

Установка MySQL

Автор: lissyara.

Оригинал: <http://www.lissyara.su/articles/freebsd/programms/mysql/>

Опять статья навеянная многочисленными просьбами трудящихся :) Заодно не придётся во всех новых статьях описывать установку MySQL :) Так что, в любом случае пригодится. Вообще, с MySQL вроде всё просто, но почему-то народ активно морозится баз данных, используя системных юзеров в почте, да всякие файлы текстовые, под почту да хранение трафика. Хотя с БД оно всё на порядок удобней получается.

Итак, рассматривать будем установку и небольшую настройку двух версий MySQL - 5.0 и 3.23 под FreeBSD6.0. Почему именно они? На данный момент, 5.1 вроде всё ещё бета, потому используется на рабочих серверах только отъявленными маньяками, или теми у кого есть необходимость. А вот 3.23 нужна в случаях, когда на какой-нить старенькой машинке нужна "записная книжка" для почтовых юзеров, да БД куда складывать статистику по трафику инета, например. Подымать для этого 5.0 или 4.1 - расточительство оперативки и ресурсов слабенькой машинки. У меня и на некоторых, очень хороших машинах, стоит 3.23 - ибо его хватает.

Итак, ставим 5.0/`usr/home/lissyara/>cd /usr/ports/databases/mysql50-server/`

`/usr/ports/databases/mysql50-server/>make`

`====> Vulnerability check disabled, database not found`

You may use the following build options:

`WITH_CHARSET=charset` Define the primary built-in charset (latin1).

`WITH_XCHARSET=list` Define other built-in charsets (may be 'all').

`WITH_COLLATION=collate` Define default collation (latin1_swedish_ci).

`WITH_OPENSSL=yes` Enable secure connections.

`WITH_LINUXTHREADS=yes` Use the linuxthreads pthread library.

`WITH_PROC_SCOPE_PTH=yes` Use process scope threads

(try it if you use libpthread).

`BUILD_OPTIMIZED=yes` Enable compiler optimizations

(use it if you need speed).

`BUILD_STATIC=yes` Build a static version of mysqld.

(use it if you need even more speed).

`WITHOUT_INNODB=yes` Disable support for InnoDB table handler.

`WITH_ARCHIVE=yes` Enable support for Archive Storage Engine.

`WITH_FEDERATED=yes` Enable support for Federated Storage Engine.

`WITH_NDB=yes` Enable support for NDB Cluster.

`====> Extracting for mysql-server-5.0.22`

`^C`

`/usr/ports/databases/mysql50-server/>`

Установка MySQL

Автор: Administrator

11.10.2006 03:42 - Обновлено 28.05.2010 13:49

Прервал. Доступна куча опций, потому с ними и поиграемся, для чего в файл /etc/make.conf надо внести такие строки:# Для некоторых, особо тупых приложений, лучше указать версию MySQL,
что используется, тут. Но - для совсем тупых не поможет и это.
DEFAULT_MYSQL_VER=50

Директория где лежат порты
PORTSDIR?= /usr/ports

для сервера

.if \${.CURDIR} == \${PORTSDIR}/databases/mysql50-server

Дефолтовая кодировка. Вообще, этим пунктом увлекаться не стоит -
могут возникнуть проблемы при переносе на другой сервер. На самом
деле и не проблемы вовсе - просто дампы перекодировать да поменять
кодировку-коллэйшен у таблиц в дампе, перед заливкой, но первый
раз столкнувшись, вызывает кучу проблем...

WITH_CHARSET=cp1251

Другие вкомпиленные кодировки. Можно задать 'all' - все.

#WITH_XCHARSET=all

Дефолтовая кодировка сравнения (другого слова подобрать не могу...)

WITH_COLLATION=cp1251_bin

Поддержка OpenSSL - для шифрования передаваемых данных.

Шифрование - это конечно хорошо, но - повышается нагрузка на
машину сервера и на машину клиента (если и тот и другой на одной
машине - вообще не вижу смысла в этом пункте.)

#WITH_OPENSSL=yes

Интересная опция - позволяет работать в несколько "нитей",
обрабатывая несколько запросов одновременно - в результате
повышается производительность.

