



23-11-2022

## Práctica Servidores web1º trimestre

**Práctica Servidores web**  
**1º trimestre**

Vamos a instalar un servidor web interno para un instituto. Se Pide:

- Instalación del servidor web apache. Usaremos dos dominios mediante el archivo hosts: centro.intranet y departamentos.centro.intranet. El primero servirá el contenido mediante wordpress y el segundo una aplicación en Python
- Lo primero que vamos a realizar es la instalación del servidor apache:
- Para ello vamos entrar como super usuario para poder realizar cambios en el sistema sin tener que estar poniendo la contraseña todo el tiempo.

```
alvaro@alvaro-VirtualBox:~$ sudo su
[sudo] contraseña para alvaro:
root@alvaro-VirtualBox:/home/alvaro#
```

Ahora procedemos a instalar el servidor; actualizamos con un update, de la siguiente manera:

```
root@alvaro-VirtualBox:/home/alvaro# apt update
```

```
root@alvaro-VirtualBox:/home/alvaro# apt install apache2
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
apache2 ya está en su versión más reciente (2.4.29-1ubuntu4.25).
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
efibootmgr fonts-liberation2 fonts-opensymbol gir1.2-geocodeglib-1.0 gir1.2-gst-plugins-base-1.0
gir1.2-gstreamer-1.0 gir1.2-gudev-1.0 gir1.2-udisks-2.0 grilo-plugins-0.3-base gstreamer1.0-gtk3
libboost-date-time1.65.1 libboost-filesystem1.65.1 libboost-iostreams1.65.1 libboost-locale1.65.1
libcdr-0.1-1 libclucene-contribs1v5 libclucene-core1v5 libcmis-0.5-5v5 libcolamd2 libdazzle-1.0-0
libe-book-0.1-1 libedataserverui-1.2-2 libegl1-mesa libeot0 libepubgen-0.1-1 libetonyek-0.1-1
libevent-2.1-6 libexiv2-14 libfreerdp-client2-2 libfreerdp2-2 libfwup1 libgc1c2 libgee-0.8-2 libgexiv2-2
libgom-1.0-0 libgpgmepp6 libgpod-common libgpod4 liblangtag-common liblangtag1 liblirc-client0 libllvm8
liblua5.3-0 libmediaart-2.0-0 libmspub-0.1-1 libodfgen-0.1-1 libqwing2v5 libraw16 libvenge-0.0-0
libsgutils2-2 libssh-4 libsuitesparseconfig5 libvncclient1 libwayland-egl1-mesa libwinpr2-2 libxapian30
libxmlsec1 libxmlsec1-nss lp-solve media-player-info python3-mako python3-markupsafe syslinux
syslinux-common syslinux-legacy usb-creator-common
Utilice «sudo apt autoremove» para eliminarlos.
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 16 no actualizados.
root@alvaro-VirtualBox:/home/alvaro#
```

Ya tenemos instalado nuestro servidor apache, ahora vamos a aplicar algunas configuraciones en los ficheros para gestionar los dos dominios que tenemos que crear.

