind9:

```
miadmin@PCB-ED:~$ sudo service bind9 start
```

Comprobamos el servicio bind9:

```
miadmin@PCB-ED:~$ sudo service bind9 status
● bind9.service - BIND Domain Name Server
   Loaded: loaded (/lib/systemd/system/bind9.service; enabled; vendor preset: enabled)
   Drop-In: /run/systemd/generator/bind9.service.d
            └─50-insserv.conf-$named.conf
   Active: active (running) since vie 2018-01-26 15:54:06 CET; 8h ago
     Docs: man:named(8)
   Main PID: 977 (named)
    Tasks: 4
   Memory: 12.3M
      CPU: 9min 37.347s
   CGroup: /system.slice/bind9.service
            └─977 /usr/sbin/named -f -u bind
```

## Configurar reenviadores

Vamos al fichero de configuración:

```
miadmin@PCB-ED:~$ sudo nano /etc/bind/named.conf.options
```

Cambiamos el forwarder:

```
forwarders {  
    8.8.8.8;  
};
```

Reiniciamos el Servicio:

```
miadmin@PCB-ED:~$ sudo service bind9 restart
```

## Agregar a un dominio

Vamos al fichero de interfaces:

```
miadmin@PCB-ED:~$ sudo nano /etc/network/interfaces
```

Añadimos lo siguiente:

```
-----  
dns-domain pablo.local
```

Reiniciamos la máquina:

```
miadmin@PCB-ED:~$ sudo reboot now
```

Una vez hecho comprobamos el fichero resolv.conf:

```
miadmin@PCB-ED:~$ cat /etc/resolv.conf  
# Dynamic resolv.conf(5) file for glibc r  
# DO NOT EDIT THIS FILE BY HAND -- YO  
nameserver 192.168.1.150  
search pablo.local
```

## Configuración de un servidor DNS master:

Configuramos las zonas en el fichero named.conf.local:

```
miadmin@PCB-ED:~$ sudo nano /etc/bind/named.conf.local
```

Configuramos la zona directa:

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

Configuración de la zona inversa:

```
zone "1.168.192.in-addr.arpa"{  
    type master;  
    file "/etc/bind/db1.168.192.in-addr.arpa";  
};
```

Copiamos el fichero bind.0:

```
miadmin@PCB-ED:~$ sudo cp /etc/bind/db.0 /etc/bind/db.pablo.local
```

Una vez copiado lo modificamos:

```
$TTL      604800  
@         IN      SOA      PCB-ED.pablo.local. root.localhost. (  
                                1          ; Serial  
                                604800     ; Refresh  
                                86400      ; Retry  
                                2419200    ; Expire  
                                604800 )   ; Negative Cache TTL  
;  
@         IN      NS       PCB-ED.pablo.local.  
PCB-ED    IN      A        192.168.1.150  
PC-PC     IN      A        192.168.1.101  
ftp       IN      CNAME    PCB-ED.pablo.local.  
www       IN      CNAME    PCB-ED.pablo.local.
```

Una vez realizado comprobamos que está correcto:

```
miadmin@PCB-ED:~$ sudo named-checkzone pablo.local /etc/bind/db.pablo.local  
zone pablo.local/IN: loaded serial 1  
OK
```

Comprobamos que se han aplicado los cambios:

```
miadmin@PCB-ED:~$ sudo nslookup PCB-ED
Server:      192.168.1.150
Address:     192.168.1.150#53

Name:   PCB-ED.pablo.local
Address: 192.168.1.150

miadmin@PCB-ED:~$ sudo nslookup PC-PC
Server:      192.168.1.150
Address:     192.168.1.150#53

Name:   PC-PC.pablo.local
Address: 192.168.1.101
```

## Creación de Alias:

Vamos al fichero de configuración.

```
miadmin@PCB-ED:~$ sudo nano /etc/bind/db.pablo.local
```

Y añadimos los alias:

```
ftp      IN      CNAME    PCB-ED.pablo.local.
www      IN      CNAME    PCB-ED.pablo.local.
```

Chequeamos que esté correcto:

```
miadmin@PCB-ED:~$ sudo named-checkzone pablo.local /etc/bind/db.pablo.local
```

A continuación reiniciamos el servicio:

```
miadmin@PCB-ED:~$ sudo service bind9 restart
```

## Configuración de la zona inversa:

Copiamos el fichero de configuración de nuestra zona:

```
miadmin@PCB-ED:~$ sudo cp /etc/bind/db.pablo.local /etc/bind/1.168.192.in-addr.arpa
```

Configuramos el fichero:

```
$TTL      604800
@         IN      SOA      PCB-ED.pablo.local. root.localhost. (
                        1      ; Serial
                        604800  ; Refresh
                        86400   ; Retry
                        2419200  ; Expire
                        604800 ) ; Negative Cache TTL
;
@         IN      NS       PCB-ED.pablo.local.
101       IN      PTR      PC-PC.pablo.local.
150       IN      PTR      PCB-ED.pablo.local.
```

## Resolución del nombre del dominio:

Vamos al fichero:

```
miadmin@PCB-ED:~$ sudo nano /etc/bind/db.pablo.local
```

Editamos el fichero, en lugar del nombre del dominio también se podría poner un @

```
pablo.local.      A      192.168.1.150
```

Reiniciamos el servicio:

```
miadmin@PCB-ED:~$ sudo service bind9 restart
```

Comprobamos que funciona:

```
miadmin@PCB-ED:~$ nslookup pablo.local
Server:      192.168.1.150
Address:     192.168.1.150#53

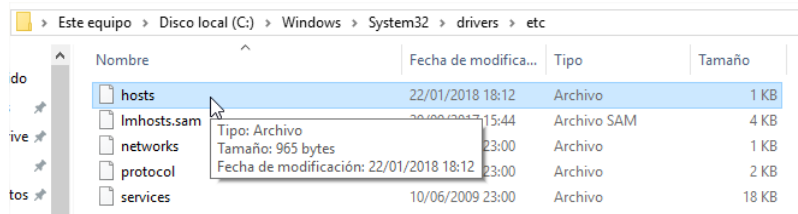
Name:   pablo.local
Address: 192.168.1.150
```

## Actividad 02:

[http://daw-usgit.sauces.local/PCB\\_1718/DAW/blob/master/bind9/README.md](http://daw-usgit.sauces.local/PCB_1718/DAW/blob/master/bind9/README.md)