WITH_LINUXTHREADS=yes

Программные треды FreeBSD (если я верно всё понял). Тестов на скорость,
в сравнении с линуксовыми не нашлось - потому не знаю, кто быстрее.
Кстати, видел рекомендацию - включать и те и другие треды одновременно.
Смысла, особого в этом не вижу, но и не утверждаю ничего.

#WITH_PROC_SCOPE_PTH=yes

Собирать с флагами оптимизации (-O2), сам не мерял, но по слухам,
несколько процентов производительности можно выиграть. Будет
полезным на старых машинах, или высоконагруженных.

BUILD_OPTIMIZED=yes

Собрать статическую версию mysqld (со вкомпиленными либами, чтоль)
Также, как и предыдущая опция, позволит выиграть несколько процентов
производительности (по слухам - до 10% - но чё-то слабо верится...)

BUILD_STATIC=yes

Отключить тип таблиц InnoDB (если не используете - будет поменьше
коду, и, соответственно, быстрее работать будет)

WITHOUT_INNODB=yes

Установка MySQL

Автор: Administrator

11.10.2006 03:42 - Обновлено 28.05.2010 13:49

```
# Специальный тип хранения данных (не всех, тока тех, что без индексов),
# позволяет хранить их в виде архива, тем самым экономится дисковое
# пространство (в ущерб производительности, я думаю... Так что смысл
# есть, лишь если надо хранить кучу неиндексированных данных -
# те же данные по траффику, к примеру, там всё равно перебором почти всё :)
#WITH_ARCHIVE=yes
# Фенька, позволяющая работать с удалёнными (находящимися на другом хосте)
# таблицами данных, как будто они находятся на локальной машине
# (NFS, чтоль, уже отменили? :) Хотя, если выборки будут идти на другом
# хосте - то нагрузка на сеть будет меньше, чем с NFS. Короче - не пробовал,
# ничё не утверждаю... Пару строк в документации прочёл.)
#WITH_FEDERATED=yes
# Опция, нужная лишь в случае, если будете собирать кластер MySQL-серверов
#WITH_NDB=yes
#endif
# для клиента
.if ${.CURDIR} == ${PORTSDIR}/databases/mysql50-client
# Многие из опция сервера, применяются и в клиенте. Вобщем-то, какие не
# применяются, интуитивно понятно - то, что касается всяких фенек, типа
# кластеров, архивных таблиц и прочего. Ставить их тут можно, но они не
# произведут никакого действия. Также, хочу заметить, что кодировка,
# 'по-умолчанию' бывает не тока у сервера, но и у клиента. Частенько,
# именно на этом прокалываются. (И вовсе не обязательно она должна быть
# такая же как у сервера.)
# Также хочу заметить, что клиента не надо собирать с какими бы то ни
# было тредами - линуксовыми, или родными - я прокололся на линуксовых,
# вроде всё пашет, с консоли, а вот апач падает... Методом исключения
# выяснил виновника - клиент с тредами - падает mysql-модуль PHP,
# валит апача...
WITH_CHARSET=cp1251
WITH_COLLATION=cp1251_bin
BUILD_OPTIMIZED=yes
#endif
```

Итак, собираем MySQL: /usr/home/lissyara/>cd /usr/ports/databases/mysql50-server/
/usr/ports/databases/mysql50-server/>make && make install && make clean

```
После инсталляции клиента сборка прекращается с ошибкой - не может найти
библиотеку.==> Installing ldconfig configuration file
==> Compressing manual pages for mysql-client-5.0.22
==> Registering installation for mysql-client-5.0.22
==> Returning to build of mysql-server-5.0.22
Error: shared library "mysqlclient.15" does not exist
*** Error code 1
```

Установка MySQL

Автор: Administrator

11.10.2006 03:42 - Обновлено 28.05.2010 13:49

Stop in /usr/ports/databases/mysql50-server.

```
/usr/ports/databases/mysql50-server/>
```

Я перезагрузился, помогло, но можно и иначе, дать команду:
/usr/home/lissyara/>/sbin/ldconfig -m /usr/local/lib/mysql
/usr/home/lissyara/>

Должно помочь и без перезагрузки. Вообще, раньше, вместе с клиентом, инсталлся такой скрипт:

```
/usr/local/etc/rc.d/000.mysql-client.sh#!/bin/sh
```

```
case "$1" in
start)
/sbin/ldconfig -m /usr/local/lib/mysql
;;
stop)
;;
*)
echo ""
echo "Usage: `basename $0` { start | stop }"
echo ""
exit 64
;;
esac
```

Нынче он не инсталлится. Если ошибки будут продолжаться, при сборке других приложений из портов, то можно его добавить в автозагрузку, или, что грамотней, добавить путь в /etc/rc.conf (подробности можно найти здесь, поиск в странице, по ключевому слову ldconfig_paths).

После чего продолжаем инсталляцию:
/usr/ports/databases/mysql50-server/>make install &&
make clean

По окончании инсталляции, стругаем конфиг для mysql - /usr/local/etc/my.cnf# Этот конфигурационный файл сделан на основе файла для маленьких # систем - /usr/local/share/mysql/my-small.cnf. Большую часть его # делал не я, а один знакомый. Имени, к сожалению, не помню... # Но всё же предупреждаю - копирайт на настройки конфига не мой :) # # Вообще, конфигурационный файл можно положить в несколько мест: # /etc/my.cnf # /var/db/mysql/my.cnf # /usr/local/etc/my.cnf

Установка MySQL

Автор: Administrator

11.10.2006 03:42 - Обновлено 28.05.2010 13:49

```
# В любом из них mysqld его найдёт, и достанет из него настройки.  
# Самое корректное место, если судить по стартовому скрипту, это  
# директория где лежат базы данных - /var/db/mysql/my.cnf.
```

```
# Опции для всех клиентов MySQL
```

```
[client]
```

```
# Пароль для подключения к БД
```

```
#password = your_password
```

```
# Порт на котором висит MySQL
```

```
port = 3306
```

```
# Сокет MySQL
```

```
socket = /tmp/mysql.sock
```

```
# Опции MySQL-сервера
```

```
[mysqld]
```

```
# Порт
```

```
port = 3306
```

```
# Адрес, который будем слушать (если вам не нужно подключаться к
```

```
# MySQL с других машин, то оставьте здесь 127.0.0.1)
```

```
bind-address = 127.0.0.1
```

```
# Где лежит сокет
```

```
socket = /tmp/mysql.sock
```

```
# Не использовать средства системных блокировок.
```

```
skip-locking
```

```
# Вообще, в новых версиях, (после 3.21) этот пункт правильно называется
```

```
# key_buffer_size, но можно использовать и старое имя. Значение этого
```

```
# пункта - размер буфера, используемого для блоков индексов. Чтобы
```

```
# улучшить обработку индексов (для всех операций чтения и записи нескольких
```

```
# элементов), необходимо увеличить это значение настолько, насколько возможно.
```

```
# Рекомендуется, 1/4 от объёма оперативки, но не более 1/2 - иначе система
```

```
# может начать сохранять временные файлы на диске, что значительно
```

```
# снизит производительность.
```

```
key_buffer = 16K
```

```
# Максимальный размер одного пакета. Изначально размер буфера сообщений
```

```
# устанавливается в net_buffer_length байтов, но при необходимости может
```

```
# возрасти до max_allowed_packet байтов. Это значение по умолчанию не
```

```
# настолько велико, чтобы отсеивать большие (возможно ошибочные) пакеты.
```

```
# Если используются большие столбцы BLOB, его необходимо увеличить.
```

```
# Значение должно быть не меньше самого большого BLOB, который будет
```

```
# использоваться. Ограничение протокола для max_allowed_packet
```

```
# составляет 16 Мб в MySQL 3.23 и 1Гб в MySQL 4.0.
```

```
max_allowed_packet = 1M
```

```
# Количество открытых таблиц для всех потоков. С увеличением этого
```

```
# значения увеличивается количество дескрипторов файлов, необходимых
```

Установка MySQL

Автор: Administrator

11.10.2006 03:42 - Обновлено 28.05.2010 13:49

```
# для mysqld. Чтобы узнать, необходимо ли изменять значение кэша таблиц,
# следует проверить значение переменной Opened_tables.
# Если у этой переменной большое значение, а команда FLUSH TABLES
# (которая закрывает все таблицы, а потом открывает их повторно)
# используется не часто, то необходимо увеличить ее значение.
table_cache = 4
# Каждый поток, которому необходимо произвести сортировку, выделяет
# буфер данного размера. Увеличение данного значения позволит ускорить
# выполнение операторов ORDER BY или GROUP BY.
sort_buffer_size = 64K
# Каждый поток, осуществляющий последовательное сканирование, выделяет
# буфер указанного размера для каждой сканируемой таблицы. Если
# проводится много последовательных сканирований, это значение
# можно увеличить.
read_buffer_size = 256K
# При считывании строк, после проведения сортировки, в отсортированном
# порядке строки считываются через буфер, чтобы избежать операций поиска
# по диску. Это может улучшить выполнение ORDER BY весьма и весьма,
# если параметр установлен в большое значение. Т.к. эта переменная
# имеет отношение к потоку, то не устанавливайте слишком большое
# значение глобально, но просто меняйте его при выполнении некоторых
# больших запросов.
read_rnd_buffer_size = 256K
# В данное значение устанавливается, в промежутках между запросами,
# буфер соединения. Обычно это значение не изменяется, но если у вас
# очень мало памяти, можно установить его по размеру ожидаемого
# запроса (т.е. равным предполагаемой длине операторов SQL, отправляемых
# клиентами; если оператор превысит указанную длину, буфер будет
# автоматически увеличен как максимум до max_allowed_packet байтов).
net_buffer_length = 2K
# Размер стека для каждого потока. От данного значения зависит большое
# количество ограничений, обнаруживаемых при помощи теста crash-me.
# По умолчанию этот размер достаточен для нормальной работы.
thread_stack = 64K

# Вообще не слушать порты TCP/IP. Это может применяться для большей
# безопасности, если все процессы, соединяющиеся с MySQL висят на том же
# хосте, что и mysqld. Все взаимодействия с mysqld будут осуществляться
# через Unix-сокеты, или именованные каналы.
# Заметьте, что использование этой опции под форточками, без включения
# именованных каналов (используйте опцию "enable-named-pipe") сделает
# работу MySQL бесполезной - ибо с mysqld никто не сможет соединиться :)
skip-networking
# Если Вы используете InnoDB, то прокомментируйте эту опцию
skip-innodb
# С этой опцией MySQL не будет инициализировать библиотеку Berkeley DB,
```

Установка MySQL

Автор: Administrator

11.10.2006 03:42 - Обновлено 28.05.2010 13:49

```
# что позволит сэкономить большое количество памяти.
skip-bdb
# Некоторое уникальное число между 2 и 2^32-1. Значения server-id должны
# быть различными на каждом сервере, участвующем в репликации. Если
# значение server-id не определено, оно будет установлено в 1, если
# также не определено значение master-host, оно будет установлено в 2.
# Обратите внимание, что если значение server-id опущено, то головной
# сервер будет отказывать в соединении всем подчиненным серверам, а
# подчиненный сервер - отказывать в соединении головному серверу.
# Таким образом, опускать установку значения server-id можно лишь в
# случае резервного копирования с использованием двоичного журнала.
server-id      = 1
# Раскомментируйте эту опцию, для включения логгирования всех запросов
# Заметьте - тока на время отладки! Потом надо закомментировать и
# рестартануть MySQL!
# Файл должен существовать, с соответствующими правами на него:
# touch /var/log/mysql.log
# chown mysql:wheel /var/log/mysql.log
# chmod 640 /var/log/mysql.log
log            = /var/log/mysql.log

# Указывает местоположение двоичного журнала обновлений,
# в котором будут вестись записи.
#log-bin=mysql-bin

[mysqldump]
# Если задан этот параметр, то обработчик таблицы при выполнении
# удаления не будет объединять индексы - в некоторых случаях это
# может ускорить данную операцию
quick
# Максимальная величина пакета, посылаемого/принимаемого с сервера
max_allowed_packet = 16M

[mysql]
# Отключает автоматическое рехеширование. rehash следует использовать
# для получения хеша таблиц и полей. Это обеспечивает более
# быстрый старт mysql.
no-auto-rehash
# Опция, которую рекомендуется раскомментировать начинающим :)
# Разрешает выполнять только операции UPDATE и DELETE, используя ключи.
#safe-updates

[isamchk]
key_buffer = 8M
sort_buffer_size = 8M
```

Установка MySQL

Автор: Administrator

11.10.2006 03:42 - Обновлено 28.05.2010 13:49

```
[myisamchk]
key_buffer = 8M
sort_buffer_size = 8M
```

```
[mysqlhotcopy]
# Допускать простой длительностью interactive_timeout секунд (вместо
# wait_timeout секунд) перед закрытием данного соединения.
interactive-timeout
```

```
# P.S. Большинство текста - это из мануала по MySQL 4.0, за который
# мы не так давно воевали на www.mysql.com (его убрали на некоторое
# время, типа он по старой версии, потому не актуален...
# но - отвоевали, вернули :)))
```

С таким конфиг-файлом mysqld занимает в 6 раз меньше памяти, чем без него. Итак, запускаем:

```
./usr/home/lissyara/>echo 'mysql_enable="YES"' >> /etc/rc.conf
./usr/home/lissyara/>./usr/local/etc/rc.d/mysql-server.sh start
Starting mysql.
./usr/home/lissyara/>
./usr/home/lissyara/>ps -axj | grep mysqld
mysql 44512 1 44510 504 0 S p0 0:00,08 /bin/sh /usr/local/bin
mysql 44530 44512 44510 504 0 SN p0 0:00,15 /usr/local/libexec/mys
mysql 44531 44530 44510 504 0 SN p0 0:00,00 /usr/local/libexec/mys
mysql 44532 44531 44510 504 0 SN p0 0:00,00 /usr/local/libexec/mys
mysql 44533 44531 44510 504 0 SN p0 0:00,00 /usr/local/libexec/mys
root 44548 507 44547 504 2 S+ p0 0:00,02 grep mysqld
./usr/home/lissyara/>
```

Так много процессов - потому как с тредами (на самом деле он один...).

С 5-кой разобрались. Теперь пример установки, для самой старой версии mysql, что есть в портах - 3.23.

```
./usr/home/lissyara/>cd /usr/ports/databases/mysql323-server/
./usr/ports/databases/mysql323-server/>make && make install && make clean
===> Vulnerability check disabled, database not found
```

You may use the following build options:

```
WITH_CHARSET=charset Define the primary built-in charset (latin1).
WITH_XCHARSET=list Define other built-in charsets (may be 'all').
WITH_OPENSSL=yes Enable secure connections.
WITH_LINUXTHREADS=yes Use the linuxthreads pthread library.
WITH_PROC_SCOPE_PTH=yes Use process scope threads
(tried it if you use libpthread).
BUILD_OPTIMIZED=yes Enable compiler optimizations
```

Установка MySQL

Автор: Administrator

11.10.2006 03:42 - Обновлено 28.05.2010 13:49

(use it if you need speed).

BUILD_STATIC=yes Build a static version of mysqld.

(use it if you need even more speed).

WITHOUT_INNOODB=yes Disable support for InnoDB table handler.

====> Extracting for mysql-server-3.23.59.n.20050301_3

^C

/usr/ports/databases/mysql323-server/>

Чтож, опции, в большинстве своём, точно такие же, как и у 5.0. Соответственно правим файл /etc/make.conf (приведено без комментов, ибо нового ничё нет - смотрите предыдущий листинг)# Для некоторых, особо тупых приложений, лучше указать версию MySQL,

что используется, тут. Но - для совсем тупых не поможет и это.

DEFAULT_MYSQL_VER=323

Директория где лежат порты

PORTSDIR?= /usr/ports

для сервера

.if \${.CURDIR} == \${PORTSDIR}/databases/mysql323-server

BUILD_OPTIMIZED=yes

BUILD_STATIC=yes

Единственный пункт, по которому нужны, наверно, пояснения.

Просто без него у меня не заводилось...

#WITHOUT_INNOODB=yes

.endif

для клиента

.if \${.CURDIR} == \${PORTSDIR}/databases/mysql323-client

.endif

После рихтовки /etc/make.conf снова запускаем

компиляцию:/usr/ports/databases/mysql323-server/>make && make install && make clean

Которая по окончании установки клиента вываливается с ошибкой, похожей на ту, что и у mysql 5.0:====> Installing ldconfig configuration file

====> Compressing manual pages for mysql-client-3.23.59.n.20050301_2

====> Registering installation for mysql-client-3.23.59.n.20050301_2

====> Returning to build of mysql-server-3.23.59.n.20050301_3

Error: shared library "mysqlclient.10" does not exist

*** Error code 1

Stop in /usr/ports/databases/mysql323-server.

Установка MySQL

Автор: Administrator

11.10.2006 03:42 - Обновлено 28.05.2010 13:49

```
/usr/ports/databases/mysql323-server/>
```

Лечение тоже самое - перезагрузка, или `/usr/home/lissyara/>/sbin/ldconfig -m`

```
/usr/local/lib/mysql
```

```
/usr/home/lissyara/>
```

После чего всё нормально доинсталлируется. Можно создавать конфиг `/usr/local/etc/my.cnf` # Этот конфигурационный файл сделан на основе файла для маленьких

```
# систем - /usr/local/share/mysql/my-small.cnf.
```

```
#
```

```
# Вообще, конфигурационный файл можно положить в несколько мест:
```

```
# /etc/my.cnf
```

```
# /var/db/mysql/my.cnf
```

```
# /usr/local/etc/my.cnf
```

```
# В любом из них mysqld его найдёт, и достанет из него настройки.
```

```
# Самое корректное место, если судить по стартовому скрипту, это
```

```
# директория где лежат базы данных - /var/db/mysql/my.cnf.
```

```
# Опции для всех клиентов MySQL
```

```
[client]
```

```
# Пароль для подключения к БД
```

```
#password = your_password
```

```
# Порт на котором висит MySQL
```

```
port = 3306
```

```
# Сокет MySQL
```

```
socket = /tmp/mysql.sock
```

```
# Опции MySQL-сервера
```

```
[mysqld]
```

```
# Порт
```

```
port = 3306
```

```
# Адрес, который будем слушать (если вам не нужно подключаться к
```

```
# MySQL с других машин, то оставьте здесь 127.0.0.1)
```

```
bind-address = 127.0.0.1
```

```
# Где лежит сокет
```

```
socket = /tmp/mysql.sock
```

```
# Не использовать средства системных блокировок.
```

```
skip-locking
```

```
# Вообще не слушать порты TCP/IP. Это может применяться для большей
```

```
# безопасности, если все процессы, соединяющиеся с MySQL висят на том же
```

Установка MySQL

Автор: Administrator

11.10.2006 03:42 - Обновлено 28.05.2010 13:49

```
# хосте, что и mysqld. Все взаимодействия с mysqld будут осуществляться
# через Unix-сокеты, или именованные каналы.
# Заметьте, что использование этой опции под форточками, без включения
# именованных каналов (используйте опцию "enable-named-pipe") сделает
# работу MySQL бесполезной - ибо с mysqld никто не сможет соединиться :)
skip-networking
# Если Вы используете InnoDB, то прокомментируйте эту опцию
skip-innodb
# С этой опцией MySQL не будет инициализировать библиотеку Berkeley DB,
# что позволит сэкономить большое количество памяти.
skip-bdb
# Некоторое уникальное число между 2 и 2^32-1. Значения server-id должны
# быть различными на каждом сервере, участвующем в репликации. Если
# значение server-id не определено, оно будет установлено в 1, если
# также не определено значение master-host, оно будет установлено в 2.
# Обратите внимание, что если значение server-id опущено, то головной
# сервер будет отказывать в соединении всем подчиненным серверам, а
# подчиненный сервер - отказывать в соединении головному серверу.
# Таким образом, опускать установку значения server-id можно лишь в
# случае резервного копирования с использованием двоичного журнала.
server-id      = 1
# Раскомментируйте эту опцию, для включения логгирования всех запросов
# Заметьте - тока на время отладки! Потом надо закомментировать и
# рестартануть MySQL!
# Файл должен существовать, с соответствующими правами на него:
# touch /var/log/mysql.log
# chown mysql:wheel /var/log/mysql.log
# chmod 640 /var/log/mysql.log
log            = /var/log/mysql.log

# Указывает местоположение двоичного журнала обновлений,
# в котором будут вестись записи.
#log-bin=mysql-bin
```

