

SA04AF03B. Instal·lació i configuració d'un servidor MySQL i interacció amb un servidor d'aplicacions.

Índex

1. Introducció.....	1
2. Servidor MySQL.....	1
2.1 Instal·lació del servei.....	2
2.2 Usuari d'administració.....	2
2.3 Estructura de directoris i fitxers.....	3
2.3.1 Directives del fitxer mysqld.cnf.....	4
3. Stack LEMP.....	4
3.1 Topologia física/lògica.....	5
3.2 Creació d'usuaris per a l'aplicació.....	6
3.3 Instal·lació del mòdul php-MySQL.....	7
4. Treball a realitzar.....	8
5. Referències.....	9

1. Introducció

Al llarg de la pràctica analitzarem les principals arquitectures que podem implementar dins de l'**stack LEMP**, centrant-nos en la instal·lació i configuració del SGBD i el seu accés des del servidor d'aplicacions **PHP-FPM**.

2. Servidor MySQL

MySQL és un sistema de gestió de bases de dades relacional desenvolupat sota llicència dual; Llicència pública general i comercial per l'empresa Oracle Corporation. Actualment, és considerada com a la base de dades de codi obert més popular del món, sobretot en entorns web.



MySQL va ser inicialment desenvolupada per la companyia MySQL AB, però la seua compra per part d'Oracle, va propiciar l'aparició de MariaDb, un fork directe de MySQL que assegura l'existència d'una versió d'aquest producte amb llicència GPL, ja que el seu creador estava convençut que l'únic **interés d'Oracle en MySQL** era reduir la competència que suposava per al major proveïdor de bases de dades relacionals del món, que és Oracle.

2.1 Instal·lació del servei

En distribucions **Linux** la instal·lació del SGBD MySQL és molt senzilla, es fa a terme mitjançant el gestor de paquets:

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

Amb aquest comandament instal·larem el servidor mysql, i els paquets del **client de línia de comandaments** de mysql, que ens permetran accedir i gestionar les diferents bases de dades.

Per a gestionar el servei, utilitzarem el comando **systemctl** o el seu corresponent **service**.

```
systemctl [enable|disable|start|stop|  
restart] mysql
```

```
service mysql [start|stop|  
restart]
```

2.2 Usuari d'administració

Als sistemes operatius **ubuntu server**, l'autenticació de l'usuari **root** del SGBD, s'estableix per defecte mitjançant **auth_socket** en lloc del tradicional **usuari/password**, el que proporciona una major seguretat.

Aquesta manera d'autenticació es basa a crear i obrir un **socket Unix** i assignar quins usuaris del sistema operatiu host poden llegir i escriure en ell (en el nostre cas serà l'usuari **root**). D'aquesta manera, quan vulguem connectar-nos, només haurem d'executar el client mysql com a usuari root i tindrem accés directe a la gestió del SGBD.



```
$ sudo mysql
```



Deurem tindre en compte que la instal·lació per defecte de MySQL només permet connectar-nos al SGBD des de localhost i a través de l'usuari root utilitzant `auth_socket`, per la qual cosa **no podrem** establir la connexió **des de PHP**,

Activitat 1

Llig la següent [documentació](#) sobre l'ús del **client mysql**. Quin comandament i opcions utilitzaries si necessitem connectar-nos mitjançant **usuari/password** amb les següents credencials?

- **user:** app1prod
- **password:** 1234
- **database:** crm_db

2.3 Estructura de directoris i fitxers

La instal·lació de MySQL estableix el directori `/etc/mysql` com a base per a l'establiment de les **configuracions globals del servei**. L'arxiu principal el trobem a `/etc/mysql/mysql.cnf`. No obstant, amb la finalitat d'afavorir la llegibilitat i sostenibilitat del servei, s'estableix una estructura modular d'arxius de configuració, que podem trobar a través del path `/etc/mysql/conf.d/`. Podem destacar els següents arxius:

fitxer/directori	Descripció
<code>/etc/mysql/conf.d/mysqld.cnf</code>	Conté les configuracions globals que volem a aplicar a la nostra instància de mysql.
<code>/etc/mysql/mysql.conf.d/mysqld.cnf</code>	Amb la finalitat de mantenir retro-compatibilitat cap endarrere, trobem el mateix fitxer amb les configuracions per defecte en el path <code>/etc/mysql/mysql.conf.d/mysqld.cnf</code>

2.3.1 Directives del fitxer `mysqld.cnf`

Si analitzem el fitxer `/etc/mysql/mysql.conf.d/mysqld.cnf`, Podem veure que s'han establert una sèrie de directives per defecte, com són:

Directiva	Descripció
user user=mysql	Estableix l'usuari del S.O. amb el qual s'executarà el procés de mysql.
port port=3306	Port per defecte en el qual el servidor es mantindrà a l'escolta
bind-address bind-address=127.0.0.1	Estableix la interfície de xarxa per a la qual atendra les peticions el servidor mysql.
log_error log_error=/var/log/mysql/error.log	Fitxer de log per defecte del servei
slow_query_log slow_query_log=1	Activació / desactivació del registre de consultes pesades que s'estan duent a terme en el servidor

Pots consultar una llista completa de directives/variables del sistema en el [següent](#) document.

Activitat 2

- Quina directiva ens permet especificar una **política d'expiració** de contrasenyes? Posa un exemple perquè per defecte assigne 6 mesos a tots els comptes creats.
- Quin és el valor per defecte de la directiva **bind-address**? Des que interfícies es mantindrà a l'escolta el servidor si no s'especifica la directiva? Què implica aquesta opció?

3. Stack LEMP

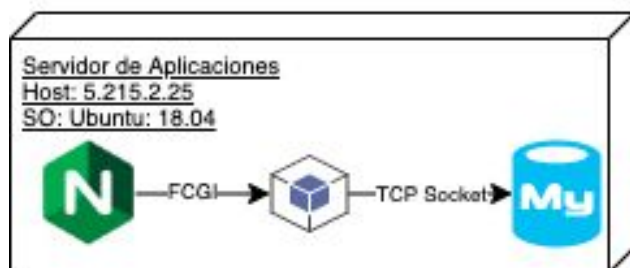
Com ja s'ha comentat, l'**stack LEMP** és una de les arquitectures més comunes i utilitzades hui dia. Es tracta de la combinació d'un sistema operatiu **Linux**, un servidor web **Nginx**, la tecnologia **PHP** i el servidor **MySql** com a Sistema Gestor de Bases de

dades Relacionals.

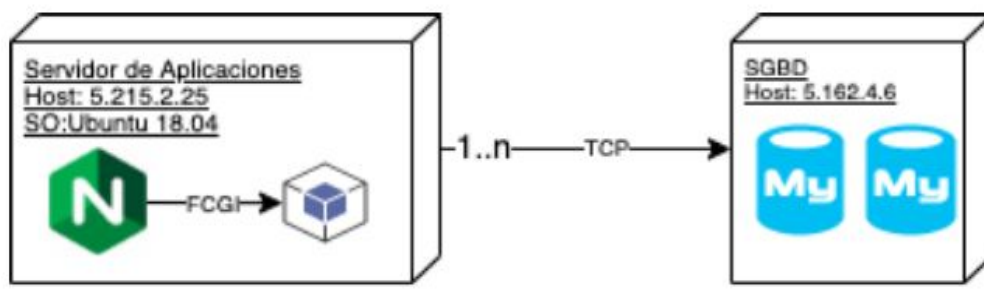
3.1 Topologia física/lògica

Des del punt de vista de la capa de dades, podem trobar-nos diferents topologies físiques o lògiques.

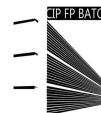
- **SGBD local:** El **Sistema gestor de bases de dades** i el **servidor d'aplicacions** es troben en la mateixa **màquina** pel que no és necessària una connexió de xarxa remota en cada petició. Aquesta topologia és vàlida quan l'aplicació que es desenvoluparà és xicoteta i ha d'atendre poc trànsit.



- **SGBD remot:** A mesura que l'aplicació creix o els serveis que ofereix l'empresa són més nombrosos, és una bona idea separar el SGBD i el servidor d'aplicacions de manera que cadascun pugui operar amb els seus propis recursos distribuint la càrrega derivada de respondre les peticions dels visitants.



Com s'ha comentat anteriorment, la instal·lació per defecte només permet les connexions des de la mateixa màquina. Si volem establir un SGBD remot, haurem d'habilitar l'escolta per les interfícies remotes establint la directiva bind-address 0.0.0.0



3.2 Creació d'usuaris per a l'aplicació

Cada **aplicació** ha de disposar d'un **usuari específic** amb els **mínims privilegis** que necessita per al seu funcionament, d'aquesta manera mantindrem diferent dominis de seguretat i una vulnerabilitat descoberta en una aplicació no afectarà la resta.

En primer lloc, deurem crear la base de dades que la nostra aplicació utilitzarà:

```
mysql> CREATE DATABASE myDataBase;
```

Una vegada creada la base de dades crearem l'usuari que utilitzarà l'aplicació per a l'establiment de les connexions.

```
mysql> CREATE USER 'user'@'localhost' IDENTIFIED BY 'password';
```

- **user**: Usuaris que utilitzarem per a connectar-nos amb el servidor
- **localhost**: Ip des de la qual aquest usuari podrà connectar-se al servidor. (Si utilitzem '%' podrà connectar-se des de qualsevol IP)
- **password**: Password que utilitzarem per a autenticar-nos. (Podem fer ús de l'eina openssl per a generar un password segur)

```
$ openssl rand -base64 32
```

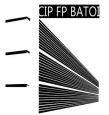


En mysql els usuaris s'identifiquen tant pel nom com per l'origen de les connexions, per la qual cosa un usuari [user@localhost](#) serà diferent que `user@'%'`

Podem consultar els usuaris de mysql mitjançant una consulta al diccionari de dades del sistema.

```
mysql> SELECT user,plugin,host FROM mysql.user;
```

Finalment, **assignarem** els **permisos** que necessita la nostra aplicació **a l'usuari** i els aplicarem



```
mysql> GRANT ALL PRIVILEGES ON myDataBase.* TO 'user'@'localhost';  
mysql> FLUSH PRIVILEGES;  
mysql> exit();
```

Activitat 3

- Amb el comandament anterior hem assignat a l'usuari "**user@localhost**", tots els privilegis sobre la base de dades myDataBase. Com establiries únicament permisos DML per a un usuari? Pots consultar la **taula** de privilegis de la [documentació oficial](#).

3.3 Instal·lació del mòdul php-MySQL

Ja tenim instal·lats tots els components del **stack** de la nostra aplicació, no obstant això encara no podem establir comunicació amb el SGBD des del servidor d'aplicacions.

Encara que la **comunicació amb el SGBD** es durà a terme a partir del llenguatge **SQL**, a un nivell menor d'abstracció, **la nostra aplicació** necessitarà poder establir una connexió i comunicar-se amb l'**API nativa** que ha proporcionat el **desenvolupador** del **SGBD**.

Aquesta tasca es duu a terme a partir dels **controladors** o **drivers** que, com podem veure en la [pàgina oficial](#), es troben disponibles per a la major part de tecnologies existents i que utilitzarem a partir d'un mòdul específic del nostre servidor d'aplicacions o **intèrpret PHP (PHP-FPM)**.

En sistemes operatius Unix i Like-Unix podem fer ús del gestor de paquets per a la seua instal·lació.

```
$ sudo apt install php-mysql;
```

