

APACHE + PHP:

1. Instalamos apache:

```
sudo apt install apache2
```

2. Ajustamos el cortafuegos:

```
sudo ufw app list
```

3. Debera salirnos:

```
Available applications:
Apache
Apache Full
Apache Secure
OpenSSH
```

4. Permitimos el trafico HTTP y HTTPS:

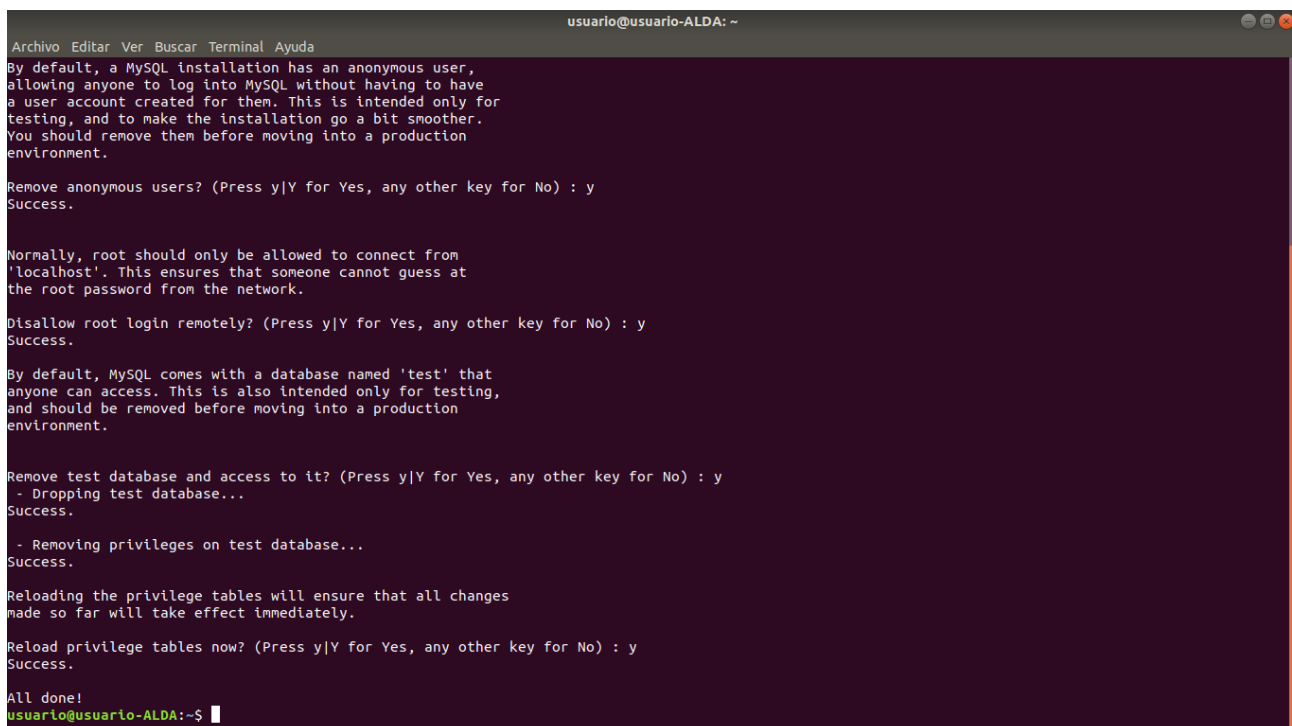
```
sudo ufw allow in "Apache Full"
```

5. Instalamos Mysql:

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

6. Ejecutamos archivo de comandos de seguridad:

```
sudo mysql_secure_installation
```



```
usuario@usuario-ALDA: ~
Archivo Editar Ver Buscar Terminal Ayuda
By default, a MySQL installation has an anonymous user,
allowing anyone to log into MySQL without having to have
a user account created for them. This is intended only for
testing, and to make the installation go a bit smoother.
You should remove them before moving into a production
environment.

Remove anonymous users? (Press y|Y for Yes, any other key for No) : y
Success.

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

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

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

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

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

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

All done!
usuario@usuario-ALDA:~$
```

7. Instalamos php:

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

```
usuario@usuario-ALDA:~$ sudo apt install php libapache2-mod-php php-mysql
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  libapache2-mod-php7.2 php-common php7.2 php7.2-cli php7.2-common php7.2-json php7.2-mysql php7.2-opcache php7.2-readline
Paquetes sugeridos:
  php-pear
Se instalarán los siguientes paquetes NUEVOS:
  libapache2-mod-php libapache2-mod-php7.2 php php-common php-mysql php7.2 php7.2-cli php7.2-common php7.2-json php7.2-mysql php7.2-opcache
  php7.2-readline
0 actualizados, 12 nuevos se instalarán, 0 para eliminar y 4 no actualizados.
Se necesita descargar 3.985 kB de archivos.
Se utilizarán 17,6 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] S
Des:1 http://es.archive.ubuntu.com/ubuntu bionic/main amd64 php-common all 1:60ubuntu1 [12,1 kB]
Des:2 http://es.archive.ubuntu.com/ubuntu bionic-updates/main amd64 php7.2-common amd64 7.2.15-0ubuntu0.18.04.1 [882 kB]
Des:3 http://es.archive.ubuntu.com/ubuntu bionic-updates/main amd64 php7.2-json amd64 7.2.15-0ubuntu0.18.04.1 [18,8 kB]
Des:4 http://es.archive.ubuntu.com/ubuntu bionic-updates/main amd64 php7.2-opcache amd64 7.2.15-0ubuntu0.18.04.1 [165 kB]
Des:5 http://es.archive.ubuntu.com/ubuntu bionic-updates/main amd64 php7.2-readline amd64 7.2.15-0ubuntu0.18.04.1 [12,1 kB]
Des:6 http://es.archive.ubuntu.com/ubuntu bionic-updates/main amd64 php7.2-cli amd64 7.2.15-0ubuntu0.18.04.1 [1.409 kB]
Des:7 http://es.archive.ubuntu.com/ubuntu bionic-updates/main amd64 libapache2-mod-php7.2 amd64 7.2.15-0ubuntu0.18.04.1 [1.351 kB]
Des:8 http://es.archive.ubuntu.com/ubuntu bionic/main amd64 libapache2-mod-php all 1:7.2+60ubuntu1 [3.212 B]
```

8. Hacemos que apache busque index.php en primer lugar:

`sudo nano /etc/apache2/mods-enabled/dir.conf`

9. Ponemos index.php en primero de la lista y reseteamos el sistema:

`sudo systemctl restart apache2`

10. Comprobamos el estado:

`sudo systemctl status apache2`

```
usuario@usuario-ALDA: ~
usuario@usuario-ALDA:~$ sudo systemctl status apache2
[sudo] password for usuario:
● apache2.service - LSB: Apache2 web server
   Loaded: loaded (/etc/init.d/apache2; bad; vendor preset: enabled)
   Drop-In: /lib/systemd/system/apache2.service.d
            └─apache2-systemd.conf
   Active: active (running) since lun 2019-03-18 08:20:29 CET; 1 day 2h ago
     Docs: man:systemd-sysv-generator(8)
  Process: 16361 ExecReload=/etc/init.d/apache2 reload (code=exited, status=0/SU)
    Tasks: 6
   Memory: 21.2M
      CPU: 2.668s
   CGroup: /system.slice/apache2.service
           └─3025 /usr/sbin/apache2 -k start
             16379 /usr/sbin/apache2 -k start
             16380 /usr/sbin/apache2 -k start
             16381 /usr/sbin/apache2 -k start
             16382 /usr/sbin/apache2 -k start
             16384 /usr/sbin/apache2 -k start

mar 18 08:20:29 usuario-ALDA apache2[2527]: *
mar 18 08:20:29 usuario-ALDA systemd[1]: Started LSB: Apache2 web server.
mar 18 08:24:26 usuario-ALDA systemd[1]: Reloading LSB: Apache2 web server.
mar 18 08:24:26 usuario-ALDA apache2[6016]: * Reloading Apache httpd web server
mar 18 08:24:26 usuario-ALDA apache2[6016]: *
lines 1-23
```