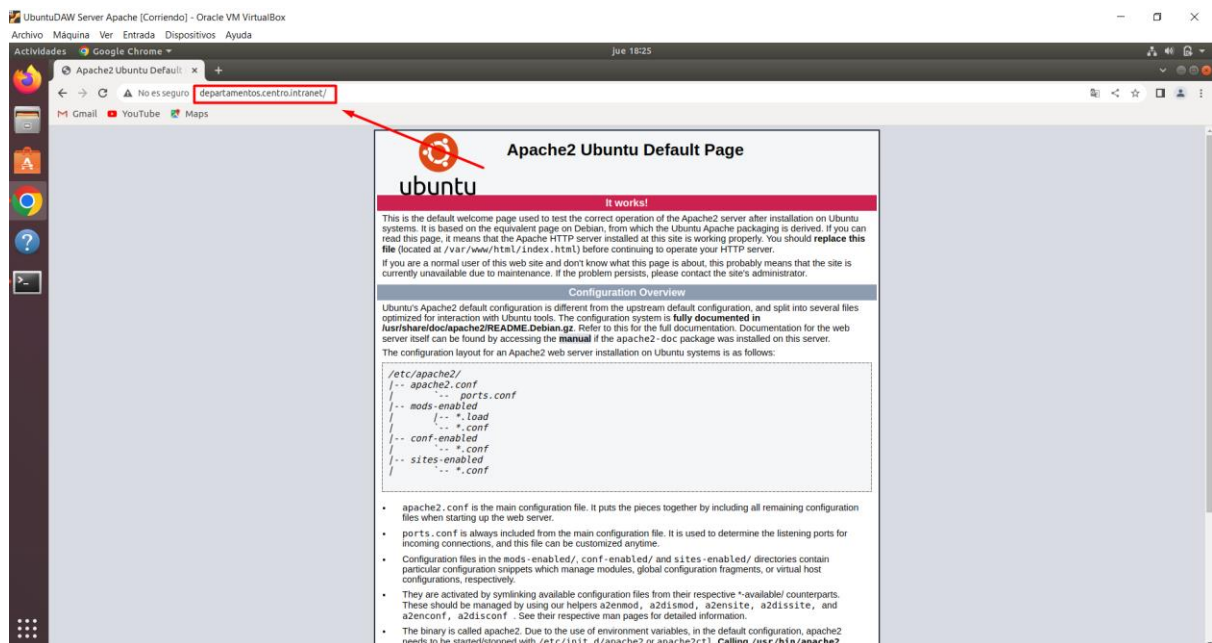
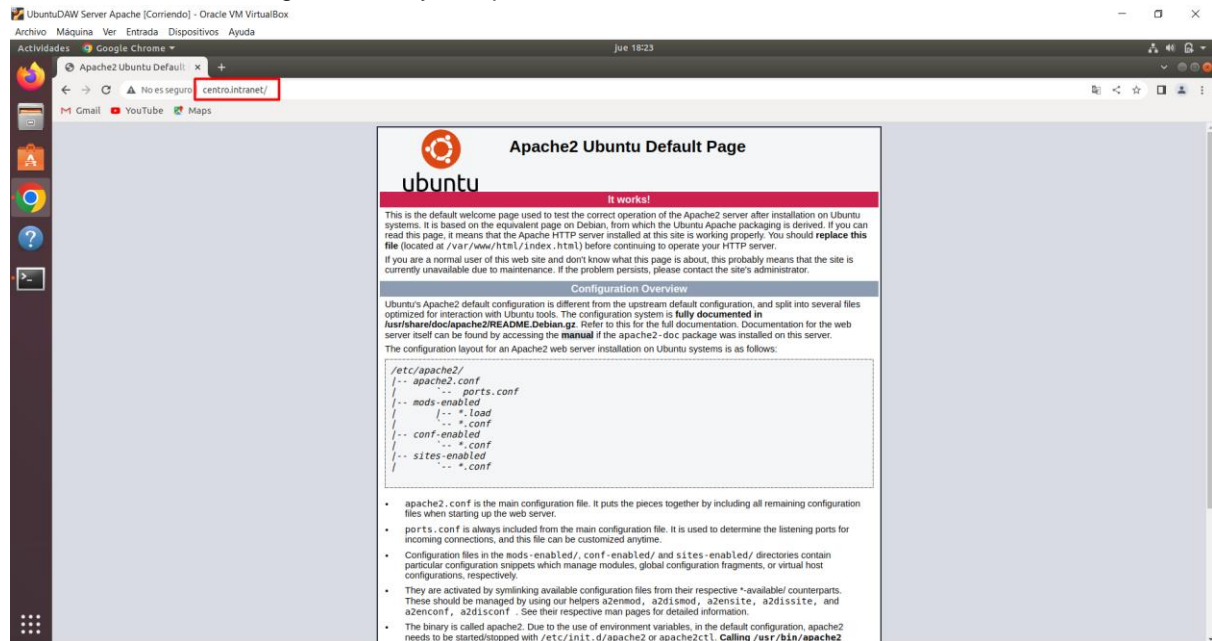
Para acceder a la configuración de los dominios hacemos lo siguiente:

Tenemos que acceder al archivo de configuración /etc/hosts.

```
root@alvaro-VirtualBox: /etc
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.9.3 hosts Modificado
127.0.0.1 localhost
127.0.1.1 alvaro-VirtualBox
127.0.0.1 marisma.intranet
127.0.0.1 centro.intranet
127.0.0.1 departamentos.centro.intranet
# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

## Álvaro Navarro Brenes

Añadimos los 2 nuevo guardamos y comprobamos.



- Activar los módulos necesarios para ejecutar php y acceder a mysql
  - Como podemos comprobar anteriormente ya tenemos nuestro servidor funcionando con los nuevos dominios y mostrándonos la página de default de apache, a continuación, vamos a proceder a activar los módulos para indexar en estos dominios nuestra página en php y en Python respectivamente.
  - Para ellos tenemos que ejecutar los siguientes comandos:
- Instalamos PHP y sus extensiones a la vez

```
root@alvaro-VirtualBox:/etc/apache2/sites-available# sudo apt install php libapache2-mod-php php-mysql php-curl php-gd php-xml php-mbstring php-xmllrpc php-zip php-soap php-intl -y  
Leyendo lista de paquetes... Hecho
```

## Álvaro Navarro Brenes

```
root@alvaro-VirtualBox: /etc/apache2/sites-available
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.9.3 /etc/apache2/mods-enabled/dir.conf
<IfModule mod_dir.c>
    DirectoryIndex index.html index.cgi index.pl index.php index.xhtml index.htm
</IfModule>
# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

Después de esto vamos a crear un archivo php que nos va a servir de ejemplo en el directorio principal de la web para comprobar si php funciona.

```
root@alvaro-VirtualBox:/var/www# cd /var/www/html
root@alvaro-VirtualBox:/var/www/html# nano ejemplo.php
root@alvaro-VirtualBox:/var/www/html#
```

Dentro del archivo que hemos creado ponemos un pequeño código de php para probar.

```
root@alvaro-VirtualBox:/var/www/html# cat ejemplo.php
<?php
echo("Bienvenido a la pagina de prueba de php ");
phpinfo();
?>
```

Ahora vamos a configurar MySQL y a crear una base de datos, para ello ponemos los siguientes comandos.

```
root@alvaro-VirtualBox:/var/www/html# apt install mysql-server -y
```

Una vez instalado abrimos el terminal de MySQL mediante el siguiente comando **sudo mysql**, en mi caso solo tengo que poner mysql ya que estoy como super usuario.

Una vez estamos en la consola de MySQL creamos la base de datos que vamos a necesitar mediante el siguiente comando.

```
mysql> CREATE DATABASE WordPressDB DEFAULT CHARACTER SET utf8 COLLATE utf8_unicode_ci;
Query OK, 1 row affected (0,02 sec)

mysql>
```

Ahora, crearemos una cuenta de usuario MySQL para operar en la nueva base de datos de WordPress. Utilizaremos WordPressDB como nombre de la base de datos y UsuarioWordPress como nombre de usuario y como contraseña "usuariousuario"

```
mysql> GRANT ALL ON WordPressDB.* TO 'UsuarioWordPress'@'localhost' IDENTIFIED BY 'usuariousuario';
Query OK, 0 rows affected, 1 warning (0,01 sec)
```

Para aplicar cambios utilizamos "FLUSH PRIVILEGES"  
Y salimos con "Exit".

```
mysql> EXIT;
Bye
root@alvaro-VirtualBox:/var/www/html#
```

### Instala WordPress

## Álvaro Navarro Brenes

- Lo primero que vamos a realizar es crear un archivo de configuración que va a ser por así decirlo una réplica del archivo original y lo vamos a crear en nano **/etc/apache2/sites-available/WordPress.conf**
- Lo siguiente que vamos a hacer es activar **.htaccess** para ello añadimos dentro del archivo que hemos creado anteriormente lo siguiente:

```
root@alvaro-VirtualBox: /etc/apache2/sites-available
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.9.3 /etc/apache2/sites-available/WordPress.conf Modificado
<Directory /var/www/wordpress />
    <AllowOverride All
</Directory>
```

- Ahora vamos a crear un directorio para WordPress en **/var/www/wordpress**

```
root@alvaro-VirtualBox:/var/www# ls
html wordpress
root@alvaro-VirtualBox:/var/www#
```

- Ahora, activamos el **mod\_rewrite** para utilizar la función de permalink o enlace permanente de WordPress ejecutando el siguiente comando en el terminal:

```
root@alvaro-VirtualBox:/var/www# sudo a2enmod rewrite
Enabling module rewrite.
To activate the new configuration, you need to run:
    systemctl restart apache2
root@alvaro-VirtualBox:/var/www#
```

- Ahora vamos a proceder a reiniciar el servidor apache para aplicar cambios
  - Reiniciamos con el siguiente comando: **systemctl restart apache2**
- El siguiente paso es cambiar la directiva **ServerName** en el archivo **/etc/apache2/apache2.conf**. para ello abrimos el archivo con el siguiente comando: **nano /etc/apache2/apache2.conf**
- Dentro de este archivo añadimos "serverName" y la "ip" de nuestro servidor
- Guardamos cambios y cerramos.
- Comprobamos si la configuración de apache es correcta y si es así nos tendrá que devolver Syntax OK

```
root@alvaro-VirtualBox:/var/www# apachectl configtest
Syntax OK
root@alvaro-VirtualBox:/var/www#
```

### Configuramos WordPress:

Lo primero va a ser instalar el paquete **wget** que nos servirá para descargar los archivos de wordpress mediante el comando: **apt install wget -y**.

A continuación, utiliza el comando **wget** seguido del enlace de descarga de WordPress:

```
root@alvaro-VirtualBox:/var/www/html# apt install wget -y
```

Y procedemos a instalar el paquete que nos hemos descargado.

