

# LINUX SERVER ADMINISTRATION

## DOCUMENTATIE CURS

DOCUMENTATIE

INTREABA PROFESORUL

CURSURILE MELE

8 Serverul MySQL » 8.3 Configurare MySQL » 8.3.1 Configurare de baza

- 1. Shell Scripts
- 2. Linux Kernel
- 3. Serverul DHCP
- 4. Serverul FTP
- 5. NFS - Network File System
- 6. Serverul DNS
- 7. Serverul Apache
- 8. Serverul MySQL
  - 8.1 Prezentare generala
  - 8.2 Instalare MySQL
  - 8.3 Configurare MySQL
    - 8.3.1 Configurare de baza
    - 8.3.2 PHP & MySQL
- 9. NETFILTER
- 10. Sistemul de e-Mail
- 11. Serverul Postfix
- 12. Serverul POP/IMAP
- 13. Managementul Logurilor
- 14. Exemple practice (Ubuntu 14.04 LTS)
- 15. Webmin

### Configurare de baza

Fisierul de configurare atat pentru server ( `mysqld` ), pentru client ( `mysql` ) cat si pentru scriptul `mysqld_safe` este default `/etc/mysql/my.cnf`

Sectiunea `[mysqld]` sau `[server]` identifica directivele de configurare pentru server, sectiunea `[client]` sau `[mysql]` identifica directivele pentru client, iar sectiunea `[mysqld_safe]` identifica directivele pentru scriptul de initializare `mysqld_safe`.

Exemplu `/etc/mysql/my.cnf`

```
[client]
port      = 3306
socket    = /var/run/mysql/mysql.sock

[mysqld_safe]
socket    = /var/run/mysql/mysql.sock
nice      = 0

[mysqld]
user      = mysql
pid-file  = /var/run/mysql/mysql.pid
socket    = /var/run/mysql/mysql.sock
port      = 3306
basedir  = /usr
datadir  = /var/lib/mysql
tmpdir    = /tmp
lc-messages-dir = /usr/share/mysql
skip-external-locking
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address      = 127.0.0.1
#
# * Fine Tuning
#
key_buffer        = 16M
max_allowed_packet = 16M
```

Detalii:

`#` (diez) -> liniile care incep cu `#` reprezinta un comentariu;

`[mysqld]` -> identifica inceputul sectiunii de configurare pentru server;

`datadir` -> reprezinta directorul in care se gasesc fizic fisierele care reprezinta bazele de date. Acest director trebuie sa fie owned de userul sub care ruleaza `mysqld`;

`socket` -> reprezinta fisierul de tip socket prin care clientul comunica cu serverul daca acestea se gasesc pe acelasi host;

`user` -> user-ul sub care ruleaza serverul (procesul `mysqld`). Acesta trebuie sa fie un user neprivilegiat folosit doar pt. serverul MySQL;

`bind-address = 127.0.0.1` -> in cazul in care clientul `mysql` se afla pe aceeaasi masina fizica cu serverul, se recomanda dezactivarea suportului de retea in `mysqld` (acesta nu mai asculta in retea). Comunicatia dintre client si server se realizeaza printr-un socket. In acest fel se elimina incercarile neautorizate din retea de accesare a serverului. In cazul aplicatiilor web (web site-uri), clientul `mysql` este php, iar atat php cat si `mysqld` se gasesc pe acelasi calculator. Suportul de retea poate fi dezactivat; `[client]` -> identifica inceputul sectiunii de configurare pentru client (`mysql`);

`port` -> seteaza portul la care se conecteaza clientul;

`[mysqld_safe]` -> identifica inceputul sectiunii de configurare pentru scriptul de initializare al serverului;

Configurarea de baza a serverului MySQL presupune:

1. Stabilirea username-ului sub care ruleaza serverul. Acesta nu trebuie sa fie un user generic precum `nobody` ci un user folosit special si doar pentru `mysql`.

In fisierul de configurare `/etc/mysql/my.cnf`: `user=mysql`

2. Setarea ca owner pentru directorul in care se gasesc bazele de date ( `/var/lib/mysql` ) a userului sub care ruleaza serverul.

Se ruleaza: `chown -R mysql /var/lib/mysql`

### 3. Initializarea bazelor de date `mysql` si `test`

Serverul MySQL pastreaza informatii despre userii care se conecteaza la acesta, permisiunile fiecaruia etc tot intr-o baza de date.

Aceasta se numeste `mysql.test` este o baza de date folosita pentru benchmarking sau alte teste.

Se ruleaza: `mysql_install_db` care este un script ce creeaza bazele de date mysql si test

### 4. Securizarea serverului MySQL

Serverul MySQL autentifica fiecare conexiune pe baza unui `cont`. Un cont mysql este format din `username+parola+host`, unde host reprezinta IP-ul sau numele hostului de unde se conecteaza userul.

**Exemplu:** un cont pentru conectarea la server poate fi: `dan+passtest+localhost`, iar `dan+passtest+192.168.0.1` reprezinta un cu totul alt cont fara nicio legatura cu primul. Acest concept este fundamental pentru intelegerea modului in care MySQL functioneaza.

## Important

Serverul MySQL instalat default adauga o grava problema de securitate. Implicit exista 3 conturi de root fara parola. Acestea sunt `root+PAROLA_LIPSA+localhost`, `root+PAROLA+LIPSA+127.0.0.1` si `root+PAROLA_LIPSA+hostname`. Userii folositi de mysql sunt "mysql only users". Acestia nu trebuie sa exista in `/etc/passwd` si se folosesc doar pentru conectarea la server. Implicatia modului in care mysql opereaza default este ca ORICE user se poate conecta ca root de mysql la server si are acces complet.

Securizarea minimala a serverului presupune stergerea userilor root sau setarea unei parole pentru acestia.

Etape (serverul mysqld trebuie sa ruleze):

a) conectarea la server de pe localhost folosind userul root fara parola

```
mysql -h localhost -u root
```

b) alegerea bazei de date mysql pentru urmatoarele comenzi

```
mysql>use mysql;
```

c) setarea de parole pentru userii root sau stergerea acestora

Userii de mysql impreuna cu toate privilegiile acestora se gasesc in tabelul `user` din baza de date `mysql`.

Colectarea de informatii:

- pentru vizualizarea structurii tabelii `user` se ruleaza: `mysql>describe user;`
- coloanele care prezinta interes sunt: `user`, `password` si `host`
- pentru vizualizarea conturilor de root se ruleaza: `select user,password,host from user;`

Setare parola pentru un cont: `mysql>set password for root@127.0.0.1=password('parola_dorita');`

Stergere cont root fara parola: `mysql>delete from user where user='root' and password='';`

Crearea unui nou cont(`admin+mypass+192.168.0.1`) cu setarea de privilegii: `mysql>grant all privileges on mysql.* to admin@192.168.0.1 identified by 'mypass';`

mysql reprezinta baza de date pe care userul va avea privilegiile, iar `*` inlocuieste orice tabel din baza de date. Dupa executarea comenzii de mai sus userul admin cu parola mypass care se conecteaza la server de la 192.168.0.1 va avea drepturi totale pentru toate tabellele din baza de date mysql. Acesta este echivalent cu root.

Pe Ubuntu (sau distributii derivate) modificarea parolei userului root al serverului MySQL se poate realiza si cu `dpkg-reconfigure`.

**Exemplu:**`sudo dpkg-reconfigure mysql-server-5.5`

## Nota

1. Dupa crearea sau modificarea informatiilor despre un cont se recomanda rularea comenzii `mysql>flush privileges;` si apoi restartarea serverului.



2. Un user se conecteaza la server folosind comanda `mysql` executand din consola: `mysql -h IP_SERVER_MYSQL -u USERNAME -p`, unde `-h` identifica serverul, `-u` username-ul cu care se conecteaza, iar `-p` faptul ca acesta are parola setata.

Important pentru securizarea serverului, este si folosirea optiunii `bind-address = 127.0.0.1` in `my.cnf` din sectiunea serverului. Astfel mysqld nu va mai accepta conexiuni din retea. Daca folosim aplicatii web care se gasesc pe acelasi calculator cu serverul mysqld, suportul de retea poate fi dezactivat.

[Resurse](#)

- [MySQL Usernames and Passwords](#)
- [Securing the Initial MySQL Accounts](#)

Configurare server MySql



Durata: 5.45 min  
Marime: 1.08MB