## Actividad 03:

Vamos al fichero hosts del cliente:



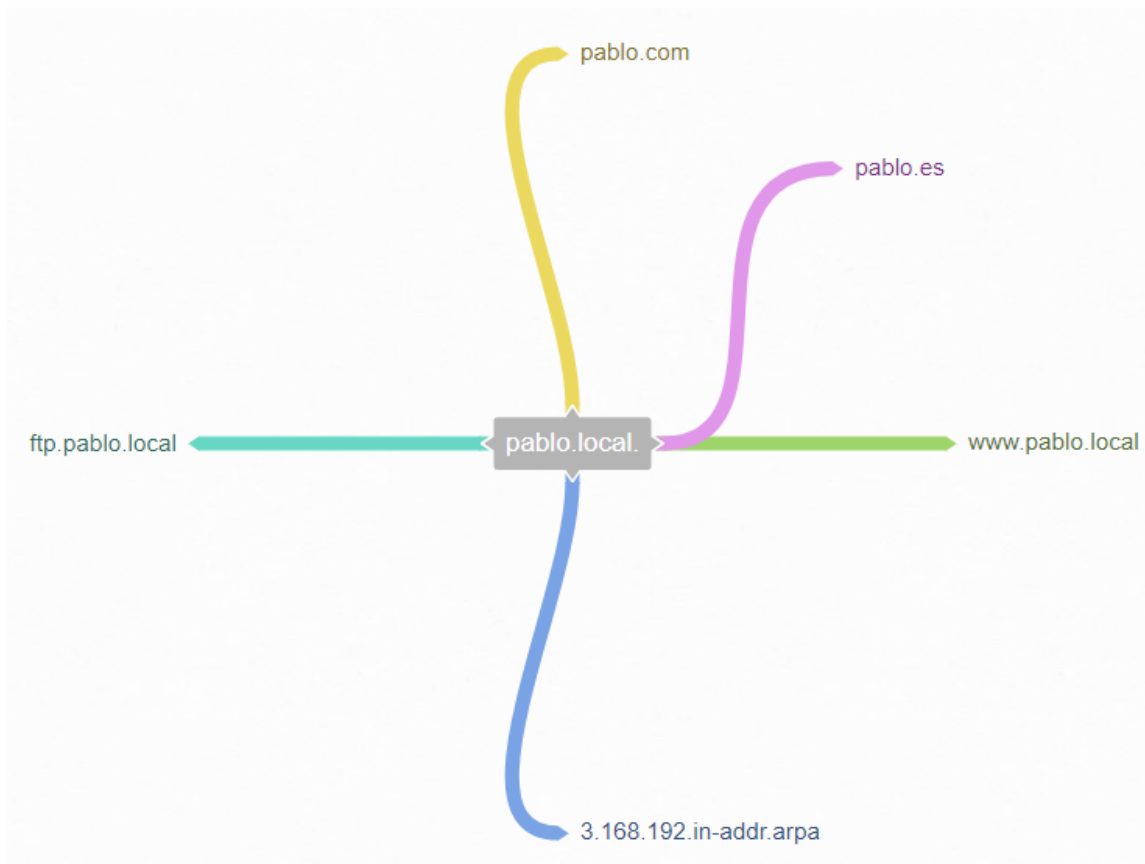
Añadimos los nombres de acceso para esa IP:

```
192.168.1.150 www.pablo.com pablo.com
192.168.1.150 www.pablo.es pablo.es
```

Guardamos cambios y comprobamos que funciona:



## Actividad 04:



## UD 5 FTP

### Instalación y configuración de ProFTP:

Instalamos el servidor proftpd:

```
miadmin@PCB-Nube:~$ sudo apt-get install proftpd
```

Seleccionamos la opción de ejecución, en este caso seleccionamos la opción inetd, que es para pocas conexiones:

Ejecutar proftpd:

desde inetd  
independiente

<Aceptar>

Comprobamos que se han creado los usuarios proftpd por defecto, se encuentran en "/bin/false" para que estos usuarios no puedan loguearse en el sistema:

```
miadmin@PCB-Nube:~$ cat /etc/passwd
proftpd:x:112:65534:./run/proftpd:/bin/false
ftp:x:113:65534:./srv/ftp:/bin/false
```

Comprobamos a que grupos pertenecen:

```
miadmin@PCB-Nube:~$ cat /etc/group
nogroup:x:65534:
```

Comprobamos el fichero de configuración:

```
miadmin@PCB-Nube:~$ cat /etc/proftpd/proftpd.conf
```

Los usuarios del sistema pueden conectarse desde ftp con el cliente y les permite moverse a través de los directorios, en cambio el usuario anónimo no puede conectarse.

## Enjaular usuarios:

Vamos al fichero de configuración:

```
miadmin@PCB-Nube:~$ sudo nano /etc/proftpd/proftpd.conf
```

Comentamos el DefaultRoot:

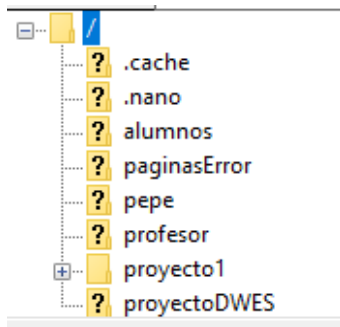
```
# Use this to jail all users in chroot.
DefaultRoot ~
```

Reiniciamos el servicio:

```
miadmin@PCB-Nube:~$ sudo service proftpd restart
```

Comprobamos que funciona y observamos como el usuario no puede acceder nada más que a su directorio raíz, en este caso "/var/www/html":





## Asignación de los permisos:

Por defecto Umask es 022, y lo que realiza es restar 022 a 666 en caso de que sea un archivo y a 777 en caso de que sea un directorio y por lo tanto los permisos serían 664 y 755.