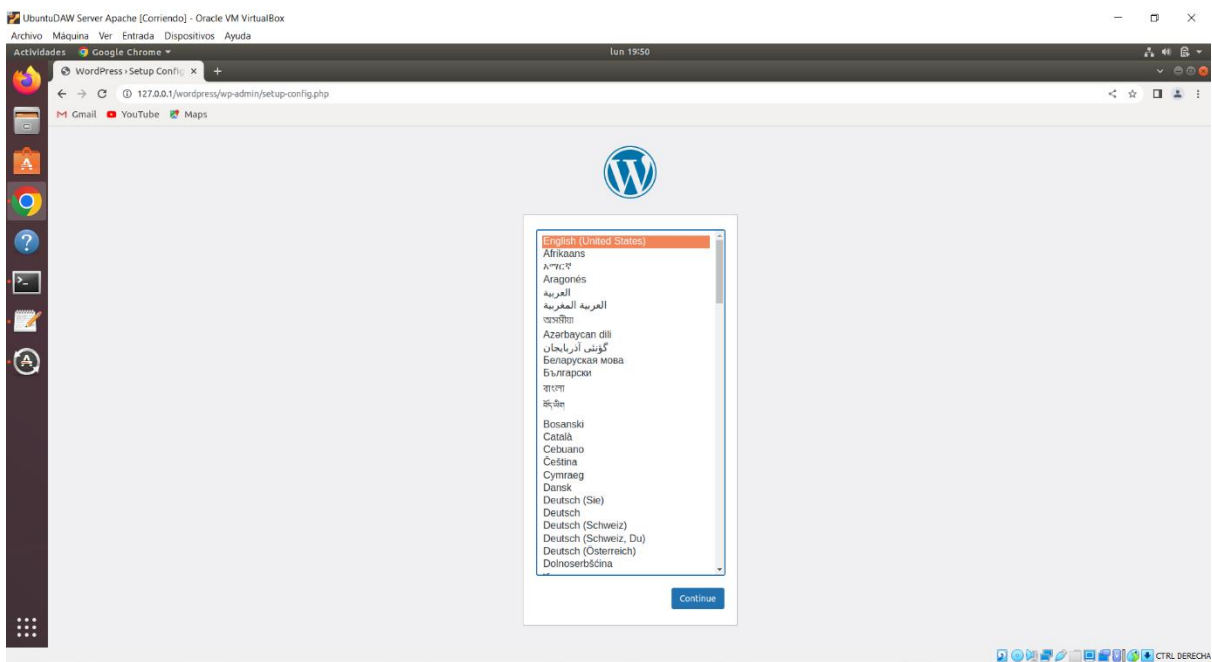
**Álvaro Navarro Brenes**

```
root@alvaro-VirtualBox:/var/www/html# ls
ejemplo.php  index.html  latest.zip
root@alvaro-VirtualBox:/var/www/html# apt install unzip -y
```

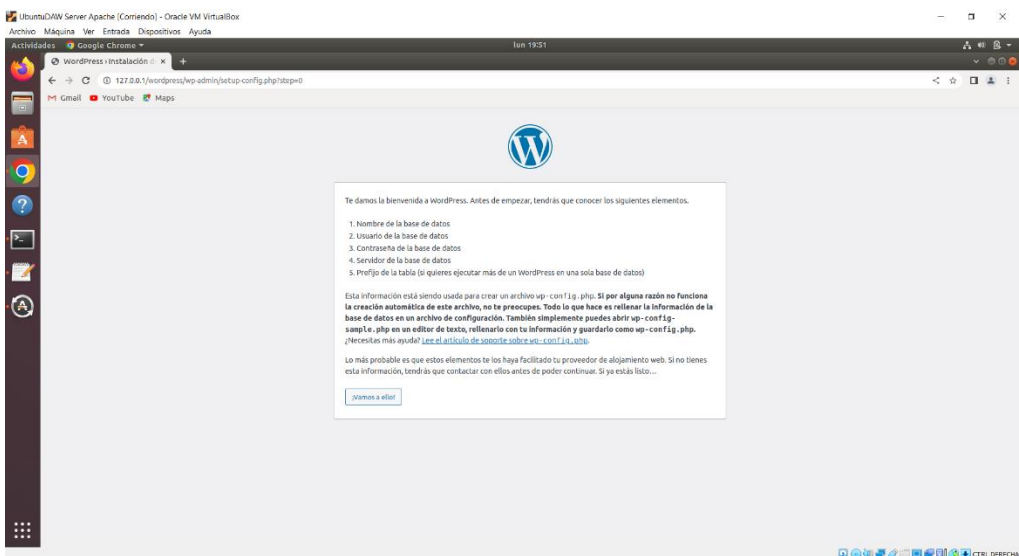
Y lo descomprimos con: “unzip latest.zip”, movemos el directorio: “mv -f wordpress/\* ./” y eliminamos el index.html mediante: “rm -rf index.html”

```
root@alvaro-VirtualBox: /var/www/html# rm -rf index.html
root@alvaro-VirtualBox: /var/www/html# ls
ejemplo.php  latest.zip  wordpress
root@alvaro-VirtualBox: /var/www/html#
```

Reiniciamos apache y terminamos la configuración desde el navegador.  
"sudo systemctl restart apache2"

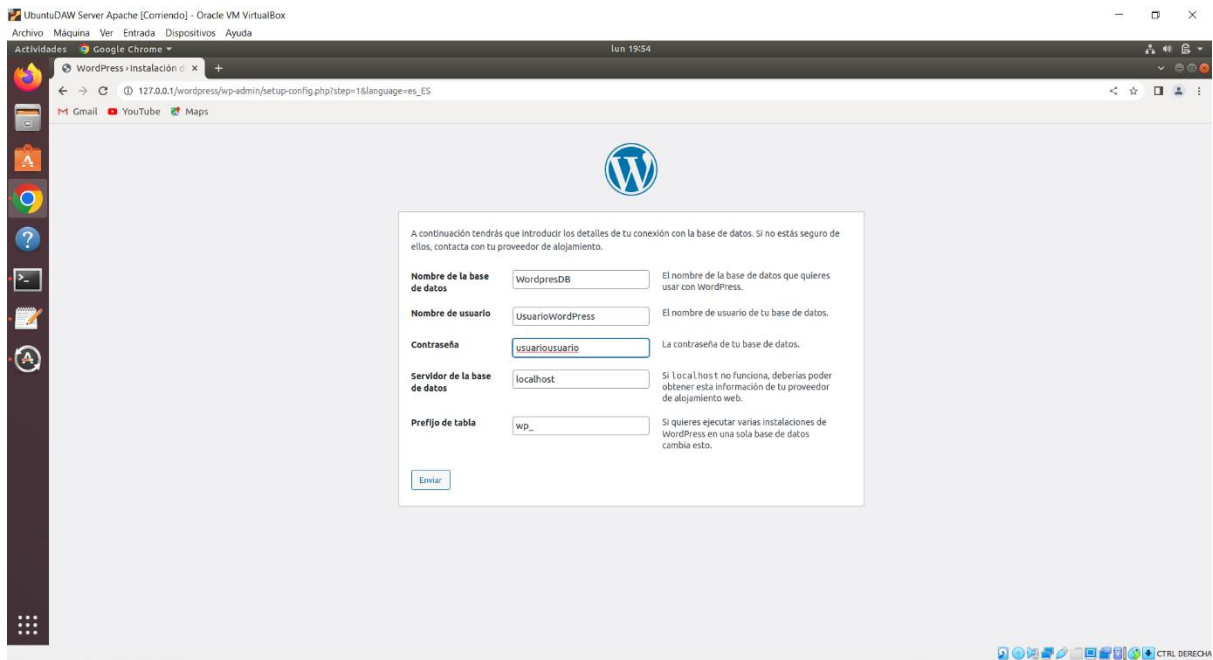


Seleccionamos un idioma, en nuestro caso español

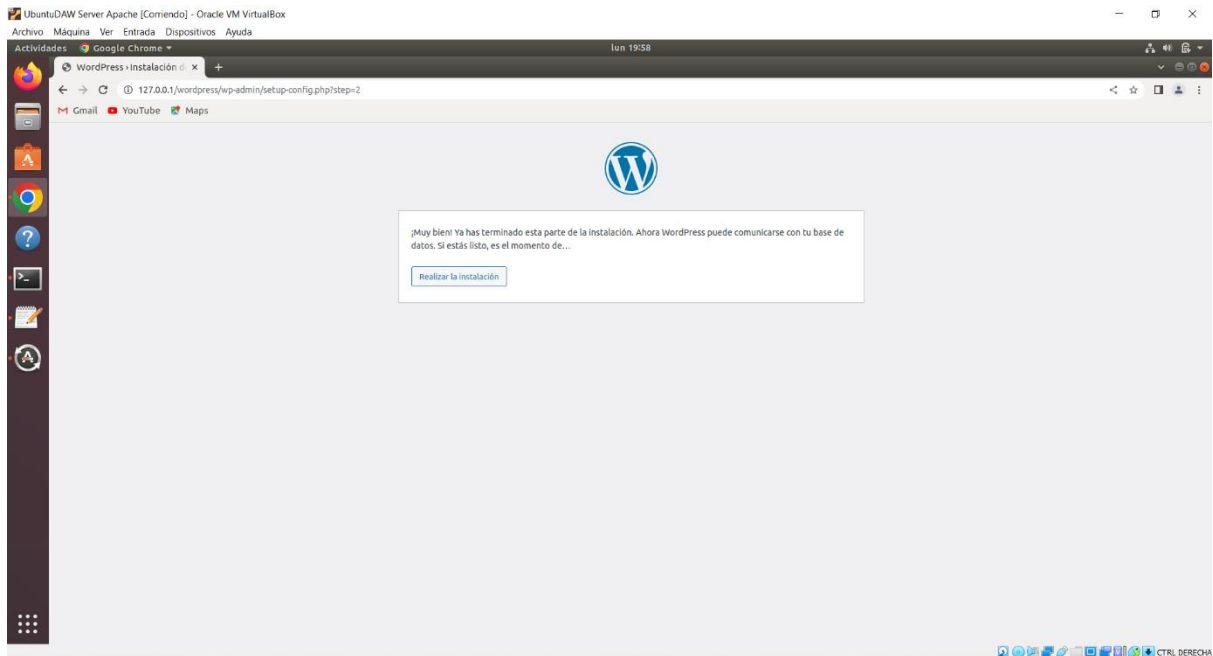


## Álvaro Navarro Brenes

Pulsamos sobre “vamos a ello”.