[mysqldump]

```
# Если задан этот параметр, то обработчик таблицы при выполнении
# удаления не будет объединять индексы - в некоторых случаях это
# может ускорить данную операцию
quick
```

[mysql]

```
# Отключает автоматическое рехеширование. rehash следует использовать
# для получения хеша таблиц и полей. Это обеспечивает более
# быстрый старт mysql.
no-auto-rehash
# Опция, которую рекомендуется раскомментировать начинающим :)
```

Установка MySQL

Автор: Administrator

11.10.2006 03:42 - Обновлено 28.05.2010 13:49

```
# Разрешает выполнять только операции UPDATE и DELETE, используя ключи.
```

```
#safe-updates
```

```
[isamchk]
```

```
key_buffer = 8M
```

```
sort_buffer_size = 8M
```

```
[myisamchk]
```

```
key_buffer = 8M
```

```
sort_buffer_size = 8M
```

```
[mysqlhotcopy]
```

```
# Допускать простой длительностью interactive_timeout секунд (вместо
```

```
# wait_timeout секунд) перед закрытием данного соединения.
```

```
interactive-timeout
```

Можно заметить, что отличия от 5.0 минимальны - отсутствуют опции буферов, да и всё, пожалуй. Запускаем: /usr/home/lissyara/>echo 'mysql_enable="YES"' >> /etc/rc.conf

```
/usr/home/lissyara/>/usr/local/etc/rc.d/mysql-server.sh start
```

```
Starting mysql.
```

```
/usr/home/lissyara/>ps -axj | grep mysql
```

```
mysql  8848  1 8846 504  0 S  p0  0:00,07 /bin/sh /usr/local/bi
```

```
mysql  8868 8848 8846 504  0 S  p0  0:00,06 /usr/local/libexec/my
```

```
root   8870  507 8869 504  2 L+  p0  0:00,01 grep mysql
```

```
root   8754 5368 8754 5344  1 S+  p1  0:00,02 tail -f /var/db/mysql
```

```
/usr/home/lissyara/>
```

Всё пашет.

Теперь общие вопросы. По дефолту, в MySQL заведено несколько пользователей, типа гостей, и прочих. Я их обычно удаляю - нефига гостям по БД

```
шляться.../usr/home/lissyara/>mysql --database=mysql --execute="SELECT COUNT(*) FROM user"
```

```
+-----+
```

```
| COUNT(*) |
```

```
+-----+
```

```
|    4    |
```

```
+-----+
```

```
/usr/home/lissyara/>mysql --database=mysql --execute="DELETE FROM user
```

```
? WHERE User=""
```

```
/usr/home/lissyara/>mysql --database=mysql --execute="SELECT COUNT(*) FROM user"
```

```
+-----+
```

```
| COUNT(*) |
```

```
+-----+
```

Установка MySQL

Автор: Administrator

11.10.2006 03:42 - Обновлено 28.05.2010 13:49

```
|    2 |
+-----+
/usr/home/lissyara/>mysql --database=mysql --execute="DELETE FROM user
? WHERE Host != 'localhost'"
/usr/home/lissyara/>mysql --database=mysql --execute="SELECT COUNT(*) FROM user"
+-----+
| COUNT(*) |
+-----+
|    1 |
+-----+
/usr/home/lissyara/>
/usr/home/lissyara/>mysql --database=mysql --execute="UPDATE user SET
? Password = PASSWORD('тут ввести пароль рута')"
/usr/home/lissyara/>mysql --database=mysql --execute="FLUSH privileges"
/usr/home/lissyara/>mysql --database=mysql --execute="SELECT * FROM user"
ERROR 1045: Access denied for user: 'root@localhost' (Using password: NO)
/usr/home/lissyara/>
```

Всё. Без пароля к БД не добаться. Вообще, общая рекомендация - если плохо знаете MySQL - либо займитесь изучением, либо поставьте графический интерфейс, типа phpMyAdmin. Я вначале пошёл по второму пути, щас пытаюсь свернуть на первый :))) Единственное - не забывайте его запаролить.

P.S. При проблемах (не запускается), смотрим лог ошибок:/usr/home/lissyara/>tail -f /var/db/mysql/^uname -a | awk '