Por eso es necesario que se cambie la máscara por 002, en el fichero de configuración:

```
miadmin@PCB-Nube:~$ sudo nano /etc/proftpd/proftpd.conf
```

```
Umask                                002  002
```

Reiniciamos el servicio:

```
miadmin@PCB-Nube:~$ sudo service proftpd restart
```

Comprobamos los cambios:

Antes de los cambios:

	index.html	721	Chrome H...	29/09/2017 13:39:22	adfrw (0644)
	profesor		Carpeta de...	08/11/2017 9:19:31	flcdmpe (0755)

Aplicados los cambios:

	index.html	134	Chrome H...	05/02/2018 18:...	adfr (0644)	0 0
--	------------	-----	-------------	-------------------	-------------	-----

## Usuarios que tienen acceso al sistema:

Fichero de usuarios que no pueden entrar en el sistema FTP, para ello únicamente es necesario añadir el usuario al archivo:

```
miadmin@PCB-Nube:~$ cat /etc/ftpusers
# /etc/ftpusers: list of users disallc

root
daemon
bin
sys
sync
games
man
lp
mail
news
uucp
nobody
```

## Configuración del usuario anonymous ftp:

Tenemos que acceder al fichero de configuración:

```
miadmin@PCB-Nube:~$ sudo nano /etc/proftpd/proftpd.conf
```

Descomentamos las siguientes líneas:

```
<Anonymous ~ftp>
  User ftp
  Group nogroup
  # We want clients to be able to login with "anonymous" as well as "ftp"
  UserAlias anonymous ftp
  # Cosmetic changes, all files belongs to ftp user
  DirFakeUser on ftp
  DirFakeGroup on ftp

  RequireValidShell off

  # Limit the maximum number of anonymous logins
  MaxClients 10

  # We want 'welcome.msg' displayed at login, and '.message' displayed
  # in each newly chdired directory.
  DisplayLogin welcome.msg
  DisplayChdir .message

  # Limit WRITE everywhere in the anonymous chroot
  <Directory *>
    <Limit WRITE>
      DenyAll
    </Limit>
  </Directory>

</Anonymous>
```

Reiniciamos el servicio:

```
miadmin@PCB-Nube:~$ sudo service proftpd restart
```

Comprobamos que funciona, para ello en la conexión debemos introducir "anonymous" como usuario y un correo como contraseña o algo que contenga "@".

Servidor:	192.168.1.250	Nombre de usuario:	anonymous	Contraseña:	...	Puerto:	21	Enter
-----------	---------------	--------------------	-----------	-------------	-----	---------	----	-------

También podremos observar que no puede acceder a ningún archivo.

Sitio remoto:	/

Para eso es necesario saber cuál es el directorio donde se encuentra el usuario ftp, que es el que corresponde con anonymous, para ello debemos acceder al fichero `/etc/passwd`.

```
miadmin@PCB-Nube:~$ cat /etc/passwd
profptd:x:112:65534::/run/proftpd:/bin/false
ftp:x:113:65534::/srv/ftp:/bin/false
```

En este directorio es en el que tendremos que almacenar los archivos a los que va a acceder dicho usuario.

```
miadmin@PCB-Nube:~$ sudo nano /srv/ftp/index.html
```

```
!DOCTYPE html>
<html lang="es">
  <head>
    <title>ANONYMOUS</title>
  </head>
  <body>
    <h1>PÁGINA DE ANONYMOUS</h1>
  </body>
</html>
```

Comprobamos que funciona:

index.html	134	Chrome H...	05/02/2018 18:...	adfr (0644)	0 0
welcom...	170	Elemento ...	05/04/2016 15:...	adfr (0644)	0 0

## Control de acceso:

En primer lugar tendremos que crear un grupo para los usuarios:

```
miadmin@PCB-Nube:~$ sudo groupadd grupoftp
```

Comprobamos que el grupo se ha creado.

```
miadmin@PCB-Nube:~$ cat /etc/group|grep grupo*
grupoftp:x:1001:
```

Ahora crearemos el primer usuario:

```
miadmin@PCB-Nube:~$ sudo useradd -g www-data -G grupoftp -m -d /var/www/daw201 -p paso daw201
```

Comprobamos que se ha creado el usuario:

```
miadmin@PCB-Nube:~$ cat /etc/passwd|grep daw201
daw201:x:1001:33::/var/www/daw201:
```

Comprobamos que se ha creado el home del usuario:

```
miadmin@PCB-Nube:~$ ls /var/www
daw201  html
```

Para que el usuario no pueda iniciar sesión en el sistema crearemos un directorio para el shell.

```
miadmin@PCB-Nube:~$ sudo mkdir /bin/ftp
```

Luego tendremos que añadir este directorio al fichero en el que se encuentran los shells.

```
miadmin@PCB-Nube:~$ sudo nano /etc/shells
```

```
# /etc/shells: valid login shells
/bin/sh
/bin/dash
/bin/bash
/bin/rbash
/bin/ftp
/usr/bin/tmux
/usr/bin/screen
```

Si no se ha pasado este shell como parámetro al crear el usuario, será necesario editar el fichero, donde tendremos que añadirse al usuario.

```
miadmin@PCB-Nube:~$ sudo nano /etc/passwd
```

```
daw201:x:1001:33::/var/www/daw201:/bin/ftp
```

Finalmente comprobaremos que funciona correctamente.

Servidor:	192.168.1.250	Nombre de usuario:	daw201	Contraseña:	••••	Puerto:		Conexión rápida	▼
Estado:	Conectando a 192.168.1.250:21...								
Estado:	Conexión establecida, esperando el mensaje de bienvenida...								
Estado:	Servidor no seguro, no soporta FTP sobre TLS.								
Comando:	USER daw201								
Respuesta:	331 Contraseña necesaria para daw201								
Comando:	PASS ****								

## Limitar el acceso a determinados usuarios:

Ahora vamos a limitar el acceso a una serie de usuarios, como son varios, lo haremos limitando el acceso a un grupo, al cuál pertenecen aquellos usuarios que se vayan a conectar como ftp.

```
miadmin@PCB-Nube:~$ sudo nano /etc/proftpd/proftpd.conf
```

Añadiremos las siguientes líneas:

```
<Limit LOGIN>
    AllowGroup grupoftp
    DenyAll
</Limit>
```

Reiniciamos el servicio y comprobamos:

```
miadmin@PCB-Nube:~$ sudo service proftpd restart
```

Como el usuario daw201 pertenece al grupoftp vemos como puede conectarse.

Servidor:	192.168.1.250	Nombre de usuario:	daw201	Contraseña:	••••
Estado:	Conectando a 192.168.1.250:21...				
Estado:	Conexión establecida, esperando el mensaje de bienvenida...				
Estado:	Servidor no seguro, no soporta FTP sobre TLS.				
Comando:	USER daw201				
Respuesta:	331 Contraseña necesaria para daw201				
Comando:	PASS ****				

En cambio, el usuario operadorweb no pertenece al grupoftp por lo que no podrá conectarse.