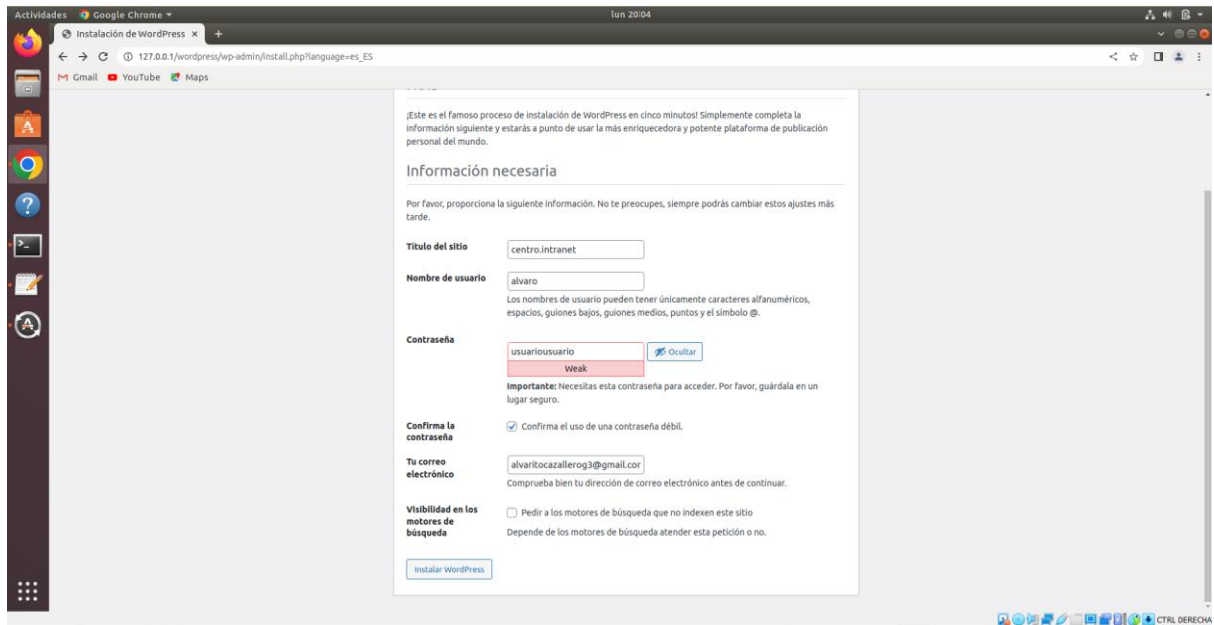
Completamos con los datos que hemos creado anteriormente y pulsamos sobre “Enviar”.



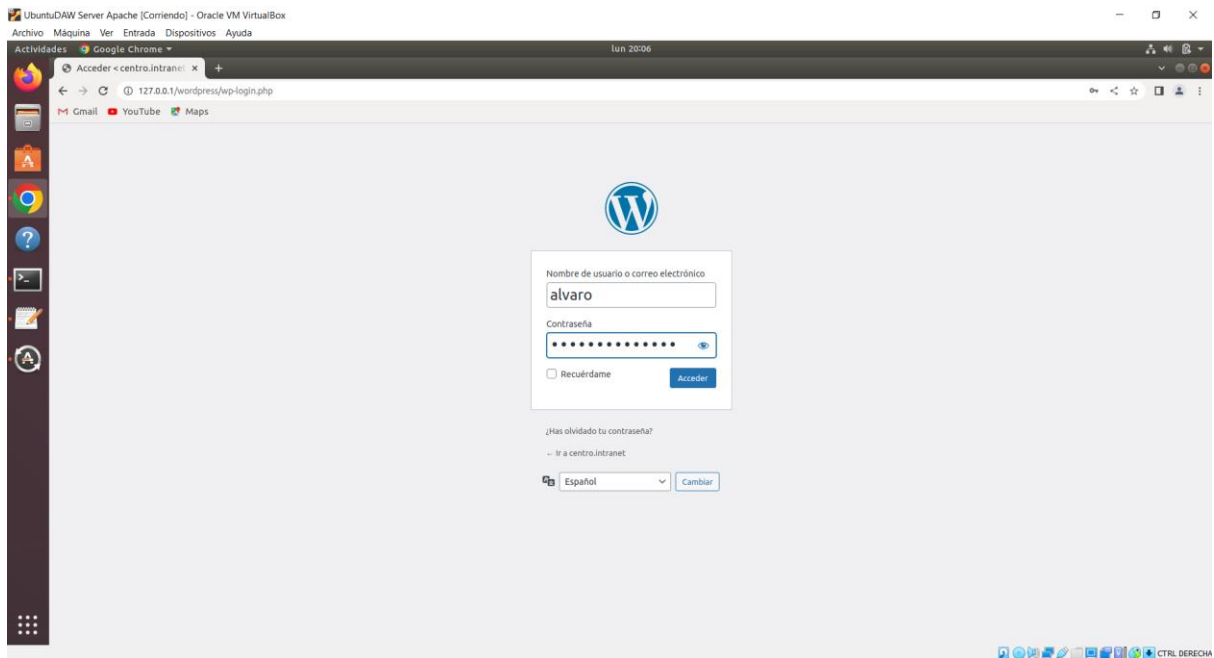


# Álvaro Navarro Brenes

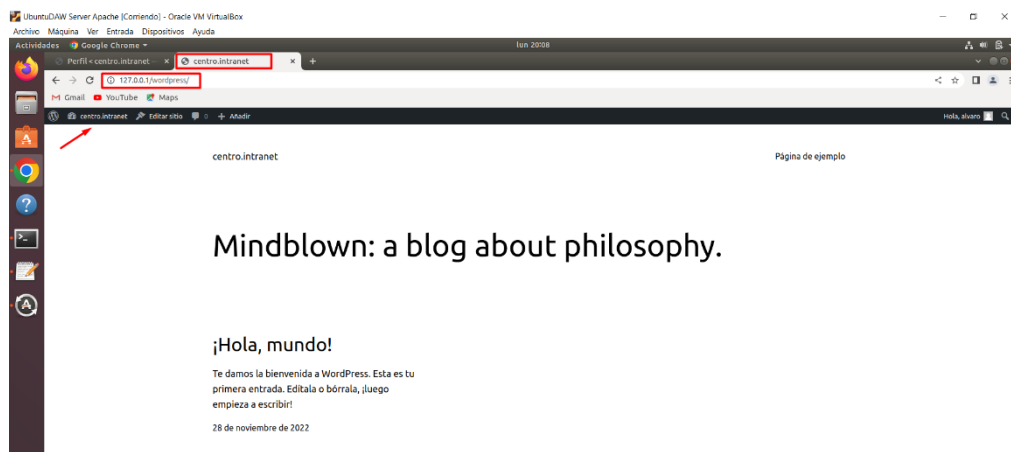
Pulsamos sobre ejecutar la instalación



Completamos los campos y finalizamos



Accedemos y ya tenemos nuestro sitio WordPress funcionando.





## Álvaro Navarro Brenes

- Activar el módulo “wsgi” para permitir la ejecución de aplicaciones Python  
→ “apt install libapache2-mod-wsgi-py3 -y”

```
root@alvaro-VirtualBox:/var/www/html# apt install libapache2-mod-wsgi-py3 -y
```

- Crea y despliega una pequeña aplicación python para comprobar que funciona correctamente.  
→ Creamos el archivo que contendrá dicho programa:

```
root@alvaro-VirtualBox: /var/www/html
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.9.3 aplicacion.py
print("Hola, esto es una prueba del archivo en python")
```

- Adicionalmente protegeremos el acceso a la aplicación python mediante autenticación

```
root@alvaro-VirtualBox:/var/www/html# apt-get install apache2-utils
```

- Creamos un archivo de contraseña “htpasswd -c /etc/apache2/.htpasswd”

```
root@alvaro-VirtualBox:/var/www/html# htpasswd /etc/apache2/.htpasswd alvaro
htpasswd: cannot modify file /etc/apache2/.htpasswd; use '-c' to create it
root@alvaro-VirtualBox:/var/www/html#
```

- Establecemos la contraseña que nos pedirá para autenticarnos:

```
root@alvaro-VirtualBox:/var/www/html# htpasswd -c /etc/apache2/.htpasswd alvaro
New password:
```

```
New password:
Re-type new password:
Adding password for user alvaro
root@alvaro-VirtualBox:/var/www/html#
```

- Aquí podemos ver la contraseña cifrada

```
root@alvaro-VirtualBox:/var/www/html# cat /etc/apache2/.htpasswd
alvaro:$apr1$PzApnNLc$56gSZnQCM0UB0nxx222mD1
root@alvaro-VirtualBox:/var/www/html#
```

- Instala y configura awstat.  
→ Instalacion:
  - apt-get update
  - apt-get install awstats libgeo-ipfree-perl libnet-ip-perl -y  
→ Configuración:

```
root@alvaro-VirtualBox: /etc
Archivo Editar Ver Buscar Terminal Ayuda
root@alvaro-VirtualBox:/etc# LogFormat=1
root@alvaro-VirtualBox:/etc# SiteDomain="example.com"
root@alvaro-VirtualBox:/etc# HostAliases="localhost 127.0.0.1 example.com"
root@alvaro-VirtualBox:/etc# AllowFullYearView=3
root@alvaro-VirtualBox:/etc# AllowAccessFromWebToFollowingIPAddresses="127.0.0.1 192.168.1.0-192.168.1.255"
root@alvaro-VirtualBox:/etc# LoadPlugin="tooltips"
root@alvaro-VirtualBox:/etc# LoadPlugin="graphgooglechartapi"
root@alvaro-VirtualBox:/etc# LoadPlugin="geoipfree"
root@alvaro-VirtualBox:/etc#
```

## Álvaro Navarro Brenes

```
root@alvaro-VirtualBox: /etc
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.9.3 /etc/awstats/awstats.conf.local Modificado

# You can overrides config directives here.
# This is particularly useful for users with several configs for
# different virtual servers, who want to reuse common parameters.
# Also, this file is not updated with each new upstream release.
SiteDomain="example.com"
HostAliases="localhost 127.0.0.1 example.com"
```

Muy importante que la línea “create 640 root www-data”, se vea así

```
root@alvaro-VirtualBox: /etc
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.9.3 /etc/logrotate.d/apache2

/var/log/apache2/*.log {
    daily
    missingok
    rotate 14
    compress
    delaycompress
    notifempty
    create 640 root adm
    sharedscripts
    postrotate
        if invoke-rc.d apache2 status > /dev/null 2>&1; then \
            invoke-rc.d apache2 reload > /dev/null 2>&1; \
        fi;
    endscrip
    prerotate
        if [ -d /etc/logrotate.d/httpd-prerotate ]; then \
            run-parts /etc/logrotate.d/httpd-prerotate; \
        fi; \
    endscrip
}

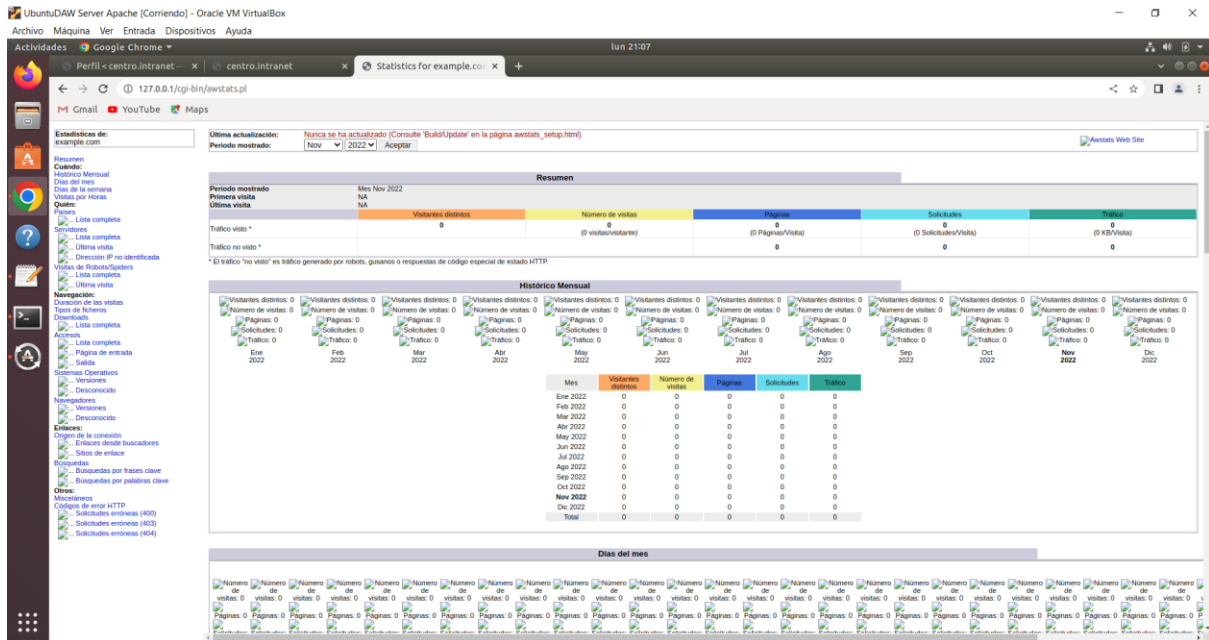
root@alvaro-VirtualBox: /etc
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.9.3 /etc/apache2/sites-available/example.com.conf Modificado

Alias /awstatsclasses "/usr/share/awstats/lib/"
Alias /awstats-icon/ "/usr/share/awstats/icon/"
Alias /awstatscss "/usr/share/doc/awstats/examples/css"
ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
ScriptAlias /awstats/ /usr/lib/cgi-bin/
Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch
```

Habilitamos modulo cgi y posteriormente reiniciamos apache.

```
root@alvaro-VirtualBox:/etc# sudo a2enmod cgi
Enabling module cgi.
To activate the new configuration, you need to run:
systemctl restart apache2
root@alvaro-VirtualBox:/etc#
```

# Álvaro Navarro Brenes



- Instala un segundo servidor de tu elección (nginx, lighttpd) bajo el dominio “servidor2.centro.intranet”. Debes configurarlo para que sirva en el puerto 8080 y haz los cambios necesarios para ejecutar php. Instala phpmyadmin.
  - Adjunto las capturas más importantes para no hacer muy pesada la practica!
  - Instalamos el servidor con “apt install nginx”
  - Habilitamos el servidor en el momento del arranque con: “systemctl enable nginx”

```
root@alvaro-VirtualBox:/# sudo ufw allow https comment 'Open all to access Nginx port 8080'
Reglas actualizadas
Reglas actualizadas (v6)
root@alvaro-VirtualBox:/#
```

- Cambiamos el puerto y probamos que nos funciona poniendo en el navegador la ip <http://104.200.23.232/>
- Para ejecutar php y phpmyadmin lo hacemos de igual manera que lo hemos realizado anteriormente.