

## Instalación LAMP

**Apache**, versión por defecto para la distribución de la instancia.

**mysql 10.2**, versión necesaria en base a los requerimientos de moodle 3.11:

- ! Asignamos la password root de mysql (+cFdWJ7>Ef)
- ! Creamos la base de datos de moodle y el usuario (moodleuser/moodleuser\_pass)

**PHP7.4**, versión mínima requerida PHP 7.3

- ! Configurar los valores de php en el php.ini
  - upload\_max\_filesize = 32M
  - post\_max\_size = 48M
  - memory\_limit = 256M
  - max\_execution\_time = 600
  - max\_input\_vars = 3000
  - max\_input\_time = 1000

Moodle 3.11, es la última versión

- User: Admin
- Pass: +cFdWJ7>Ef

Para poder hacer una estimación de los usuarios concurrentes que soportaría el servidor con las características actuales **2vCP/4GB**, podemos partir de la formula indicada en la documentación de moodle

- ! [https://docs.moodle.org/20/en/User\\_site\\_capacities](https://docs.moodle.org/20/en/User_site_capacities)

Número máximo de usuarios concurrentes debe ser igual a la cantidad de GB de memoria RAM multiplicado por 50 ( $nMuc = RAM(GB) * 50$ ), y que el número máximo aproximado de usuarios que acceden al servidor mediante el navegador web es igual al  $nMuc * 5$ .

En resumen se puede decir que es necesario **1GB por cada 50 usuarios** para una recurrencia de consultas entre 10% y 20% de los usuarios.

Con el siguiente comando por consola podemos sacar los primeros datos de rendimiento.

```
ab -n 100 -c 10 https://test.3ip.eu/
```

```
....  
...
```

Percentage of the requests served within a certain time (ms)

|      |                       |
|------|-----------------------|
| 50%  | 774                   |
| 66%  | 796                   |
| 75%  | 815                   |
| 80%  | 821                   |
| 90%  | 838                   |
| 95%  | 874                   |
| 98%  | 933                   |
| 99%  | 945                   |
| 100% | 945 (longest request) |

Letsencrypt

- ! Instalación con con certbot y configuración del virtual host

Backup:

Se ha creado un script bash para hacer el backup de los directorios moodle, moodledata y la base de datos.

el backup ejecuta cada hora con el cron `/etc/cron.d/moodle_backup` y se guarda en `/var/www/html/moodle_backup/`

*Como mejora para el sistema de backup se podría montar un sistema externo de cabina de storage y este backup podría hacer la copias incrementales con rsync. De este modo se puede asegurar el backup incluso si la instancia se elimina o se corrompe*

Llegados a este punto tenemos el servidor 100% funcional con con los requisitos solicitados

Como punto extra he añadido una monitorización basada en:

- Grafana: Como interfaz web
- Prometheus: Base de datos con la que interactúa node-exporter.
- node-exporter: Colector que recopila los datos de la instancia y los envía a prometheus

- ! <http://test.3ip.eu:3000/goto/ObCBgkDnk?orgId=1>
- ! Usuario:3ipunt (usuario con permisos solo de visualización)
- ! Clave:3ipunt

Las instalaciones tanto de la paquetería (mariadb, php, apache, prometheus, grafana, node-exportes), com el moodle, el letsencrypt y las configuraciones necesarias han sido instalado a mano desde consola

Como mejora general, ansible o salstack nos proporciona la posibilidad de automatizar el despligue de configuraciones y la instalación de paqueterías de manera rápida y escalable.

Con el uso de tecnologías como ANSIBLE o SALSTACK, que nos proporciona la posibilidad de automatizar el despique de configuraciones y la instalación de paqueterías, podemos desplegar de manera desatendida la bases LAMP, la monitorización (prometheus, grafana, node-exportes,...) y las configuraciones necesarias para cada software.