Com podem observar, en la següent imatge, s'instal·larà i carregarà el driver natiu de mysql (**mysldnd**) i les llibreries de php natives **pdo** i **mysqli** per a gestionar la comunicació amb el servidor d'aplicacions.

```

3  Fetched 132 kB in 1s (150 kB/s)
debconf: delaying package configuration, since apt-utils is not installed
Selecting previously unselected package php8.1-mysql.
(Reading database ... 106907 files and directories currently installed.)
Preparing to unpack .../php8.1-mysql_8.1.2-1ubuntu2.8_amd64.deb ...
Unpacking php8.1-mysql (8.1.2-1ubuntu2.8) ...
Selecting previously unselected package php-mysql.
Preparing to unpack .../php-mysql_2%3a8.1+92ubuntu1_all.deb ...
Unpacking php-mysql (2:8.1+92ubuntu1) ...
Setting up php8.1-mysql (8.1.2-1ubuntu2.8) ...
debconf: unable to initialize frontend: Dialog
debconf: (No usable dialog-like program is installed, so the dialog based frontend cannot be used. at /usr/share/perl5/De
bconf/FrontEnd/Dialog.pm line 78.)
debconf: falling back to frontend: Readline

mysql> echo

Creating config file /etc/php/8.1/mods-available/mysqlnd.ini with new version
debconf: unable to initialize frontend: Dialog
debconf: (No usable dialog-like program is installed, so the dialog based frontend cannot be used. at /usr/share/perl5/De
bconf/FrontEnd/Dialog.pm line 78.)
debconf: falling back to frontend: Readline

mysql> echo

Creating config file /etc/php/8.1/mods-available/mysqlcli.ini with new version
debconf: unable to initialize frontend: Dialog
debconf: (No usable dialog-like program is installed, so the dialog based frontend cannot be used. at /usr/share/perl5/De
bconf/FrontEnd/Dialog.pm line 78.)
debconf: falling back to frontend: Readline

Creating config file /etc/php/8.1/mods-available/pdo_mysql.ini with new version
Setting up php-mysql (2:8.1+92ubuntu1) ...
Processing triggers for php8.1-fpm (8.1.2-1ubuntu2.8) ...
Processing triggers for php8.1-cli (8.1.2-1ubuntu2.8) ...
debconf: unable to initialize frontend: Dialog
debconf: (No usable dialog-like program is installed, so the dialog based frontend cannot be used. at /usr/share/perl5/De
bconf/FrontEnd/Dialog.pm line 78.)
debconf: falling back to frontend: Readline
Scanning processes...
Scanning linux images...

```

Activitat 4

- Busca informació sobre les llibreries natives carregades. Quina funcionalitat ens proporciona cadascuna d'elles? Quina funció realitzen els fitxers .ini que s'han creat?

4. Treball a realitzar

Activitat 5

- Instal·la un **servidor mysql LOCAL** al **servidor d'aplicacions**, instala el mòdul **php-Mysql** i executa el virtual host creat a la primera part que conte la referència a la funció **phpinfo()** i comprova que els mòduls **pdo**, **mysql** i **mysqlnd** es troben carregats. *Pega una captura de pantalla que permeti verificar-ho*
- Du a terme el desplegament de la **següent aplicació web**, per a això hauràs



d'importar la base de dades `crm_db` present en el repositori i crear un usuari que només siga accessible de manera local i amb permisos DML (SELECT, UPDATE, INSERT,...) sobre aquesta base de dades. (Si ho prefereixes desplega una aplicació desenvolupada en el mòdul de PES que utilitze Accés a base de dades)

- Per a finalitzar, habilita la connexió remota al servidor mysql:

```
bind-address = 0.0.0.0
```

- Crea un usuari anomenat `admin_db` que tinga tots els privilegis sobre la base de dades `crm_db` que permetisca la connexió des de qualsevol host.

- Instal·la l'aplicació **MysqlWorkBench** (o equivalent) i estableix una connexió amb l'usuari de creat al punt anterior.

5. Referències

- Digital Ocean. Instal·lar un entorn LEMP.
["https://www.digitalocean.com/community/tutorials/how-to-install-linux-nginx-mysql-php-lemp-stack-ubuntu-18-04"](https://www.digitalocean.com/community/tutorials/how-to-install-linux-nginx-mysql-php-lemp-stack-ubuntu-18-04)
- dev.mysql.com. Documentació oficial Mysql.
["https://dev.mysql.com/doc/refman/5.7/en/"](https://dev.mysql.com/doc/refman/5.7/en/)
- php.net."Documentació oficial Mysql Native Driver de PHP".
<https://www.php.net/manual/en/book.mysqlnd.php>
- Digital Ocean.Com establir una SBD remot per a optimitzar el rendiment mitjançant mysql. "<https://www.digitalocean.com/community/tutorials/how-to-set-up-a-remote-database-to-optimize-site-performance-with-mysql-on-ubuntu-18-04>