Servidor:	192.168.1.250	Nombre de usuario:	operadorweb	Contraseña:	••••
Estado:	Servidor no seguro, no soporta FTP sobre TLS.				
Comando:	USER operadorweb				
Respuesta:	331 Contraseña necesaria para operadorweb				
Comando:	PASS ****				
Respuesta:	530 Login incorrecto				
Error:	Critical error: No se pudo conectar al servidor				

## Usuarios y dominios virtuales

Vamos al fichero de configuración:

Pablo Cidón Barrio

```
miadmin@PCB-Nube:~$ sudo nano /etc/passwd
```

Descomentamos la siguiente línea, para que lea las zonas virtuales:

```
#include /etc/proftpd/virtuals.conf
```

Reiniciamos el servicio:

```
miadmin@PCB-Nube:~$ sudo service proftpd restart
```

A continuación crearemos los usuarios, para ello tendremos que saber el uid del usuario ftp y el del grupo www-data:

```
miadmin@PCB-Nube:~$ sudo ftpasswd --passwd --name ucom --file /etc/passwd.usuari  
os.pablo.com --uid 113 --gid 33 --home /var/ftp/pablo.com --shell /bin/false
```

```
miadmin@PCB-Nube:~$ sudo ftpasswd --passwd --name ucom --file /etc/passwd.usuari  
os.pablo.es --uid 113 --gid 33 --home /var/ftp/pablo.es --shell /bin/false
```

Comprobamos que se han creado:

```
miadmin@PCB-Nube:~$ ls /etc/passwd.*  
/etc/passwd.usuarios.pablo.com /etc/passwd.usuarios.pablo.es  
miadmin@PCB-Nube:~$
```

Ahora vamos al fichero en el que se encuentran las zonas virtuales:

```
miadmin@PCB-Nube:~$ sudo nano /etc/proftpd/virtuals.conf
```

Añadimos las zonas virtuales:

```
<VirtualHost ftp.pablo.com>  
ServerName "Servidor FTP pablo.com"  
DefaultRoot ~  
Port 2122  
AuthUserFile /etc/passwd.usuarios.pablo.com  
</VirtualHost>  
<VirtualHost ftp.pablo.es>  
ServerName "Servidor FTP pablo.es"  
DefaultRoot ~  
Port 2121  
AuthUserFile /etc/passwd.usuarios.pablo.es  
</VirtualHost>
```

Ahora tenemos que añadir el shell al fichero:

```
miadmin@PCB-Nube:~$ sudo nano /etc/shells
```

```
/bin/false
```

Creamos los home de los usuarios:

```
miadmin@PCB-Nube:~$ sudo mkdir /var/ftp
miadmin@PCB-Nube:~$ sudo mkdir /var/ftp/pablo.com
miadmin@PCB-Nube:~$ sudo mkdir /var/ftp/pablo.es
```

Cambiamos el propietario de los ficheros:

```
miadmin@PCB-Nube:~$ sudo chown -R ftp:www-data /var/ftp
```

Tendremos que comentar las siguientes líneas del fichero de configuración de proftpd:

```
miadmin@PCB-Nube:~$ sudo nano /etc/proftpd/proftpd.conf

#<Limit LOGIN>
#     AllowGroup grupoftp
#     DenyAll
#</Limit>
```

## Control de Cuotas:

Vamos al siguiente fichero:

```
miadmin@PCB-Nube:~$ sudo nano /etc/proftpd/modules.conf
```

Allí comprobaremos si están activadas las cuotas de disco:

```
LoadModule mod_ctrls_admin.c
LoadModule mod_tls.c
```

Como están activas, iremos al fichero de configuración donde estableceremos las cuotas:

```
miadmin@PCB-Nube:~$ sudo nano /etc/proftpd/proftpd.conf
```

Ahora iremos a la directiva IfModule para su configuración. Por defecto aparecerá desactivada.

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

A continuación procederemos su configuración:

```
<IfModule mod_quotatab.c>
QuotaEngine on
QuotaLog /var/log/proftpd/quota.log
<IfModule mod_quotatab_file.c>
QuotaLimitTable file:/etc/proftpd/ftpquota.limittab
QuotaTallyTable file:/etc/proftpd/ftpquota.tallytab
</IfModule>
</IfModule>
```

Una vez hecho, procederemos a la creación de los ficheros que almacenarán las tablas:

```
miadmin@PCB-Nube:~$ sudo ftpquota --create-table --type=limit --table-path=/etc/
proftpd/ftpquota.limittab
```

```
miadmin@PCB-Nube:~$ sudo ftpquota --create-table --type=tally --table-path=/etc/
proftpd/ftpquota.tallytab
```

Comprobamos que se han creado los ficheros:

```
miadmin@PCB-Nube:~$ ls /etc/proftpd/ftpquota.*
/etc/proftpd/ftpquota.limittab /etc/proftpd/ftpquota.tallytab
```

Una vez hecho, podremos realizar la limitación de las cuotas:

```
miadmin@PCB-Nube:~$ sudo ftpquota --add-record --type=limit --name=ucom --quota-type
=user --bytes-upload=5 --units=Mb --table-path=/etc/proftpd/ftpquota.limittab
miadmin@PCB-Nube:~$ sudo ftpquota --add-record --type=limit --name=ues --quota-type=
user --bytes-upload=2 --units=Mb --table-path=/etc/proftpd/ftpquota.limittab
```

Ahora comprobamos que se han creado las cuotas:

```
miadmin@PCB-Nube:~$ sudo ftpquota --show-records --type=limit --table-path=/etc/pr
oftpdp/ftpquota.limittab
```

Name: ues	Name: ucom
Quota Type: User	Quota Type: User
Per Session: False	Per Session: False
Limit Type: Hard	Limit Type: Hard
Uploaded bytes: 2097152.00	Uploaded bytes: 5242880.00
Downloaded bytes: unlimited	Downloaded bytes: unlimited
Transferred bytes: unlimited	Transferred bytes: unlimited
Uploaded files: unlimited	Uploaded files: unlimited
Downloaded files: unlimited	Downloaded files: unlimited
Transferred files: unlimited	Transferred files: unlimited

## Recargar la configuración del servidor:

Iniciamos el servicio proftpd:

```
miadmin@PCB-Nube:~$ sudo systemctl status proftpd.service
```



Realizamos un listado de los puertos:

```
miadmin@PCB-Nube:~$ netstat -ltn
Conexiones activas de Internet (solo servidores)
Proto Recib Enviad Dirección local      Dirección remota      Estado
tcp    0      0 127.0.0.1:3306      0.0.0.0:*             ESCUCHAR
tcp    0      0 0.0.0.0:22          0.0.0.0:*             ESCUCHAR
tcp6   0      0 :::80               :::*                   ESCUCHAR
tcp6   0      0 :::22               :::*                   ESCUCHAR
miadmin@PCB-Nube:~$
```

## Documentación de las cuotas del disco.

```
miadmin@PCB-Nube:~$ du -shc /var/www/*
16K    /var/www/daw201
16K    /var/www/html
32K    total
```

## Acceso seguro mediante TLS

Vamos al fichero de configuración:

```
miadmin@PCB-Nube:~$ sudo nano /etc/proftpd/proftpd.conf
```

Descomentamos la siguiente línea:

```
include /etc/proftpd/tls.conf
```

Instalamos openssl, aunque puede que ya esté instalado:

```
miadmin@PCB-Nube:~$ sudo apt-get install openssl
```

Luego generaremos la clave privada y el certificado:

```
miadmin@PCB-Nube:~$ sudo openssl req -x509 -newkey rsa:1024 -keyout /etc/ssl/private/proftpd.key -out /etc/ssl/certs/proftp.crt -nodes -days 3650
```

A continuación cambiamos los permisos:

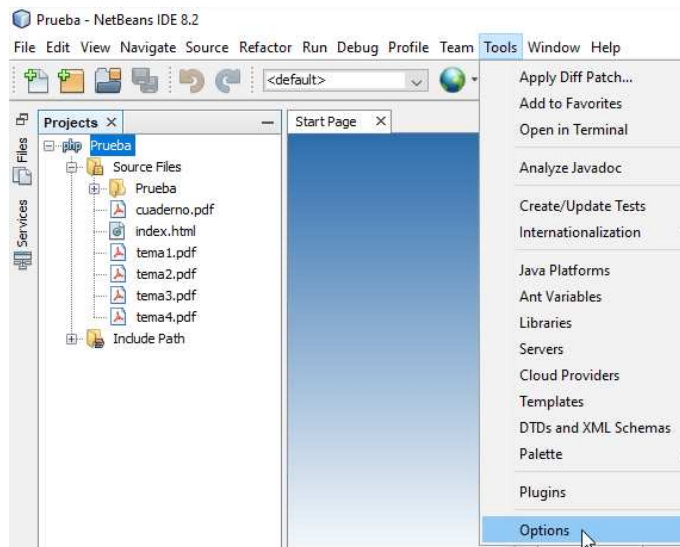
```
miadmin@PCB-Nube:~$ sudo chmod 0600 /etc/ssl/private/proftpd.key
```

```
miadmin@PCB-Nube:~$ sudo chmod 0644 /etc/ssl/certs/proftp.crt
```

## UD 7 Documentación

### NetBeans.

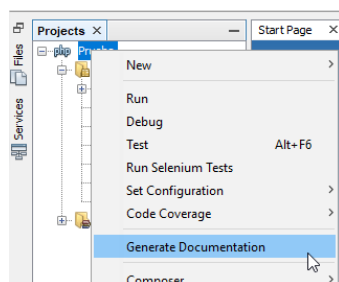
Abrimos NetBeans y vamos a tools dentro de la barra de navegación y luego vamos a opciones:



En opciones vamos a frameworks & tools, en el cuadro introduciremos la ruta en la que se encuentra el archivo phpdoc.php.



Para realizar la documentación vamos con el botón derecho sobre el proyecto, a continuación a generate documentación.



## Línea de comandos.

Otra opción de generar la documentación es a través de línea de comandos. Para ello haremos lo siguiente:

```
E:\php7>php E:\php7\phpDocumentor-2.8.5\phpDocumentor-2.8.5\bin\phpdoc.php -d E:\DWES\Tema7\MiWeb -t E:\DWES\Tema7\doc
```

Donde `E:\php7>php` es el directorio donde se encuentra el phpDocumentor, acompañado del comando php.

Donde `E:\php7\phpDocumentor-2.8.5\phpDocumentor-2.8.5\bin\phpdoc.php` es la ruta en la que se encuentra el ejecutable de phpDoc.

Donde `-d E:\DWES\Tema7\MiWeb` es el directorio del que hay que generar la documentación.

Donde `-t E:\DWES\Tema7\doc` es el directorio de destino de la documentación.

## Doxygen.

Una vez instalado vamos al directorio donde se encuentra y lo ejecutamos.

» Este equipo » SISTEMA (C:) » Archivos de programa » doxygen » bin			
	Nombre	Fecha de modifica...	Tipo
★	doxygen.exe	28/01/2018 17:11	Aplicación
★	doxyindexer.exe	28/01/2018 17:11	Aplicación
★	doxysearch.cgi.exe	28/01/2018 17:11	Aplicación
★	doxywizard.exe	16/08/2017 11:48	Aplicación

Luego establecemos desde donde se va a ejecutar.

Step 1: Specify the working directory from which doxygen will run

C:\Program Files\doxygen\bin

Step 2: Configure doxygen using the Wizard and/or Edit Configuration File

Luego establecemos el nombre del proyecto, el directorio de la documentación y donde se va a almacenar.

Provide some information about the project you are documenting

Project name:

Project synopsis:

Project version or id:

Project logo:  No Project logo selected.

---

Specify the directory to scan for source code

Source code directory:

☐ Scan recursively

---

Specify the directory where doxygen should put the generated documentation

Destination directory:

A continuación seleccionamos la salida de php.

Select the desired extraction mode:

☒ Documented entities only

☐ All Entities

☐ Include cross-referenced source code in the output

Select programming language to optimize the results for

☐ Optimize for C++ output

☐ Optimize for C++/CLI output

☐ Optimize for Java or C# output

☒ Optimize for C or PHP output

☐ Optimize for Fortran output

☐ Optimize for VHDL output

Previous Next

Luego seleccionamos la opción para generar los diagramas.

Diagrams to generate

☐ No diagrams

☐ Use built-in class diagram generator

☒ Use dot tool from the GraphViz package

Dot graphs to generate

☒ Class diagrams

☒ Collaboration diagrams

☒ Overall Class hierarchy

☒ Include dependency graphs

☒ Included by dependency graphs

☒ Call graphs

☒ Called by graphs

Previous Next

## UD 8 Servicio de directorio OpenLDAP.

### Instalación y configuración de OpenLDAP.

Procedemos a instalar los paquetes:

```
root@PCBUS11:/home/miadmin# apt-get install slapd
```

```
miadmin@PCBUS11:~$ sudo apt-get install ldap-utils
```

Una vez instalado, vamos al fichero de hosts:

```
127.0.0.1    localhost
127.0.1.1    PCBUS11.pablo.local    PCBUS11_
# The following lines are desirable for IPv6 capable hosts
::1         localhost ip6-localhost ip6-loopback
ff02::1     ip6-allnodes
ff02::2     ip6-allrouters
```

Reiniciamos la máquina y comprobamos:

```
miadmin@PCBUS11:~$ sudo reboot now
```

```
miadmin@PCBUS11:~$ hostname -f
PCBUS11.pablo.local
miadmin@PCBUS11:~$ hostname
PCBUS11
```

## Configuración de OpenLDAP:

Una vez hecho, instalamos la librería de NSS:

```
miadmin@PCBUS11:~$ sudo apt-get install libnss-ldap
```

Durante la instalación, realizamos la configuración:

Introducimos la URI, que será la IP del servidor.

**Configuración de ldap-auth-config**

Please enter the URI of the LDAP server to use. This is a string in the form of ldap://<hostname or IP>[:<port>/. ldaps:// or ldapi:// can also be used. The port number is optional.

Note: It is usually a good idea to use an IP address because it reduces risks of failure in the event name service problems.

LDAP server Uniform Resource Identifier:

ldapi:///192.168.3.111

<Aceptar>

Luego introducimos el que será el nombre:

**Configuración de ldap-auth-config**

Please enter the distinguished name of the LDAP search base. Many sites use the components of their domain names for this purpose. For example, the domain "example.net" would use "dc=example,dc=net" as the distinguished name of the search base.

Distinguished name of the search base:

dc=pablo,dc=local

<Aceptar>

Seleccionamos la versión a instalar:

Please enter which version of the LDAP protocol should be used by ldapns. It is usually a good idea to set this to the highest available version.

LDAP version to use:

3  
2

<Aceptar>

Seleccionamos si queremos o no crear una base de datos:

This option will allow you to make password utilities that use pam to behave like you would be changing local passwords.

The password will be stored in a separate file which will be made readable to root only.

If you are using NFS mounted /etc or any other custom setup, you should disable this.

Make local root Database admin:

<Sí>

<No>

Seleccionamos que no queremos identificarnos a la hora de realizar consultas a la base de datos:

Choose this option if you are required to login to the database to retrieve entries.

Note: Under a normal setup, this is not needed.

Does the LDAP database require login?

<Sí>

<No>

Creamos el usuario administrador de LDAP:

Configuración de ldap-auth-config

This account will be used when root changes a password.

Note: This account has to be a privileged account.

LDAP account for root:

cn=admin,dc=pablo,dc=local

<Aceptar>

Y añadimos la contraseña del mismo:

Configuración de ldap-auth-config

Please enter the password to use when ldap-auth-config tries to login to the LDAP directory using the LDAP account for root.

The password will be stored in a separate file /etc/ldap.secret which will be made readable to root only.

Entering an empty password will re-use the old password.

LDAP root account password:

\*\*\*\*

<Aceptar>

## Configuración de la autenticación de clientes:

Ahora vamos a configurar la autenticación de los usuarios, para ello debemos hacer:

```
miadmin@PCBUS11:~$ sudo auth-client-config -t nss -p lac_ldap
```

Comprobamos que está todo correcto, para eso tenemos que ir al siguiente fichero y observar que contiene los siguientes datos:

```
miadmin@PCBUS11:~$ sudo nano /etc/ldap.conf
```

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

# Another way to specify your LDAP server is to provide an
uri ldapi:///192.168.3.111
```

```
rootbinddn cn=admin,dc=pablo,dc=local
```

## Configurar el demonio SLDAP:

Ahora procederemos a la configuración de SLDAP, para ello haremos lo siguiente:

En primer lugar introduciremos el siguiente comando:

```
miadmin@PCBUS11:~$ sudo dpkg-reconfigure slapd
```

Luego nos aparecerá la opción para omitir la configuración, le decimos que no.

```
Configuración de slapd

No se creará la configuración ni la base de datos inicial si habilita esta opción.
¿Desea omitir la configuración del servidor OpenLDAP?

<Sí> <No>
```

Luego aceptamos que el nombre del dominio es el que aparece:

```
El nombre de dominio DNS se utiliza para construir el DN base del directorio LDAP. Por
ejemplo, si introduce «foo.example.org» el directorio se creará con un DN base de «dc=foo,
dc=example, dc=org».

Introduzca el nombre de dominio DNS:
pablo.local

<Aceptar>
```

Luego damos un nombre a la organización:

Introduzca el nombre de la organización a utilizar en el DN base del directorio LDAP.

Nombre de la organización:

pablo

<Aceptar>

Luego introducimos la contraseña del administrador:

Introduzca la contraseña para la entrada de administrador de su directorio LDAP.

Contraseña del administrador:

\*\*\*\*

<Aceptar>

A continuación seleccionamos el motor de la base de datos:

Los motores HDB y BDB utilizan formatos de almacenamiento semejantes, pero HDB permite realizar cambios de nombre de subárboles («subtree renames»). Los dos permiten las mismas opciones de configuración.

Se recomienda utilizar MDB. El motor MDB utiliza un nuevo formato de almacenamiento y requiere menos configuración que BDB o HDB.

En cualquier caso, debe revisar la configuración de la base de datos. Consulte «usr/share/doc/slapt/README.Debian.gz» para más detalles.

Motor de base de datos a utilizar:

BDB  
HDB  
MDB

<Aceptar>

Seguidamente, marcamos todas las opciones como vienen señaladas por defecto.

## Crear la estructura de directorio:

Ahora crearemos el directorio LDIF, podremos crearlo donde lo consideremos conveniente:

```
miadmin@PCBUS11:~$ sudo mkdir /etc/ldif
miadmin@PCBUS11:~$ sudo nano /etc/ldif/base.ldif
```

Dentro de ese fichero añadiremos lo siguiente:

```
dn: ou=usuarios,dc=pablo,dc=local
objectClass: organizationalUnit
ou: usuarios

dn: ou=grupos,dc=pablo,dc=local
objectClass: organizationalUnit
ou: grupos_
```

Luego añadimos esta información a la base de datos:

```
miadmin@PCBUS11:~$ sudo ldapadd -x -D cn=admin,dc=pablo,dc=local -W -f /etc/ldif/base.ldif
```



```
adding new entry "ou=usuarios,dc=pablo,dc=local"
adding new entry "ou=grupos,dc=pablo,dc=local"
```

## Añadir usuarios y grupos:

Creamos un fichero, en el que guardaremos un usuarios, lo haremos donde se encuentran los otros ficheros ldif:

```
miadmin@PCBUS11:~$ sudo nano /etc/ldif/usuarios.ldif
```

```
dn: uid=pcidon,ou=usuarios,dc=pablo,dc=local
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: shadowAccount
uid: pcidon
sn: Cidon
givenName: Pablo
cn: Pablo Cidon
displayName: Juan Lopez
uidNumber: 1100
gidNumber: 10000
userPassword: paso
gecos: Pablo Cidon
loginShell: /bin/bash
homeDirectory: /home/pcidon
shadowExpire: -1
shadowFlag: 0
shadowWarning: 7
shadowMin: 8
shadowMax: 99999
shadowLastChange: 10877
mail: pablo.cidon@sauces.com
postalCode: 49600
o: pablo
initials: PC
```

Una vez hecho, guardamos y ejecutamos el siguiente comando:

```
miadmin@PCBUS11:~$ sudo ldapadd -x -D cn=admin,dc=pablo,dc=local -W -f /etc/ldif/usuarios.ldif
```

```
adding new entry "uid=pcidon,ou=usuarios,dc=pablo,dc=local"
```

Ahora comprobamos que se ha creado el usuario:

```
miadmin@PCBUS11:~$ sudo ldapsearch -xLLL -b dc=pablo,dc=local
```

También crearemos el home del usuario:

```
miadmin@PCBUS11:~$ sudo mkdir /home/pcidon
```

```
drwxr-xr-x  2 root  root  4096 ene 30 10:33 pcidon/
```

Una vez creado el usuario, procederemos a crear el grupo, para ello crearemos el siguiente archivo:

```
miadmin@PCBUS11:~$ sudo nano /etc/ldif/grupos.ldif_
```

Dentro del archivo añadimos lo siguiente:

```
dn: cn=jefes,ou=grupos,dc=pablo,dc=local
objectClass: posixGroup
cn: jefes
gidNumber: 10000
```

Guardamos y añadimos con el siguiente comando:

```
miadmin@PCBUS11:~$ sudo ldapadd -x -D cn=admin,dc=pablo,dc=local -W -f /etc/ldif/grupos.ldif
Enter LDAP Password:
adding new entry "cn=jefes,ou=grupos,dc=pablo,dc=local"
```

Comprobamos que está creado:

```
miadmin@PCBUS11:~$ sudo ldapsearch -xLLL -b dc=pablo,dc=local_
```

## Comprobación de que está todo correcto:

Vamos a comprobar que todos los datos están correctos, en primer lugar lo haremos con el usuario. En primer lugar tendremos que introducir el siguiente comando:

```
miadmin@PCBUS11:~$ sudo ldapsearch -xLLL -b "dc=pablo,dc=local" uid=pcidon
```

Nos aparecerá la información sobre lo que se haya solicitado, ya sea un uid o una organización, etc.

```
dn: uid=pcidon,ou=usuarios,dc=pablo,dc=local
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: shadowAccount
uid: pcidon
sn: Cidon
givenName: Pablo
cn: Pablo Cidon
displayName: Juan Lopez
uidNumber: 1100
gidNumber: 10000
gecos: Pablo Cidon
loginShell: /bin/bash
homeDirectory: /home/pcidon
shadowExpire: -1
shadowFlag: 0
shadowWarning: 7
shadowMin: 8
shadowMax: 999999
shadowLastChange: 10877
mail: pablo.cidon@sauces.com
postalCode: 49600
o: pablo
initials: PC
```

También podremos comprobar el contenido del directorio, para ello introducimos el siguiente comando, y nos aparecerá toda la información.

```
miadmin@PCBUS11:~$ sudo slapcat
```

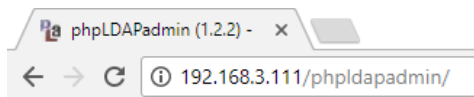
## Interfaz gráfica web: phpLDAPAdmin.

### Instalación de phpLDAPAdmin:

Ahora vamos a instalar un programa para manejar LDAP desde modo gráfico, para ello introducimos el siguiente comando:

```
miadmin@PCBUS11:~$ sudo apt-get install phpldapadmin -y
```

Una vez instalado vamos al navegador y comprobamos que funciona:



### Configuración de phpLDAPAdmin:

Ahora realizaremos la configuración del programa, para ello iremos al siguiente fichero:

```
miadmin@PCBUS11:~$ sudo nano /etc/phpldapadmin/config.php
```

Dentro del fichero, cambiamos el nombre del dominio:

```
$servers->setValue('server','base',array('dc=pablo,dc=local'));
```

```
$servers->setValue('login','bind_id','cn=admin,dc=pablo,dc=local');
```

Una vez realizados los cambios, guardamos y reiniciamos el servicio de apache.

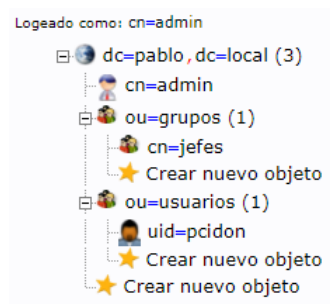
```
miadmin@PCBUS11:~$ sudo service apache2 restart
```

## Creación de usuarios:

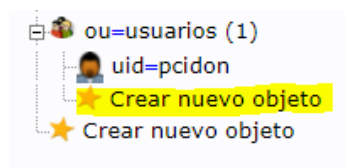
En primer lugar accedemos desde el cliente a phpldapadmin, una vez dentro, iniciamos sesión.



Una vez dentro, podemos observar las unidades organizativas y usuarios que tenemos.



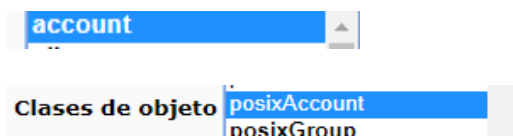
Para crear un nuevo usuario, vamos dentro de los usuarios y luego a crear nuevo objeto.



A continuación, nos aparecerán una serie de plantillas, seleccionaremos la predeterminada.



Luego nos aparecerán las clases de objeto que podemos seleccionar, seleccionaremos `account` y `posixAccount`.



A continuación, nos aparecerán las opciones de configuración para el nuevo usuario, haremos lo siguiente:

Para añadir el identificador del usuario, tendremos que comprobar en el fichero `/etc/passwd` comprobando que el número que se desee poner no esté usado.

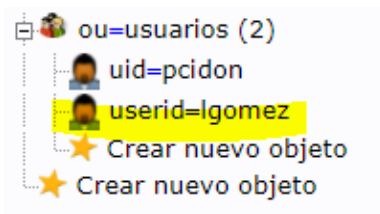
```
miadmin@PCBUS11:~$ cat /etc/passwd | grep 1101
```

Dado que buscando por este número, no nos devuelve ningún registro, podremos asignárselo al nuevo usuario.

Luego, nos aparecerá una tabla con los datos del nuevo registro.

Atributo	Nuevo valor	Omitir
userid=lgomez,ou=usuarios,dc=pablo,dc=local		
cn	Luis Gomez	<input type="checkbox"/>
gidNumber	10000	<input type="checkbox"/>
homeDirectory	/home/lgomez	<input type="checkbox"/>
loginShell	/bin/bash	<input type="checkbox"/>
objectClass	account posixAccount	<input type="checkbox"/>
organizationName	cn=admin,dc=pablo,dc=local	<input type="checkbox"/>
Password	*****	<input type="checkbox"/>
uidNumber	1101	<input type="checkbox"/>
userid	lgomez	<input type="checkbox"/>

Al final, ya nos aparecerá el nuevo usuario.



## Autenticación LDAP en Apache:

En primer lugar debemos comprobar que hay un módulo de ldap en apache.

```
miadmin@PCBUS11:~$ ls /etc/apache2/mods-available | grep ldap
authnz_ldap.load
ldap.conf
ldap.load
```

A continuación habilitamos el módulo:

```
miadmin@PCBUS11:~$ sudo a2enmod authnz_ldap.load
```

Luego, reiniciamos el servicio de apache para aplicar los cambios:

```
miadmin@PCBUS11:~$ sudo service apache2 restart
```

A continuación, activamos .htaccess en el directorio, aunque puede ser que ya esté activado.

```
miadmin@PCBUS11:~$ sudo nano /etc/apache2/apache2.conf
```

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

Finalmente reiniciamos el servicio de apache:

```
miadmin@PCBUS11:~$ sudo service apache2 restart
```

## Usuarios:

Vamos al directorio en el que se encuentra el sitio y habilitamos el AllowOverride All, aunque puede que ya esté habilitado.

```
miadmin@PCBUS11:~$ sudo nano /etc/apache2/sites-enabled/pablo-com.conf
```

```
<Directory /var/www/pablo.com/public_html>
    AllowOverride All
</Directory>
```

Ahora crearemos el fichero .htaccess dentro del mismo directorio:

```
miadmin@PCBUS11:~$ sudo nano /var/www/pablo.com/public_html/.htaccess
```

```
AuthType Basic
AuthBasicProvider ldap
AuthName "Usuario y contraseña"
# Generar el filtro de búsqueda
AuthLDAPURL "ldap://192.168.3.111/dc=pablo,dc=local?uid?sub?(objectClass=*)"
# DN opcional para enlazar con la fase de búsqueda
AuthLDAPBindDN "cn=admin,dc=pablo,dc=local"
# Contraseña opcional para enlazar la fase de búsqueda
AuthLDAPBindPassword paso
# Fase de autenticación
Require ldap-user pcidon
```

Finalmente, reiniciamos el servicio de apache:

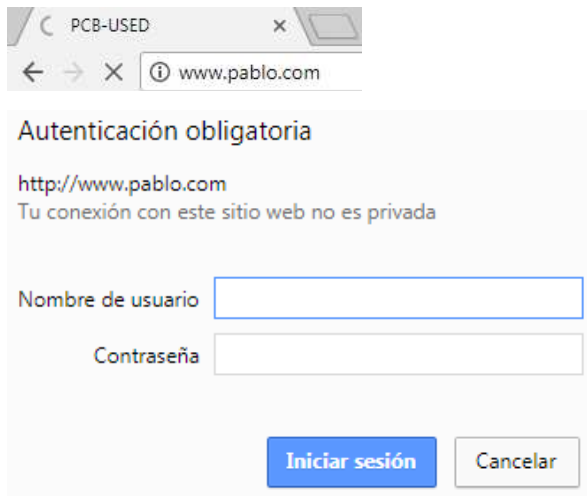
```
miadmin@PCBUS11:~$ sudo service apache2 restart
```

Puede ocurrir que nos dé error 500, por lo que tendremos que desactivar algunos sitios:

```
miadmin@PCBUS11:~$ sudo a2dissite pablo-com-ssl.conf
Site pablo-com-ssl disabled.
To activate the new configuration, you need to run:
    service apache2 reload
miadmin@PCBUS11:~$ sudo a2dissite pablo-es-ssl.conf
Site pablo-es-ssl disabled.
To activate the new configuration, you need to run:
    service apache2 reload
```

Reiniciamos el servicio y comprobamos que funciona:

*Pablo Cidón Barrio*



# PÁGINA DE PABLO.COM

## Grupos:

Partimos de nuestro fichero anterior .htaccess del que haremos una copia:

```
miadmin@PCBUS11:~$ sudo cp /var/www/pablo.com/public_html/.htaccess /var/www/pablo.com/public_html/usuario
```

Luego modificamos el fichero .htaccess para que la autenticación sea por grupos.

```
AuthType Basic
AuthBasicProvider ldap
AuthName "Usuario y contraseña"
# Generar el filtro de búsqueda
AuthLDAPURL "ldap://192.168.3.111/dc=pablo,dc=local?uid?sub?(objectClass=*)"
# DN opcional para enlazar con la fase de búsqueda
AuthLDAPBindDN "cn=admin,dc=pablo,dc=local"
# Contraseña opcional para enlazar la fase de búsqueda
AuthLDAPBindPassword paso
# Fase de autenticación
Require ldap-group cn=jefes,ou=grupos,dc=pablo,dc=local
```

Aplicamos cambios y reiniciamos el servicio:

```
miadmin@PCBUS11:~$ sudo service apache2 restart
```

Finalmente comprobamos que funciona correctamente: