

Phpmyadmin — популярный инструмент для управления через браузер базами данных mysql. Я почти всегда устанавливаю его на сервер, если там есть базы mysql. Мне он кажется наиболее простым и удобным средством управления. В статье опишу подробно процесс установки на web сервер apache и nginx.

Если у вас есть желание научиться **профессионально** строить и поддерживать высокодоступные виртуальные и кластерные среды, рекомендую познакомиться с онлайн-курсом **Администратор Linux. Виртуализация и кластеризация**. в OTUS. Курс не для новичков, для поступления нужно пройти .

## Содержание:

- 1 Введение
- 2 Подготовка web сервера apache
- 3 Установка phpmyadmin
- 4 Установка на nginx
- 5 Настройка phpmyadmin
- 6 Заключение
- 7 Видео

## Введение

Ранее я уже касался темы установки phpmyadmin в статье про настройку web сервера на базе nginx. Сейчас хочу рассмотреть более подробно этот вопрос не только на nginx, но и традиционном apache (httpd).

Несколько слов о том, что же из себя представляет инструмент под названием phpmyadmin. Это набор php скриптов, которые устанавливаются на веб сервер. С их помощью можно подключиться к локальному или удаленному mysql серверу и управлять им через веб браузер. Я чаще всего с помощью phpmyadmin делаю следующие операции:

1. Создание новых баз и пользователей, назначение прав.
2. Экспорт или импорт баз не очень большого объема, до 50-ти мегабайт. Если база большего объема, использую утилиты командной строки (mysqldump и mysql) для управления сервером БД.
3. Просмотр содержимого баз данных или информации о них.

По сути, делаю практически все стандартные действия с базами данных. Без phpmyadmin можно обходиться, для самых простых конфигураций серверов (например голый zabbix сервер) мне хватает консольных утилит. Но если есть, к примеру, веб сервер с сайтами, я обязательно поставлю phpmyadmin. Мне так быстрее и удобнее.

Приступим к простой установке phpmyadmin на CentOS 7.

## Подготовка web сервера apache

Далее я буду считать, что у вас уже есть установленный и настроенный сервер CentOS 7. Если это не так, то смотрите мои подробные инструкции с видео на тему установки и базовой настройки сервера CentOS 7.

Если на голом сервере выполнить:

```
# yum install phpmyadmin
```



```

=====
Package                               Arch                               Version                            Repository                          Size
=====
Installing:
phpMyAdmin                             noarch                             4.4.15.7-1.e17                     epel                                  4.7 M
Installing for dependencies:
dejavu-fonts-common                    noarch                             2.33-6.e17                          base                                  64 k
dejavu-sans-fonts                      noarch                             2.33-6.e17                          base                                  1.4 M
fontpackages-filesystem                noarch                             1.44-8.e17                          base                                  9.9 k
gamin                                   x86_64                             0.1.10-16.e17                       base                                  128 k
libX11                                  x86_64                             1.6.3-2.e17                         base                                  605 k
libX11-common                          noarch                             1.6.3-2.e17                         base                                  162 k
libXau                                  x86_64                             1.0.8-2.1.e17                       base                                  29 k
libXpm                                  x86_64                             3.5.11-3.e17                        base                                  54 k
libjpeg-turbo                          x86_64                             1.2.90-5.e17                        base                                  134 k
libpng                                  x86_64                             2:1.5.13-7.e17_2                   updates                              213 k
libtidy                                 x86_64                             0.99.0-31.20091203.e17             epel                                  132 k
libxcb                                  x86_64                             1.11-4.e17                          base                                  189 k
libxslt                                 x86_64                             1.1.28-5.e17                        base                                  242 k
libzip                                  x86_64                             0.10.1-8.e17                        base                                  48 k
lighttpd                               x86_64                             1.4.41-1.e17                        epel                                  350 k
php-bcmath                             x86_64                             5.4.16-36.3.e17_2                 updates                              56 k
php-cli                                 x86_64                             5.4.16-36.3.e17_2                 updates                              2.7 M
php-common                             x86_64                             5.4.16-36.3.e17_2                 updates                              563 k
php-gd                                  x86_64                             5.4.16-36.3.e17_2                 updates                              126 k
php-mbstring                           x86_64                             5.4.16-36.3.e17_2                 updates                              504 k
php-mysql                               x86_64                             5.4.16-36.3.e17_2                 updates                              100 k
php-pdo                                 x86_64                             5.4.16-36.3.e17_2                 updates                              97 k
php-php-gettext                        noarch                             1.0.11-12.e17                      epel                                  22 k
php-process                             x86_64                             5.4.16-36.3.e17_2                 updates                              55 k
php-tcpdf                               noarch                             6.2.11-1.e17                       epel                                  2.1 M
php-tcpdf-dejavu-sans-fonts            noarch                             6.2.11-1.e17                       epel                                  257 k
php-tidy                                 x86_64                             5.4.16-4.e17                       epel                                  22 k
php-xml                                 x86_64                             5.4.16-36.3.e17_2                 updates                              124 k
t1lib                                   x86_64                             5.1.2-14.e17                       base                                  166 k
=====
Transaction Summary
=====
Install 1 Package (+29 Dependent packages)

Total download size: 15 M
Installed size: 64 M
Is this ok [y/d/N]: █
serveradmin.ru

```

Вы увидите полный список зависимостей, но в нем не будет самого веб сервера и интерпретатора php, только его модули. Веб сервер вам нужно предварительно установить и настроить самим. Сделаем это на примере веб сервера httpd (апач). Устанавливаем его:

```
# yum install -y httpd
```

Теперь установим php, без него панель администрирования не заработает:

```
# yum install -y php
```

Запускаем веб сервер, добавляем его в автозагрузку и проверяем работу:

```
# systemctl enable httpd  
# systemctl start httpd
```

Заходите по адресу *http://ip-адрес-сервер*. Вы должны увидеть тестовую страницу апача. Теперь проверим, работает ли php. Создаем тестовую страничку в папке */var/www/html*:

```
# mcedit /var/www/html/index.php
```

```
<?php phpinfo(); ?>
```


Назначаем владельца apache на созданный файл:

```
# chown apache. /var/www/html/index.php
```

Теперь снова заходим по адресу *http://ip-адрес-сервера*, вы должны увидеть вывод `phpinfo`:





**PHP Version 5.4.16**


<b>System</b>	Linux centos7.local 3.10.0-327.28.3.el7.x86_64 #1 SMP Thu Aug 18 19:05:49 UTC 2016 x86_64
<b>Build Date</b>	Aug 11 2016 21:25:54
<b>Server API</b>	Apache 2.0 Handler
<b>Virtual Directory Support</b>	disabled
<b>Configuration File (php.ini) Path</b>	/etc
<b>Loaded Configuration File</b>	/etc/php.ini
<b>Scan this dir for additional .ini files</b>	/etc/php.d
<b>Additional .ini files parsed</b>	/etc/php.d/curl.ini, /etc/php.d/fileinfo.ini, /etc/php.d/json.ini, /etc/php.d/phar.ini, /etc/php.d/zip.ini
<b>PHP API</b>	20100412
<b>PHP Extension</b>	20100525
<b>Zend Extension</b>	220100525
<b>Zend Extension Build</b>	API220100525,NTS
<b>PHP Extension Build</b>	API20100525,NTS
<b>Debug Build</b>	no
<b>Thread Safety</b>	disabled
<b>Zend Signal Handling</b>	disabled
<b>Zend Memory Manager</b>	enabled
<b>Zend Multibyte Support</b>	disabled
<b>IPv6 Support</b>	enabled
<b>DTrace Support</b>	disabled
<b>Registered PHP Streams</b>	https, ftps, compress.zlib, compress.bzip2, php, file, glob, data, http, ftp, phar, zip

Если видите такой же вывод, значит все в порядке, веб сервер готов для работы с phpmyadmin. Приступаем к его установке.

## Установка phpmyadmin

Устанавливаем phpmyadmin со всеми зависимостями:

```
# yum install -y phpmyadmin
```

После установки в каталоге с конфигурациями httpd `/etc/httpd/conf.d` появляется файл **phpMyAdmin.conf**. Сохраним сразу на всякий случай оригинальный файл, перед тем как начать его редактировать:

```
# cp /etc/httpd/conf.d/phpMyAdmin.conf /etc/httpd/conf.d/phpMyAdmin.conf.orig
```

По-умолчанию, в конфиге закрыт доступ к панели управления для всех адресов, кроме 127.0.0.1. Чтобы открыть доступ для всех, приведите файл к следующему виду:

```
# mcedit /etc/httpd/conf.d/phpMyAdmin.conf
```

```
Alias /phpMyAdmin /usr/share/phpMyAdmin
Alias /phpmyadmin /usr/share/phpMyAdmin

<Directory /usr/share/phpMyAdmin/>
  AddDefaultCharset UTF-8
  Require all granted
</Directory>
```

```
<Directory /usr/share/phpMyAdmin/setup/>
  Require all granted
</Directory>

<Directory /usr/share/phpMyAdmin/libraries/>
  Order Deny,Allow
  Deny from All
  Allow from None
</Directory>

<Directory /usr/share/phpMyAdmin/setup/lib/>
  Order Deny,Allow
  Deny from All
  Allow from None
</Directory>

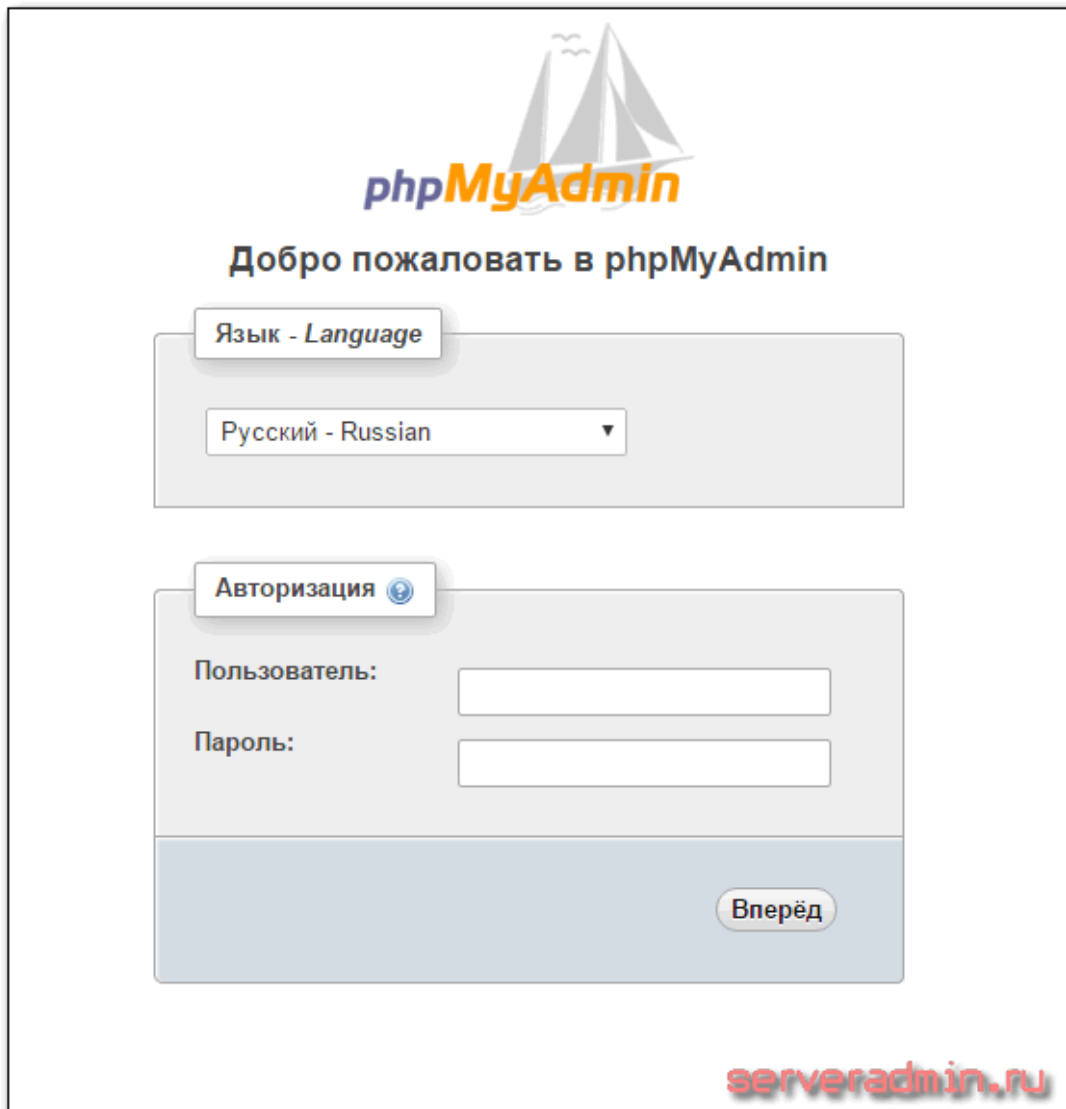
<Directory /usr/share/phpMyAdmin/setup/frames/>
  Order Deny,Allow
  Deny from All
  Allow from None
</Directory>
```


Я удалил все лишнее из файла, в том числе настройки, которые относились к версии apache 2.2. Перезапускаем httpd:

```
# systemctl restart httpd
```

Заходим по адресу <http://ip-адрес-сервера/phpmyadmin>. Вы должны увидеть страницу логина:





  
Добро пожаловать в phpMyAdmin

Язык - *Language*

Русский - Russian ▼

Авторизация ⓘ

Пользователь:

Пароль:

Вперёд

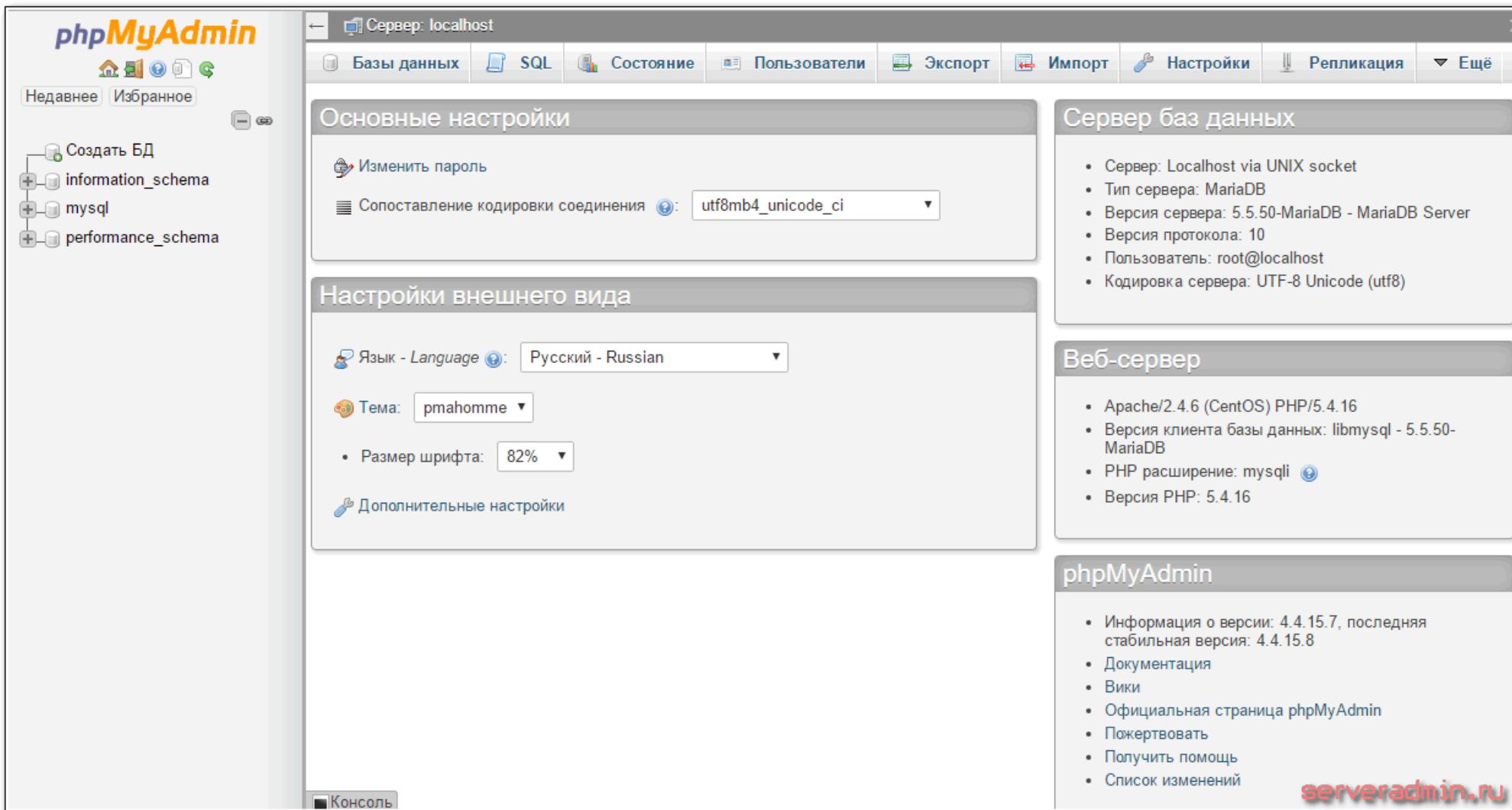
serveradmin.ru

В этот момент я понял, что мне некуда логиниться. На тестовом сервере, где я готовлю статью, нет mysql. Если у вас та же ситуация, что у меня, установите mariadb. Подробно об установке mysql на CentOS 7 я рассказал в статье, посвященной настройке web-сервера, все подробности можете там посмотреть. Сейчас быстро ставим базу данных и запускаем ее:

```
# yum install -y mariadb mariadb-server
# systemctl enable mariadb.service
# systemctl start mariadb
# /usr/bin/mysql_secure_installation
```

После этого можно еще раз открыть web интерфейс phpmyadmin и зайти под учетной записью рута. Откроется главная страница панели с общей информацией о сервере:





The screenshot displays the phpMyAdmin interface for a local server. The top navigation bar includes tabs for 'Базы данных', 'SQL', 'Состояние', 'Пользователи', 'Экспорт', 'Импорт', 'Настройки', 'Репликация', and 'Ещё'. The main content area is divided into several panels:

- Основныe настройки**: Includes 'Изменить пароль' and 'Сопоставление кодировки соединения' set to 'utf8mb4\_unicode\_ci'.
- Настройки внешнего вида**: Includes 'Язык - Language' set to 'Русский - Russian', 'Тема' set to 'pmahomme', and 'Размер шрифта' set to '82%'. A link for 'Дополнительные настройки' is also present.
- Сервер баз данных**: Lists server details: 'Сервер: Localhost via UNIX socket', 'Тип сервера: MariaDB', 'Версия сервера: 5.5.50-MariaDB - MariaDB Server', 'Версия протокола: 10', 'Пользователь: root@localhost', and 'Кодировка сервера: UTF-8 Unicode (utf8)'.
- Веб-сервер**: Lists web server details: 'Apache/2.4.6 (CentOS) PHP/5.4.16', 'Версия клиента базы данных: libmysql - 5.5.50-MariaDB', 'PHP расширение: mysqli', and 'Версия PHP: 5.4.16'.
- phpMyAdmin**: Provides version information (4.4.15.7, 4.4.15.8), documentation, wiki, official site, donation, help, and changelog links.

The left sidebar shows a tree view of databases: 'information\_schema', 'mysql', and 'performance\_schema'. A 'Консоль' tab is visible at the bottom left. The 'serveradmin.ru' logo is in the bottom right corner.



На этом установка закончена, веб панелью можно пользоваться. Некоторые полезные настройки мы рассмотрим далее в соответствующем разделе, а сейчас установим phpmyadmin на веб сервере nginx.

## Установка на nginx

Настроим работу phpmyadmin на веб сервере nginx + php-fpm. Подробно вопрос настройки такого сервера я рассматривал ранее в статье установка и настройка веб сервера nginx + php-fpm на CentOS 7. Там речь шла именно о веб сервере и уклон был в эту сторону. Но phpmyadmin может понадобиться не только на веб серверах. Так что рассмотрим общий случай быстрой и простой настройки phpmyadmin на nginx.

Устанавливаем nginx:

```
# yum install -y nginx
```

Устанавливаем php-fpm:

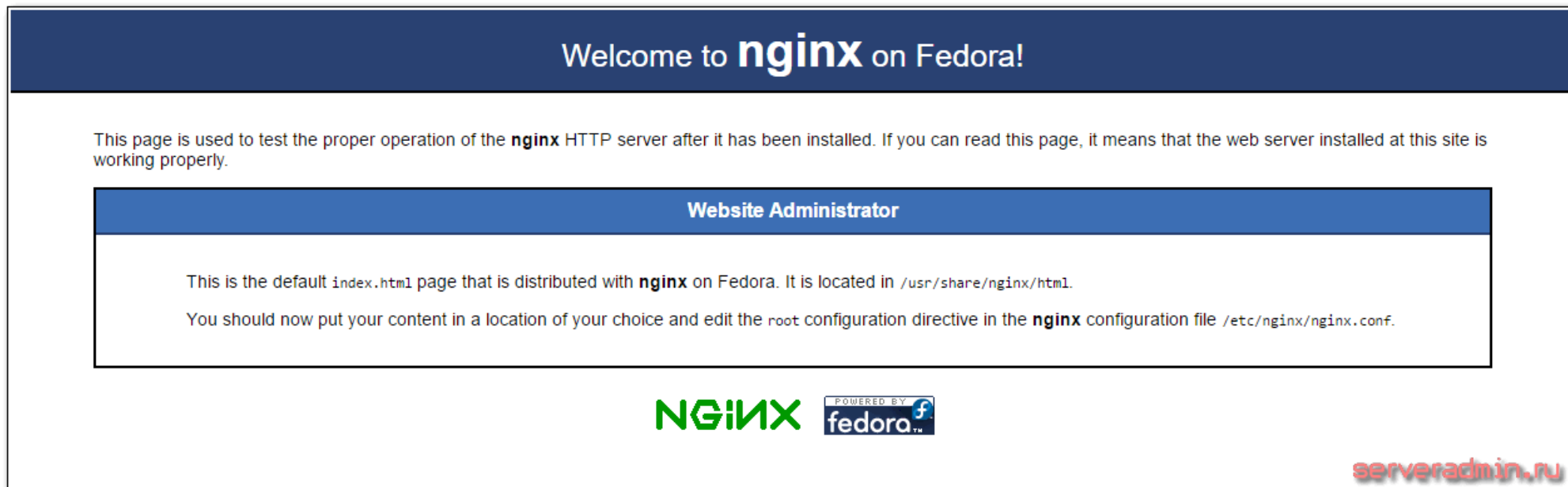
```
# yum install -y php-fpm
```

Запускаем службы и добавляем в автозагрузку:

```
# systemctl start nginx.service  
# systemctl enable nginx.service  
# systemctl start php-fpm.service  
# systemctl enable php-fpm.service
```

Заходим по адресу <http://ip-адрес-сервера> и проверяем. Вы должны увидеть тестовую страницу nginx:





Дальше устанавливаем phpmyadmin на web сервер с nginx:

```
# yum install -y phpmyadmin
```

Редактируем конфигурационный файл nginx для добавления установленной веб панели управления:

```
# mcedit /etc/nginx/nginx.conf
```

Приводим секцию server {} к следующему виду:

```
server {
    listen 80 default_server;
    listen [::]:80 default_server;
    server_name _;
    root /usr/share/nginx/html;
    index index.php index.html index.htm;

    location ~ \.php$ {
        fastcgi_pass 127.0.0.1:9000;
        fastcgi_index index.php;
        include fastcgi_params;
        fastcgi_param SCRIPT_FILENAME $request_filename;
        fastcgi_ignore_client_abort off;
    }

    include /etc/nginx/default.d/*.conf;

    location / {
    }

    error_page 404 /404.html;
    location = /40x.html {
    }

    error_page 500 502 503 504 /50x.html;
    location = /50x.html {
    }
}
```

Я взял стандартные параметры и добавил несколько новых строк для корректной работы. Теперь нам нужно сделать символическую ссылку (сим линк) из папки со скриптами phpmyadmin в корневой каталог веб сервера nginx:

```
# ln -s /usr/share/phpMyAdmin /usr/share/nginx/html/phpmyadmin
```

Перезапускаем nginx:

```
# systemctl restart nginx
```

Дальше по идее все должно работать, но у меня не работало, пришлось перезагрузить весь сервер, так как не работал модуль `php mbstring`, хотя он установлен и подключен. Перезапуск `php-fpm` тоже не помогал, а перезагрузка помогла. Проверил 2 раза, оба раза воспроизвел ошибку. Разбираться не стал в чем причина такого поведения, просто перезагрузите сервер и продолжайте.

Идем по адресу <http://ip-адрес-сервера/phpmyadmin>. Должны увидеть стандартную страницу входа. У меня на ней было предупреждение:

```
Warning in ./libraries/session.inc.php#105  
session_start(): open(/var/lib/php/session/sess_j0r4moac0oo7oh250e6t14rn1kfkl6ta, O_RDWR) failed: No such file or  
directory (2)
```

```
Warning in ./libraries/session.inc.php#105
! session_start():
open(/var/lib/php/session/sess_b0js877plmvu7g34fgf83vmcaao0psui,
O_RDWR) failed: No such file or directory (2)

Backtrace

./libraries/session.inc.php#105: session_start()
./libraries/common.inc.php#350:
require(./libraries/session.inc.php)
./index.php#12: require_once(./libraries/common.inc.php)
```

Это связано с тем, что в папке `/var/lib/php` нет директории `session`. Исправляем это:

```
# cd /var/lib/php/
# mkdir session
```

```
# chown apache:apache session/
```

Обращаю внимание на то, что я назначил владельцем папки с сессиями пользователя apache, хотя у нас веб сервер nginx. Это связано с тем, что php-fpm по-умолчанию работает под пользователем apache. Логичнее изменить это и запускать его от nginx. Для этого надо отредактировать файл `/etc/php-fpm.d/www.conf`, изменив там параметры `user` и `group`. Я не стал этого делать сейчас для простоты. Оставляю данный параметр на ваше усмотрение.

Перезагружаем страницу, ошибка должна исчезнуть. Можно подключиться к mysql. Если сервер бд у вас не установлен, смотрите выше, как быстро его поставить и запустить. На этом установка закончена, рассмотрим несколько полезных настроек phpmyadmin.

## Настройка phpmyadmin

Пхп админ готов к работе сразу после установки, дополнительные настройки не обязательны. Приведу полезный пример при использовании — ограничение доступа к phpmyadmin средствами веб-сервера. Для использования панели необходимо будет не только знать имя учетной записи mysql, но и пользователя и пароль для доступа непосредственно к панели.

Рассмотрим сначала пример с веб сервером **apache**. Будем использовать стандартное средство для ограничения доступа к каталогу с помощью `.htaccess`. Создадим такой файл в папке со скриптами phpmyadmin:

```
# mcedit /usr/share/phpMyAdmin/.htaccess
```

```
AuthName "Enter Password"  
AuthType Basic  
Require valid-user
```

```
AuthUserFile "/usr/share/phpMyAdmin/.htpasswd"
```

Теперь создадим файл с авторизационными данными:

```
# htpasswd -bc /usr/share/phpMyAdmin/.htpasswd user password
```

user имя пользователя

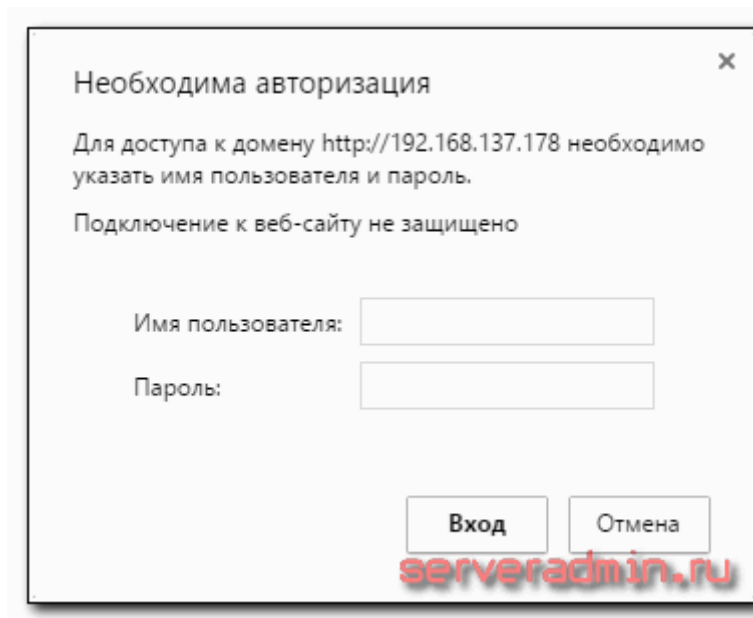
password пароль

Чтобы авторизация заработала, необходимо в файле `/etc/httpd/conf.d/phpMyAdmin.conf` в разделе `Directory` добавить параметр `AllowOverride`, чтобы получилось вот так:

```
<Directory /usr/share/phpMyAdmin/>
  AddDefaultCharset UTF-8
  Require all granted
  AllowOverride All
</Directory>
```

Перезапускаем апач и проверяем настройку. При обращении по адресу веб панели, должно выскочить окошко с авторизацией:





Прделаем то же самое на **nginx**. Так же создаем файл с паролем `.htaccess`:

```
# htpasswd -bc /usr/share/phpMyAdmin/.htpasswd user password
```

Если у вас на сервере не установлен `httpd`, то скорее всего вы увидите ошибку:

```
-bash: htpasswd: command not found
```

Необходимой утилиты нет в системе. Установим htpasswd на centos:

```
# yum install -y httpd-tools
```

Снова создаем файл с паролем:

```
# htpasswd -bc /usr/share/phpMyAdmin/.htpasswd user1 password  
Adding password for user user1
```

Отредактируем конфигурационный файл nginx, добавив в секцию server {} новый location:

```
location /phpmyadmin/ {  
    auth_basic "Enter password";  
    auth_basic_user_file /usr/share/phpMyAdmin/.htpasswd;  
}
```

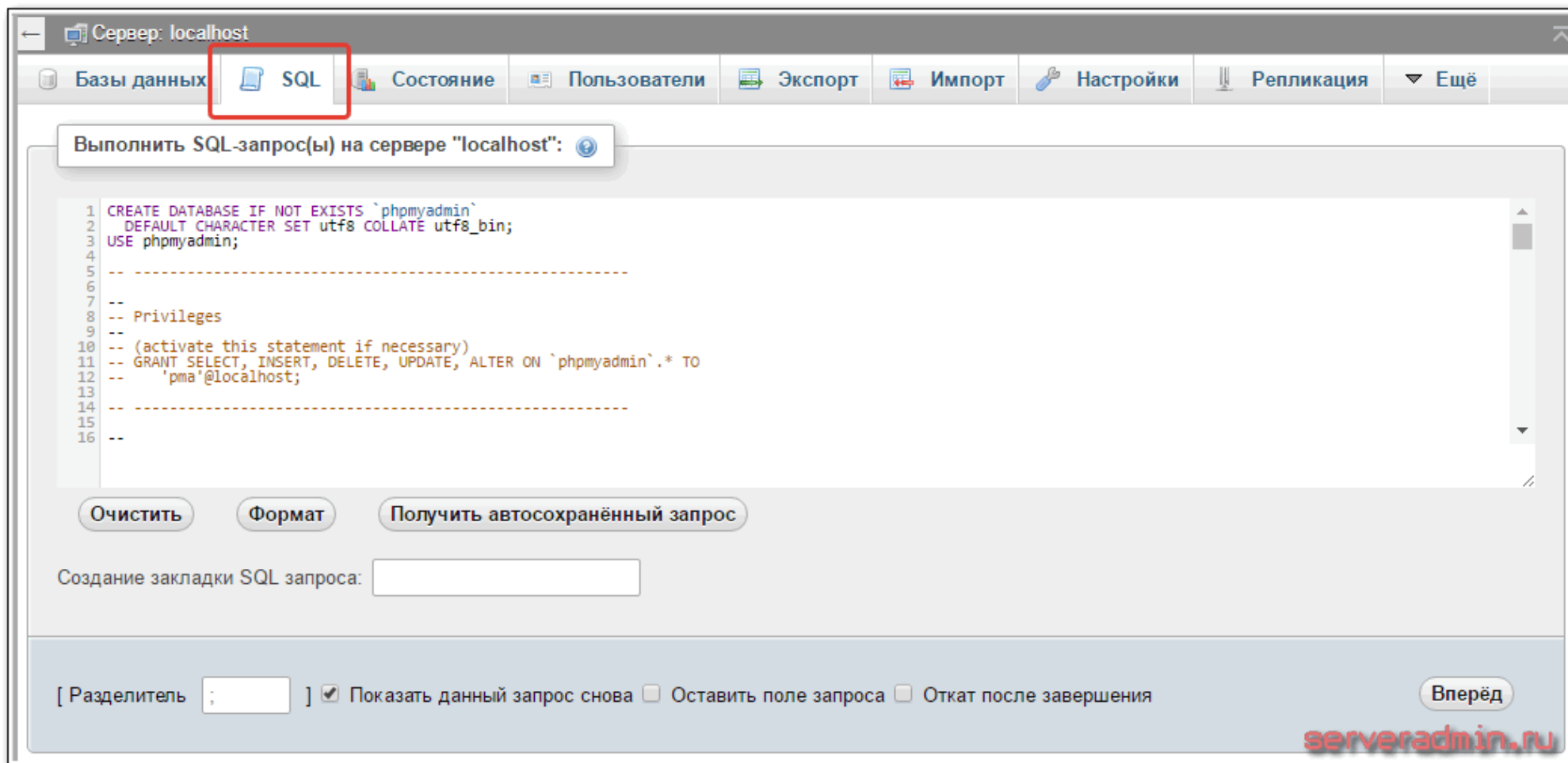
Сохраняете конфиг, перезапускаете nginx и проверяете доступ к странице. Должно выскочить такое же, как и с apache, окно авторизации.

Рассмотрим еще несколько полезных настроек phpmyadmin. После входа в панель, в разделе **Настройки** вы увидите сообщение:

```
Ваши настройки будут сохранены только для текущей сессии. Для постоянного хранения требуется подключение модуля хранения настроек phpMyAdmin.
```

Подключим необходимый модуль для сохранения настроек. Модуль это громко сказано, нам всего лишь необходимо создать отдельную базу данных для хранения настроек и активации некоторых расширенных возможностей. Файл для создания базы данных хранится в директории phpmyadmin по адресу `/usr/share/phpMyAdmin/sql/create_tables.sql`. Копируем содержимое этого файла и выполняем его тут же, в phpmyadmin в разделе SQL:





Теперь можно идти в настройки и изменять их. Я обычно отключаю проверку новой версии. Они выходят достаточно часто, обновлять мне все равно их лень, поэтому информация о новых версиях мне не нужна. Так же отключаю логотип, можно загрузить свой. Сами полистайте настройки, посмотрите, что вам интересно. Все пункты неплохо задокументированы, можно почитать за что отвечают. Иногда бывает полезно вывести отдельный столбец в списке

таблиц с информацией по дате создания и обновления. Это настраивается в разделе **Настройки -> Основная панель -> Структура базы данных.**

Если вы хотите подключиться к удаленному mysql серверу с помощью phpmyadmin, воспользуйтесь скриптом настройки подключения по адресу <http://ip-адрес-сервера/phpmyadmin/setup/>.

## Заключение

Я рассмотрел основные моменты по установке и настройке phpmyadmin на веб сервер apache и nginx. Хочу подчеркнуть, что это мой личный опыт. Возможно, я где-то ошибась или поступаю не совсем верно. Если вы заметите это, прошу написать в комментариях. Я приму к сведению и исправлю статью. Я пишу статьи в том числе и для того, чтобы самому научиться чему-то новому. Комментарии к статьям мне в этом помогают.

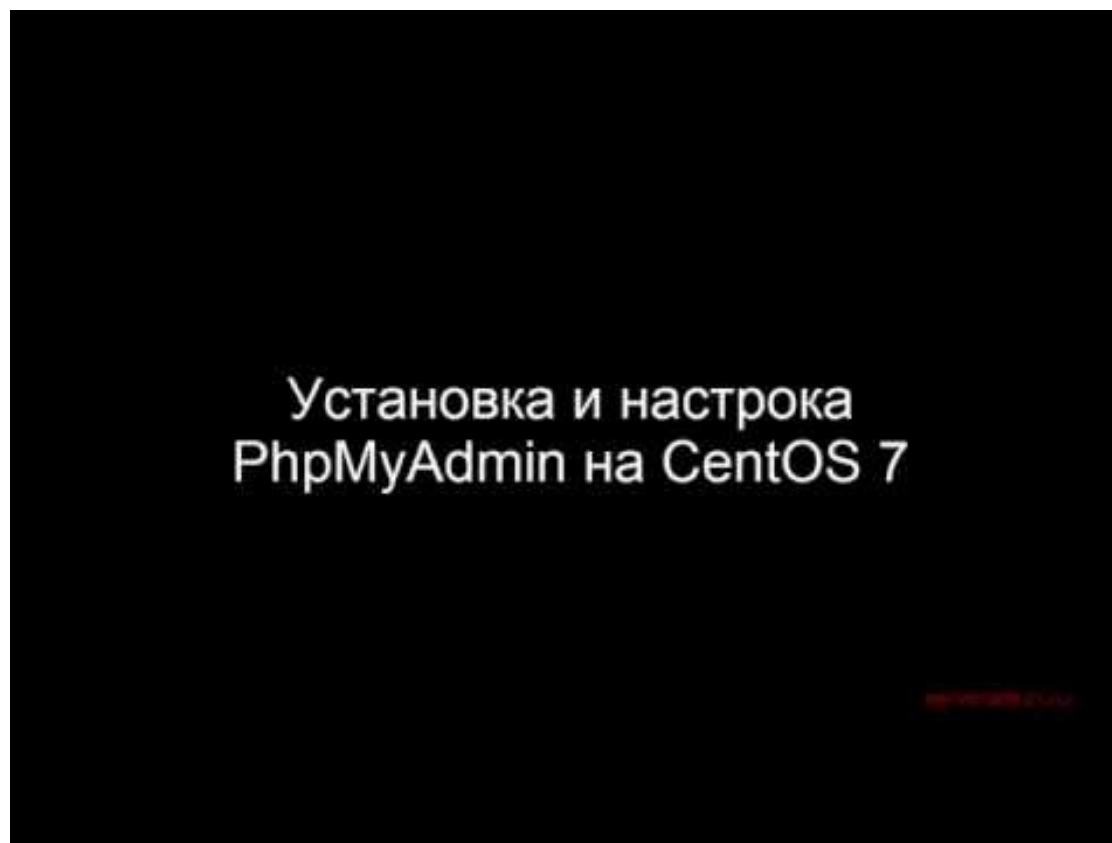
Помогла статья? Подписывайся на telegram канал автора

Анонсы всех статей, плюс много другой полезной и интересной информации, которая не попадает на сайт.

## Видео

Предлагаю посмотреть видео всего процесса установки и настройки phpmyadmin на примере веб сервера apache.





[Watch this video on YouTube](#)

## Онлайн курс по Linux

Если у вас есть желание научиться строить и поддерживать высокодоступные и надежные системы, рекомендую познакомиться с **онлайн-курсом «Администратор Linux»** в OTUS. Курс не для новичков, для поступления нужны базовые знания по сетям и установке Linux на виртуалку. Обучение длится 5 месяцев, после чего успешные выпускники курса смогут пройти собеседования у партнеров. Что



даст вам этот курс:

- Знание архитектуры Linux.
- Освоение современных методов и инструментов анализа и обработки данных.
- Умение подбирать конфигурацию под необходимые задачи, управлять процессами и обеспечивать безопасность системы.
- Владение основными рабочими инструментами системного администратора.
- Понимание особенностей развертывания, настройки и обслуживания сетей, построенных на базе Linux.
- Способность быстро решать возникающие проблемы и обеспечивать стабильную и бесперебойную работу системы.

Проверьте себя на вступительном тесте и смотрите подробнее программу по .