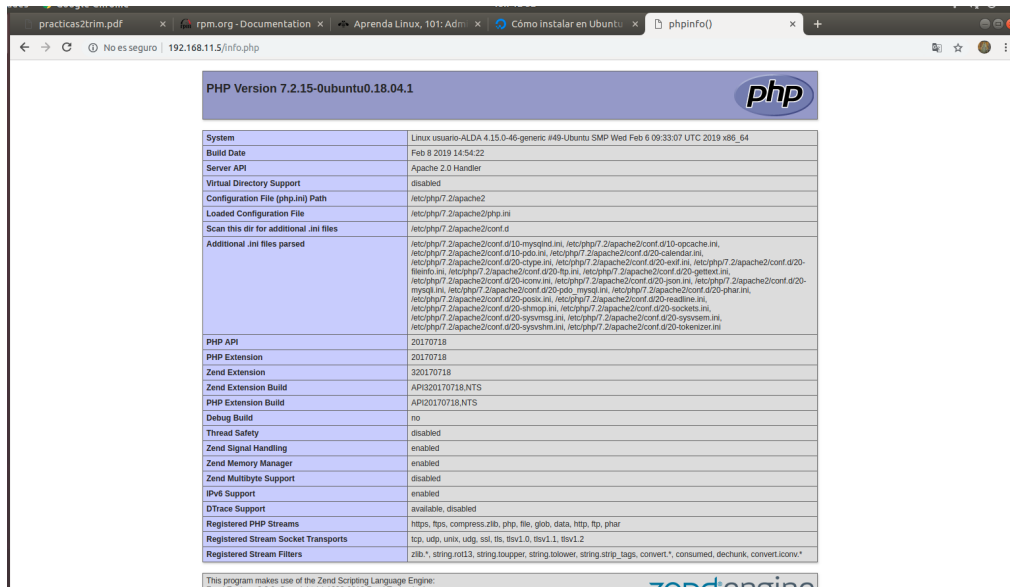
11. Comprobamos que nuestro equipo usa de manera correcta php, creamos el archivo info.php:

`sudo nano /var/www/html/info.php`

12. Colocamos dentro lo siguiente:

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

13. Ahora abrimos el navegador y poniendo "localhost" debería salir lo siguiente:



PHP Version 7.2.15-0ubuntu0.18.04.1	
System	Linux usuario-ALDA 4.15.0-46-generic #49-Ubuntu SMP Wed Feb 6 09:33:07 UTC 2019 x86_64
Build Date	Feb 8 2019 14:54:22
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.2/apache2
Loaded Configuration File	/etc/php/7.2/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/7.2/apache2/conf.d
Additional .ini files parsed	/etc/php/7.2/apache2/conf.d/10-mysqlnd.ini, /etc/php/7.2/apache2/conf.d/10-opcache.ini, /etc/php/7.2/apache2/conf.d/10-pdo.ini, /etc/php/7.2/apache2/conf.d/20-calendar.ini, /etc/php/7.2/apache2/conf.d/20-curl.ini, /etc/php/7.2/apache2/conf.d/20-exif.ini, /etc/php/7.2/apache2/conf.d/20-fileinfo.ini, /etc/php/7.2/apache2/conf.d/20-ftp.ini, /etc/php/7.2/apache2/conf.d/20-gd.ini, /etc/php/7.2/apache2/conf.d/20-gettext.ini, /etc/php/7.2/apache2/conf.d/20-iconv.ini, /etc/php/7.2/apache2/conf.d/20-json.ini, /etc/php/7.2/apache2/conf.d/20-mbstring.ini, /etc/php/7.2/apache2/conf.d/20-pdo_mysql.ini, /etc/php/7.2/apache2/conf.d/20-pdo_pgsql.ini, /etc/php/7.2/apache2/conf.d/20-readline.ini, /etc/php/7.2/apache2/conf.d/20-redis.ini, /etc/php/7.2/apache2/conf.d/20-shmop.ini, /etc/php/7.2/apache2/conf.d/20-sockets.ini, /etc/php/7.2/apache2/conf.d/20-sysvmsg.ini, /etc/php/7.2/apache2/conf.d/20-sysvsem.ini, /etc/php/7.2/apache2/conf.d/20-sysvshm.ini, /etc/php/7.2/apache2/conf.d/20-tokenizer.ini
PHP API	20170718
PHP Extension	20170718
Zend Extension	320170718
Zend Extension Build	API320170718.NTS
PHP Extension Build	API20170718.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	disabled
IPv6 Support	enabled
DTrace Support	available, disabled
Registered PHP Streams	https, ftps, compress.zlib, php, file, glob, data, http, ftp, phar
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, tls, tlsv1.0, tlsv1.1, tlsv1.2
Registered Stream Filters	zlib *, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, dechunk, convert.iconv.*

This program makes use of the Zend Scripting Language Engine: