

Navigation

[1. Installation des binaires](#)

[2. Installation des sources](#)

[2.1 Pré-requis](#)

[2.2 Compilation](#)

[3. Configuration](#)

[4. Administration](#)

[5. Administration via la console](#)

Installation et configuration de mysql 4.x

MySQL est un serveur très apprécié et utilisé par les webmasters. C'est une SGBDR (Système de gestion de base de données relationnelles) très rapide, qui supporte en grande partie le standard SQL 2. Il propose en marge, quelques extensions à SQL2, très utiles sur internet. Il est basé sur une exécution client/serveur qui se compose d'un démon, le serveur (mysqld) et de nombreux clients différents : programmes/libraries. Ce document détaille l'installation de MySQL 4.x.

1. Installation des binaires :

J'explique ici l'installation de mysql depuis les paquets de votre distribution. Si votre distribution n'est pas étudiée, recherchez dans vos cdroms des paquets aux noms similaires, ou utilisez l'installation à partir des sources. Dans tous les cas, la configuration qui suivra, fera la part belle aux sources, par soucis de généralité. Reportez-vous à la doc de votre distribution pour les détails.

- **Pour les utilisateurs de Mandriva** : Installez les paquets mysql, mysql-bench, mysql-client
- **Pour les utilisateurs de RedHat** : Installez les paquets mysql-shared mysql-client mysql php-mysql
- **Pour les utilisateurs de Debian** : Installez les paquets mysql-client mysql-commonmysql-server

2. Installation des sources :

Téléchargez la version la plus à jour de ce dernier [ICI](#). Attention, la bonne version est 4.0.x même si la version 4.1.x existe.

2.1 Pré-requis :

Est-il nécessaire de préciser que tous, mais vraiment tous les paquets, se rapportant de près ou de loin à mysql, doivent être désinstallés auparavant ? Vous devez également avoir un utilisateur mysql. Pour être fixé :

```
cat /etc/passwd | grep mysql
```

Si vous n'aviez aucune réponse, c'est que vous n'avez pas d'utilisateur mysql, vous pouvez le créer par :

```
# groupadd mysql
# useradd -g mysql -d /home/mysql mysql
```

2.2 Compilation :

J'ai choisi de mettre les bases de données dans la partition /home/, parce que c'est la plus grosse chez moi. Vous êtes libres, d'adapter à votre convenance. La compilation de mysql se résume à :

```
# tar -xzvf /où_est/mysql-4.0.23a.tar.gz
# cd mysql-4.0.23a/
# ./configure --enable-assembler --with-unix-socket-
path=/tmp/mysql.sock --prefix=/ --exec-prefix=/usr --
libexecdir=/usr/sbin --sysconfdir=/etc --datadir=/usr/share --
localstatedir=/home/mysql --infodir=/usr/info --
includedir=/usr/include --mandir=/usr/man --enable-large-files --
with-innodb --with-low-memory --with-charset=latin1 --enable-
local-infile --with-extra-charsets=all --enable-thread-safe-client
--with-client-ldflags=-lstdc++ --with-comment="MySQL by Kernel"
# make
# make install
```

Vos bases de données seront donc dans /home/mysql.

3. Configuration :

Créez maintenant une base Mysql, cette étape est indispensable :

```
# ./scripts/mysql_install_db
```

Donnez les droits à l'utilisateur mysql :

```
# chown -R mysql /home/mysql
# chgrp -R mysql /home/mysql
```

Adaptez votre fichier de configuration :

```
# cp support-files/my-medium.cnf/etc/my.cnf
```

En réalité, tout dépend du type de serveur mysql que vous souhaitez, si vous avez plus de 512 Mo, utilisez my-large.cnf et si vous êtes vraiment un acharné, n'hésitez pas à aller jusqu'à my-huge.cnf. Pour la plus part des configurations my-medium.cnf suffira. Editez, /etc/my.cnf, dans la section [mysqld], rajoutez : user = mysql.

Si vous souhaitez accéder à vos tables depuis un hôte distant ou en utilisant JDBC et le driver MySQL, vous devez commenter l'option skip-networking dans le fichier /etc/my.cnf.

Pour améliorer les performances de votre serveur mysql, vous pouvez taper la commande :

```
# strip /usr/sbin/mysqld
```

Vous pouvez désormais lancer mysql par :

```
# /usr/bin/mysqld_safe &
```

Et tester par ...

```
# /usr/bin/mysqladmin version
```

```
# /usr/bin/mysqladmin variables
```

```
# /usr/bin/mysqlshow
```

```
# /usr/bin/mysqlshow mysql
```

```
# /usr/bin/mysql -e "select host,db,user from db" mysql
```

Assignez un mot de passe au root de mysql (à ne pas confondre avec le root du système) :

```
# /usr/bin/mysqladmin -u root password 'xxxxxxxx'
```

Pour certaines versions de mysql vous obtiendrez un message d'erreur équivalent à celui-ci :

You cannot use 'password' command as mysqld runs with grant tables disabled (was started with --skip-grant-tables). Use: "mysqladmin flush-privileges password '*'" instead

Le problème vient du fait que les version récentes demysql se lancent par défaut avec vérification de droits sur les tables. Donc pas de panique, c'est normal, tapez simplement à la place :

```
# mysqladmin -u root flush-privileges password 'xxxxxxx'
```

Pour que mysql soit lancé à chaque démarrage, tapez d'abord les lignes suivantes dans un terminal :

```
# cd support-files/
```

```
# cp mysql.server /usr/bin/
```

```
# chmod 755 /usr/bin/mysql.server
```

Puis rajoutez, les lignes suivantes dans votre /etc/rc.d/rc.local:

```
if [ -x /usr/bin/mysql.server ]; then
```

```
  /usr/bin/mysql.server start
```

```
fi
```

Notez que le script /usr/bin/mysql.server vous permet de gérer le daemond mysqld, "start" pour démarrer, "stop" pour l'arrêter. Pour redémarrer, mysql vous pourrez taper à tout moment :

```
# /usr/bin/mysql.server stop
```

```
# /usr/bin/mysql.server start
```

Vous trouverez de la documentation sur Mysql, chez [nexen](#)

4. Administration :

Pour administrer Mysql, je vous conseille l'excellent [PHPMyAdmin](#), qui comme son nom le laisse penser, nécessite PHP.

5. Administration via la console :

- Faire une sauvegarde d'une base :

```
mysqldump -u login -ppassword ma_base > my_db-2005-03-06.sql
```

- Faire une sauvegarde de toutes les bases d'un serveur:

```
mysqldump -u login -ppassword --all-databases >all_db-2005-03-06.sql
```
- se connecter à la console mysql en tant qu'utilisateur root :

```
mysql -h localhost -u root -p
```
- l'état du SGBD :

```
\s
```
- exécuter un script sql (dans le répertoire courant) :

```
\. monscript.sql
```
- liste des bases de données :

```
SHOW DATABASES;
```
- Créer la base de données trustonme :

```
CREATE DATABASE trustonme;
```
- accéder à la base de données trustonme :

```
\u trustonme
```
- liste des tables de la base de données mysql :

```
\u mysql  
SHOW TABLES;
```
- la structure de la table db dans la base de données mysql :

```
\u mysql  
DESCRIBE db;
```
- rajouter 3 utilisateurs, une fois connecté en tant que root :

```
GRANT ALL PRIVILEGES ON *.* TO kernel@localhost IDENTIFIED BY 'mdp' WITH  
GRANT OPTION;  
GRANT ALL PRIVILEGES ON *.* TO kernel@%" IDENTIFIED BY 'mdp' WITH GRANT  
OPTION;  
GRANT RELOAD,PROCESS ON *.* TO dbz@localhost;  
GRANT USAGE ON *.* TO songoku@localhost;
```

Ces commandes GRANT ajoutent trois nouveaux utilisateurs :

- **kernel** : Un super-utilisateur qui peut se connecter au serveur d'où il veut, mais qui doit utiliser le mot de passe 'mdp' pour le faire. Notez que nous devons exécuter une commande GRANT pour kernel@localhost et kernel@%". Si nous n'ajoutons pas l'entrée avec localhost, l'entrée concernant l'utilisateur anonyme pour localhost qui est créée par mysql_install_db prendra précedence lors de la connexion à partir de l'hôte local, car elle a une entrée plus spécifique pour la valeur du champ Host et de plus, elle vient en premier dans l'ordre de tri de la table user.
- **dbz** : Un utilisateur qui peut se connecter depuis localhost sans mot de passe et qui a les droits administratifs RELOAD et PROCESS. Cela permet à cet utilisateur d'exécuter les

commandes `mysqladmin reload`, `mysqladmin refresh`, et `mysqladmin flush-*`, ainsi que `mysqladmin processlist`. Aucun droit lié aux bases de données n'est donné. (Ils peuvent l'être plus tard en utilisant d'autres instructions `GRANT`.)

- **songoku** : Un utilisateur qui peut se connecter sans mot de passe, mais seulement à partir de l'hôte local. Les droits globaux sont tous à 'N' -le type de droit `USAGE` vous permet de créer un utilisateur démunie de privilèges. Il est supposé que vous lui assignerez les droits spécifiques aux bases de données plus tard.
- liste des utilisateurs :
`\u mysql`
`SELECT Host, User, Password FROM user;`
- attribuer un mot de passe à dbz :
`SET PASSWORD FOR dbz@localhost=PASSWORD("le_mot_de_passe");`
- supprimer l'utilisateur songoku :
`\u mysql`
`DELETE FROM user WHERE User="songoku";`
- liste des options :
`\h`
- quitter :
`\q`