A POR TODAS, TIGRE.

Hecho por: David Tardío Montoya, 2 GM B, Servicios en Red.

Índice.

[1: Preparación. 1](#_Toc97144063)

[LAMP: Conseguir los archivos de wordpress/moodle. 1](#_Toc97144064)

[2: RED INTERNA y DNS. 1](#_Toc97144065)

[3: Creación base de datos y usuario. 2](#_Toc97144066)

[HTTP: Configuraciones principales. 5](#_Toc97144067)

[FTP: QUE UN USUARIO PUEDA CAMBIAR EL INDEX.HTML. 6](#_Toc97144068)

# 1: Preparación.

Para empezar, modo NAT en el servidor. **TENER INSTALADO**  apache2, ssh, vsftpd, bind9, php.

Luego, instalar php-mysql, php-xml, php-curl.

En la máquina con la base de datos, instalar mariadb-server.

# LAMP: Conseguir los archivos de wordpress/moodle.

**MODO PUENTE**, descargar archivos en máquina real, y pasarlos por SCP al servidor. **PERMIT ROOT LOGIN**.

Service ssh restart tras el root login.

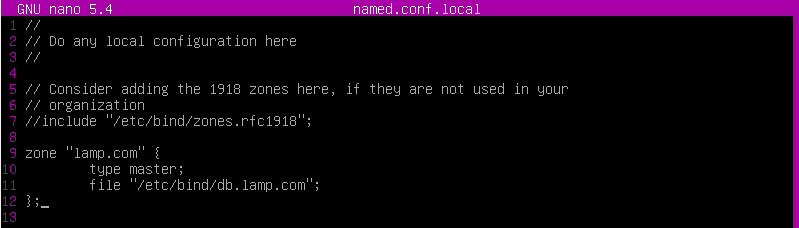
Guardar fichero en /var/www/html (crea su propia carpeta). **COMANDO UNZIP si es necesario, apt install unzip.**

Descomprimirlo usando **tar -xvf wordpress** (resto del archivo).

Copiar el sitio disponible (/etc/apache2/000-default.conf), y ponerlo en /etc/apache2/html2 (por ejemplo). **EDITAR CON NANO, Y AÑADIR DocumentRoot /var/www/html/wordpress**.

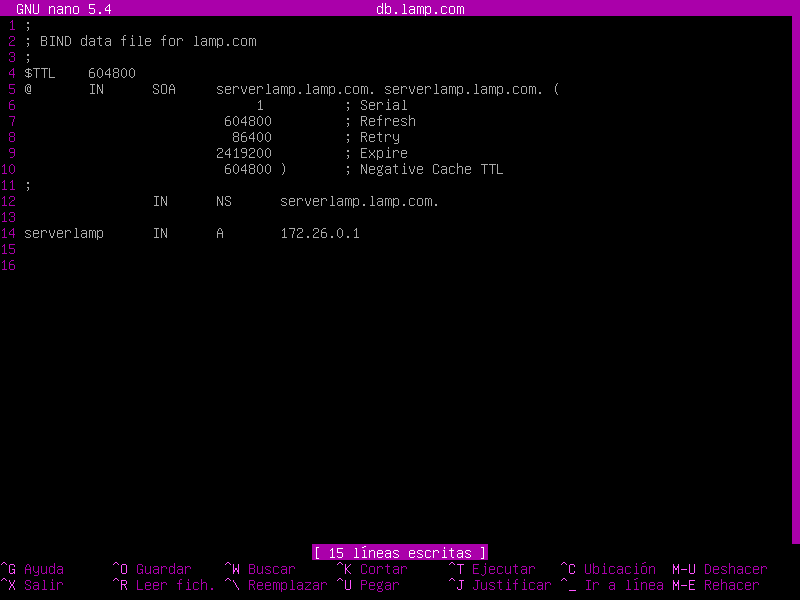
# 2: RED INTERNA y DNS.

Ponerlo en red interna, y editar el archivo **/etc/bind/named.conf.local**. Añadir zona de resolución directa (ejemplo:)



Copiar archivo **db.local** y llamarlo **db.lamp.com** (O COMO NOS DIGAN) **en el supuesto de que el dominio sea lamp.com.**

Se configura la zona, si la máquina se llamase **serverlamp**. RECORDAR HACER NAMED-CHECKCONF



**EN EL REENVIADOR, PONER EL 8.8.8.8 EN /etc/bind/named.conf.options. HAY QUE REENVIAR EL PAQUETITO PARA CONEXIÓN A INTERNET, CON IP\_FORWARD:**

**Nano /etc/sysctl.conf 🡪 descomentar net.ipv4.ip\_forward=1 🡪 sysctl –p** para refrescar.

**IPTABLES –t nat –A POSTROUTING –o enp0s**(el que se conecte a internet) **–j MASQUERADE.**

**--GUARDAR IPTABLES--**

Iptables-save >/etc/iptables.txt | nano /etc/network/if-pre-up.d/iptables

Pones #!/bin/bash

/sbin/iptables-restore < /etc/iptables.txt

Hay que dar permisos con chmod +x /etc/network/if-pre-up.d/iptables

**--FIN GUARDAR IPTABLES**

**EN EL CLIENTE**, EDITAR /etc/resolv.conf, y añadir ip. Comprobar haciendo ping serverlamp.lamp.com.

# 3: Creación base de datos y usuario.

**IMPORTANTE: EN EL ASISTENTE, CUANDO PREGUNTE COSAS DE LA BASE DE DATOS, PONER IP DE LA MÁQUINA DE VICTOR, USUARIO DE LA BASE DE VICTOR, CON LA INFORMACIÓN DE VÍCTOR. ASÍ ES EN REMOTO. SI ES ASÍ, NO REALIZAR NADA DEL TEXTO QUE HAY A CONTINUACIÓN, PORQUE LO HA HECHO VÍCTOR.**

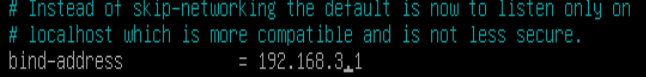
**----------SÓLO EN CASO DE QUE TENGAS UNA DATABASE EXTERNA-----------**

**-----EN CASO DE EMERGENCIA:** [**https://www.linode.com/docs/guides/configure-wordpress-remote-database/**](https://www.linode.com/docs/guides/configure-wordpress-remote-database/) **-----**

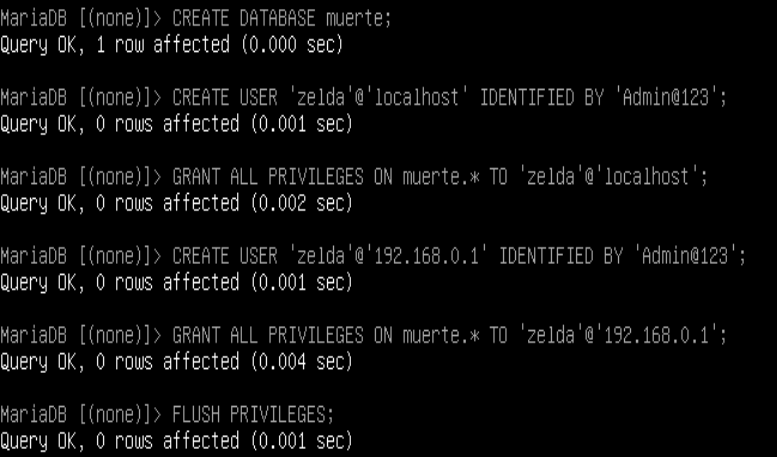
**EN MARÍADB:**

/etc/mysql/mariadb.conf.d/50-server.cnf

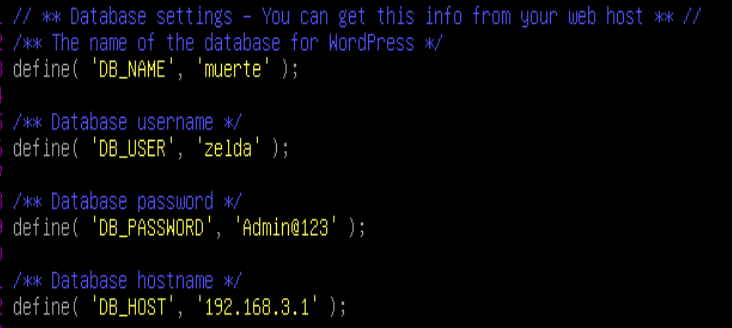
**bind-address = (IP DE MARIADB)**



**CREAR USUARIO EN LOCALHOST, Y EN EL SERVIDOR WEB.**



**EN SERVIDOR WEB, IR A /var/www/sitio2/wordpress, copiar archivo wp-config-sample.php, llamarlo wp-config.php, y hacer lo siguiente. En orden, es “nombre de database”, “usuario”; “contraseña de ese usuario”, “ip de MARIADB”.**





**---NORMAL---**

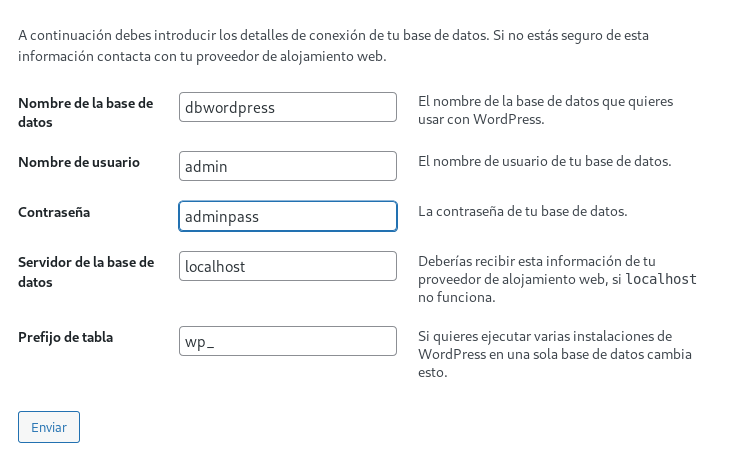
En el servidor, escribir mysql. Comando CREATE DATABASE dbwordpress (O EL NOMBRE QUE NOS INDIQUEN), CREATE USER ‘admin’@’localhost’ IDENTIFIED BY ‘adminpass’.

Admin es el nombre del usuario administrador, Adminpass es la contraseña.

GRANT ALL PRIVILEGES ON dbwordpress.\* TO ‘admin’@’localhost’. Dar privilegios.

FLUSH PRIVILEGES;

Desde el cliente, poner la dirección serverlamp.lamp.com para iniciar el asistente.



**SI NECESITAMOS UN FICHERO LLAMADO WP-CONFIG.PHP:**

Desde el cliente nos copiamos el fichero, y con SSH pasamos el fichero a /var/www/html2/wordpress.

**Necesitamos un fichero con ese nombre en el servidor.** Como no existe todavía, lo crearemos yendo a la carpeta **/var/www/html/wp/wordpress**, y haremos un ***touch wp-config.php***. Una vez realizado, deberemos copiar lo que nos indica Wordpress de una forma tan conveniente, y abriremos una sesión por terminal desde el cliente. Para ello, usaremos el comando ***ssh*** [***root@serverlamp.lamp.com***](mailto:root@serverlamp.lamp.com)***.*** Nos aseguraremos de tener copiado el texto que nos ha indicado Wordpress, y haremos ***nano /var/www/html/wp/wordpress/wp-config.php***. Una vez ahí, seleccionaremos la opción **pegar** dentro del menú **editar**. Guardamos, y reiniciamos el servicio Apache2.

**Serverlamp.lamp.com/wp-login.php.**

# HTTP: Configuraciones principales.

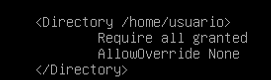
RECORDAR REINICIAR CONFIGURACIÓN CADA POCO.

CREAR SITIO, copiando el directorio por defecto (html) y pegarlo en (html1).

-ACTIVAR UN SITIO: a2ensite html2.conf.

**-HACER QUE SE ALOJE EN UN SITIO DISTINTO (INCLUYENDO /HOME/USUARIO):** cp /etc/apache2/sites-available/000-default.conf html1. Editar con Nano, y poner **DocumentRoot /var/www/html1/.**

**--SI ES EN /HOME/USUARIO, EN /ETC/APACHE2/APACHE2.CONF:**



**-QUE SEA ACCESIBLE POR IP Y PUERTO:** En virtualhost, poner **IP:puerto**, además de editar el archivo **ports.conf**, añadir línea **Listen** (puerto).

**-QUE BUSQUE UN ARCHIVO QUE NO SE LLAME Index.html:** En /var/www/html1.conf, poner **DirectoryIndex** (nombre del archivo).

Editar un archivo de apache2, poner

<HTML>

<body>

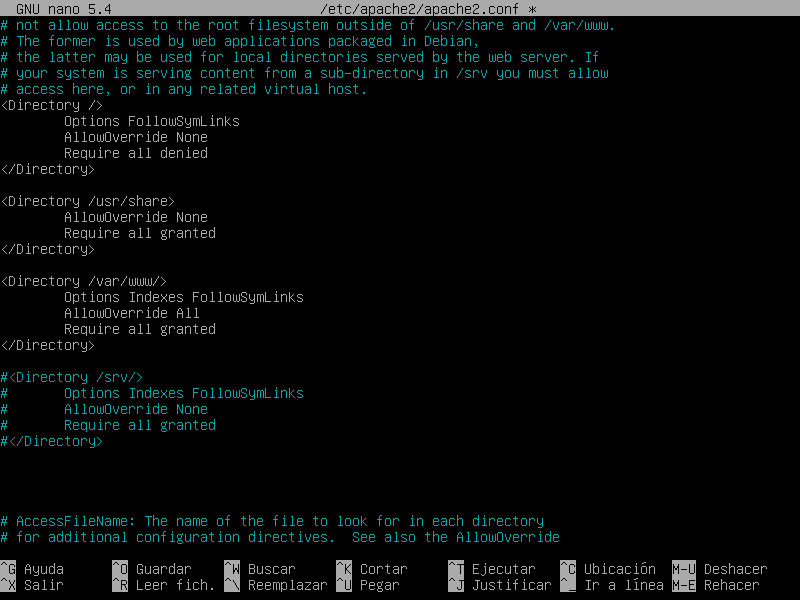
<H1> hola</H1>

</body>

<HTML>

Y usar un **service apache2 reload**. Para probarlo, poner IP:80.

**--PROTEGER CON CONTRASEÑA EL DIRECTORIO ROOT POR SITIO, Y QUE SÓLO UN USUARIO PUEDA ACCEDER**: tras lo de documentroot y todo eso, ***htpasswd –c /etc/apache2/.htpasswd*** (nombre usuario). También, ir a /etc/apache2/apache2.conf, en DIRECTORY /var/www, poner **AllowOverride All.**

****

Reiniciar servicio, y crear un archivo llamado .htaccess, y escribir:

**AuthType Basic**

**AuthName “Introduzca contraseña”**

**Require valid.user**

**AuthUserFile /etc/apache2/.htpasswd.**

# FTP: QUE UN USUARIO PUEDA CAMBIAR EL INDEX.HTML.

Haz un mecanismo para que los admins puedan sólo modificar sólo su sitio:

Crear cuentas de usuario para que administre cada sitio (user1 para /var/html1).

**SI ES UN SITIO WEB ESTÁTICO (NO WORDPRESS) EL DIRECTORIO /VAR/HTML1 DEBE SER DUEÑO DE USUARIO1, Y DARLE PERMISOS DE ESCRITURA.**

FTP, ENJAULAR A CADA USUARIO CORRESPONDIENTE, Y QUE SU DIRECTORIO DE CONEXIÓN SEA SU DIRECTORIO: /etc/vsftpd.conf

write\_enable=YES

chroot\_list\_enable=YES

chroot\_list\_file=/etc/vsftpd.chroot\_list

EDITAR ESE ARCHIVO Y DEJARLO VACÍO.

Puertos de escucha /etc/apache2/ports.conf

Entrar con contraseña a un sitio

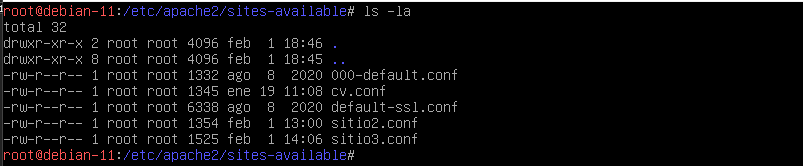


Ilustración : Vemos los archivos que están en la carpeta

y configuramos el archivo

cambiando la ip, DocumentRoot y que tengamos que poner usuario y contraseña

# nano sitio3.conf

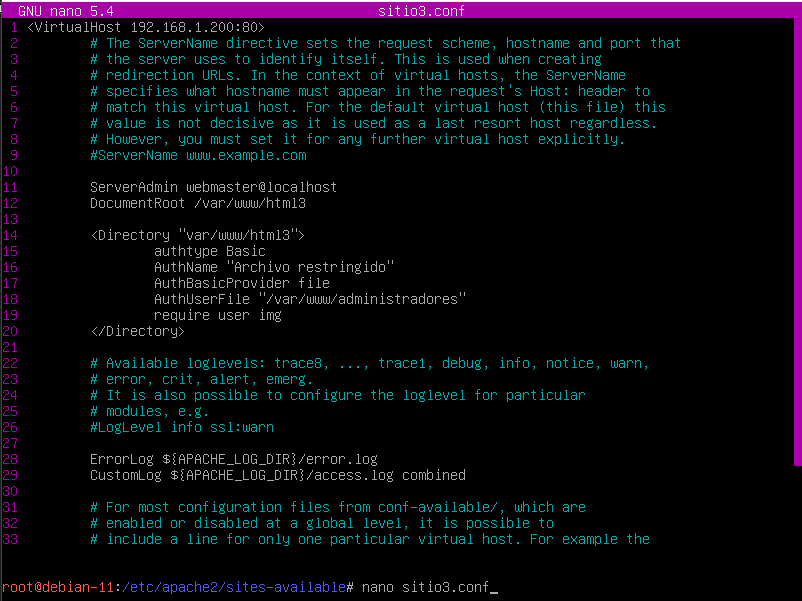


Ilustración : Configuramos el tercer sitio

De nada nos sirve tener esto, tenemos que crear los directorios y también crear el usuario y contraseña y que solo se conecte por usuario y contraseña

# nano htpasswd -c /var/www/administradores img

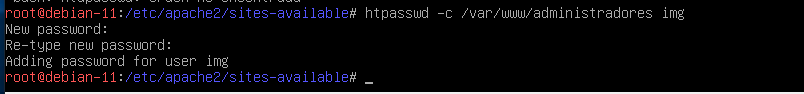


Ilustración 22: Creamos usuario y contraseña para conectarnos a la pagina web

Con este comando lo que hacemos es crear un usuario y una contraseña

# a2enmod auth\_basic



Ilustración 23: Hacemos que se conecte solo por usuario y contraseña

Con este comando lo que hacemos es que solo se conecte por usuario y contraseña

# a2ensite sitio3.conf

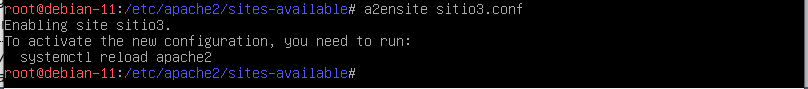


Ilustración 24: Añadimos el tercer sitio a la configuración de apache

y luego añadimos el sitio a la configuración de apache para que también lo cargue y reiniciamos el servicio

y vamos a probar que funciona correctamente

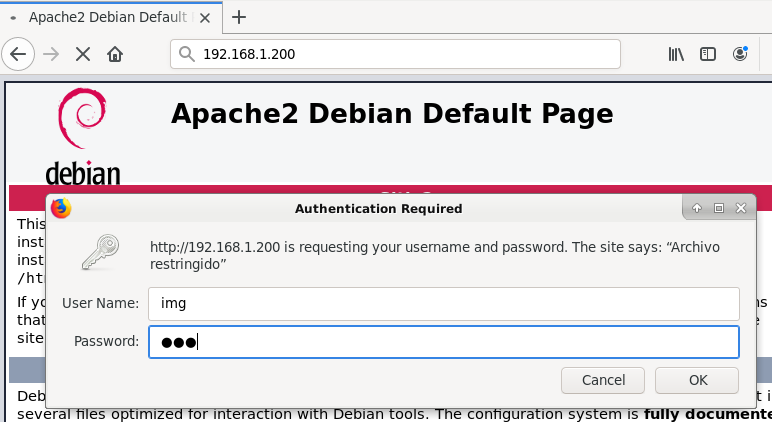


Ilustración 25: Vemos que nos pide usuario y contraseña

Me estoy volviendo loco ☹.