



# Базовое Администрирование Linux

Занятие 9



Дмитрий  
Молчанов

# LAMP: php+mysql

---



1. Способы использования http-сервера и скрипто-языка
2. Mysql
  1. Настройки
  2. StorageEngines
  3. Минимальные административные команды

Самым ранним способом выполнения скриптов на стороне сервера является CGI – Common Gateway Interface.

- + Запустить можно скрипт на чем угодно. Главное, чтобы вернул валидный HTTP-ответ
- Накладные расходы на запуск процесса

# Apache + PHP

---



Наиболее простым и популярным способом является использование `mod_php` для `apache`.

- + есть почти во всех репозиториях
- + не надо ничего пересобирать
- + легко подключить/отключить

# Apache + PHP (as static module)



Apache позволяет использовать не только динамически подгружаемые модули, но и включать их «жестко» на этапе компиляции кода http-сервера.

+ это приносит некоторый прирост производительности, имеет смысл применять когда есть необходимость экономить на свичках

- требует отдельной сборки, тяжелее в поддержке.

# HTTP-server + FastCGI

---



Еще один из способов подключения.

Скрипт или среда выполнения скриптов (PHP) запускаются рядом и выполняются в фоне.

Общение между http-сервером и приложением происходит по протоколу FastCGI.

+ Позволяет одним http-сервером обслуживать несколько версий среды выполнения, например PHP или Python. Не перегружая при этом HTTP-сервер кучей модулей.

Очень популярная СУБД.

- Открытый исходный код/ Свободно распространяемое ПО.
- Кроссплатформенность.
- богатые возможности (покрывают большинство потребностей большинства проектов).
- простое обслуживание.

# MySQL: параметры

---



- Общение с сервером происходит по собственному бинарному протоколу
- по-умолчанию используется TCP/3306



- Разные способы хранения данных Storage Engines:
  - InnoDB
  - MyISAM
  - MEMORY
- Репликация «из коробки»

# MySQL: Storage Engines



- MyISAM – способ хранения данных по умолчанию:
  - + Простой, каждая таблица – 3 файла (описание, данные, индексы).
  - + Портативный, «ремонтпригодный»
  - + Позволяет делать копии на файловом уровне
  - - блокировка уровня таблицы
  - - отсутствие поддержки транзакций
  - - отсутствие поддержки внешних ключей, ссылочной целостности.

# MySQL: Storage Engines

---



- InnoDB
  - + Сточная блокировка
  - + Поддержка транзакций
  - + более устойчив к сбоям, за счет журналов транзакций
  - - слабо «ремонтпригоден», если ломается
  - - сложная организация хранения

Чаще всего его плюсы перекрывают его минусы при параллельном доступе к таблице на запись.

# MySQL: Storage Engines

---



- MEMORY
  - Как MYISAM, только в памяти
  - + за счет отсутствия дисковых операций очень быстр
  - - при перезагрузке данные не сохраняются
- BLACKHOLE
  - Можно использовать для ведения бинарного лога без сохранения данных

# MySQL: Параметры



- log-bin – ведение бинарных логов, необходимо для репликации и Point-in-Time восстановления.
- port – порт который работает сервер, по-умолчанию 3306
- bind-address – адрес на котором работает сервер. По-умолчанию – все.
- table\_cache – кэш открытых таблиц.
- thread\_cache – размер кэша потоков, чтобы
- query\_cache\* - управление query cache'ем.

# MySQL: Параметры

---



- для MyISAM
  - `key_buffer` – размер памяти выделяемый для работы с индексами
- для InnoDB
  - `innodb_buffer_pool_size` – размер памяти выделяемый для работы InnoDB
  - `innodb_file_per_table` – хранить данные таблиц в отдельных фалах (файл на таблицу)

- Master->Slave
  - 1 мастер может иметь много слейвов
  - 1 слейв может иметь получать данные только от одного мастера в один момент.
  - однопоточная
- Statement based
  - Каждая команда изменяющая данные выполненная на мастер-сервере выполняется на слейв-сервере.
- Row-based
  - Между серверами идет обмен «изменениями», результатами выполнения запросов.

# MySQL: настройки репликации.



- log-bin – ведение binary log'ов
- server-id – Идентификатор сервера.
- Команды
  - change master to master\_server='host', master\_user='user', master\_password='pass', master\_log\_file='file', master\_log\_pos=1234;
  - show master status;
  - show slave status;



# MySQL: минимальная диагностика

---



- mysqladmin
  - proccesslist
  - ping
  - status
- mysql
  - show global status
  - show [full] processlist
  - show databases
  - show tables
  - show table status

- Резервное копирование
  - mysqldump
  - lvm snapshot
- Прочее
  - mysqlcheck/mysql
    - analyze (-a) / ANALYZE table
    - optimize (-o) / OPTIMIZE table
    - check / CHECK table
    - repair (-r)
  - myisamchk

- CREATE DATABASE name [charset=CHSET]
- ALTER DATABASE
- CREATE TABLE
- ALTER TABLE
- GRANT *priv* ON *db.tbl* TO *user@host* [IDENTIFIED by '*password*'];
- SHOW GRANTS
- REVOKE *priv* ON *db.tbl* FROM *user@host*
- (CREATE|DROP) USER