



Вы здесь: > Главная > Simply

14 июля 2022

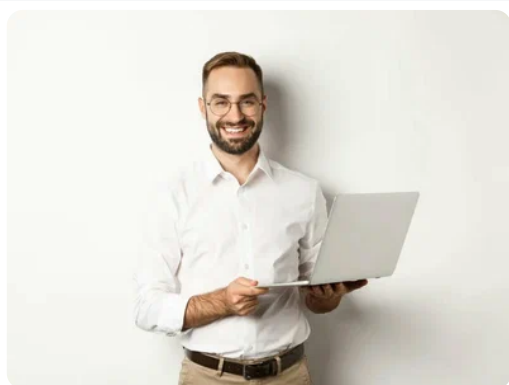
LEMP сервер на базе Alt Linux

LEMP Сервер Просто

ALT LINUX



LEMP-сервер. Часть 2. Сервер на базе Alt Linux – Базальт СПО.



itcom-academy.ru

РЕКЛАМА

Курсы по информационной безопасности и Astra Linux

Курсы по информационной безопасности. Диплом, удостоверение. Гибкий график. Звоните!

[Узнать больше](#)

В предыдущей [статье](#) мы рассмотрели установку сервера из образа Starterkit на базе Alt Linux от Базальт СПО.

Сегодня мы рассмотрим установку LEMP сервера – Linux, Nginx, MariaDB, PHP.

В репозитории Alt Linux уже присутствуют пакеты PHP версий 8.0 и 8.1.

MariaDB (MySQL) версии 10.4, хотя на официальном сайте стабильная уже 10.8.3, это не является большой проблемой для нас, так как судя по номеру версии, никаких критичных фич эта версия не приносит.

Установка пакетов

Обновим дистрибутив:

```
sudo apt-get update
sudo apt-get dist-upgrade
```

Установим все требуемые пакеты:

```
sudo apt-get install nginx curl mc php7 php7 php7-bz2 php7-curl php7-exif php7-mbstring php7-fileinfo
```

Вы можете заменить php7 на php8.0 или php8.1, но так как многие проекты не до конца адаптировали свой код к php версии 8, до октября-ноября 2022 можно еще посидеть на 7.4!

Регистрация и запуск служб

Установленные сервера не настроены на запуск по умолчанию. Нам нужно зарегистрировать службы серверов (включить их) и запустить сами сервера. Откроем консоль и запустим:

```
sudo systemctl enable php7-fpm
sudo systemctl start php7-fpm
```

```
sudo systemctl enable nginx
sudo systemctl start nginx
```

```
sudo systemctl enable mysqld
sudo systemctl start mysqld
```

Настройка Nginx

По умолчанию в Nginx нет активных сайтов, создадим симлинк:

```
sudo ln -s /etc/nginx/sites-available/default.conf /etc/nginx/sites-enabled.d
```

Откроем файл /etc/nginx/sites-available/default.conf

```
sudo mcedit /etc/nginx/sites-available/default.conf
```

И заменим содержимое на:

```
server {
    listen 80 default_server;
    listen [::]:80 default_server;

    root /var/www/html;

    # Add index.php to the list if you are using PHP
    index index.php index.html index.htm index.nginx-debian.html;

    server_name _;

    location / {
        try_files $uri $uri/ =404;
    }

    location ~ \.php$ {
        include fastcgi_params;
        fastcgi_pass unix:/run/php7-fpm/php7-fpm.sock;
        fastcgi_param SCRIPT_FILENAME $document_root/$fastcgi_script_name;
    }
}
```

```
}
```

Создадим папку для веб-сервера:

```
sudo mkdir -p /var/www/html
```

Создадим файл с домашней страницей:

```
echo "<h1>NGINX INSTALLED</h1>" | sudo tee /var/www/html/index.html
```

Перезапустим Nginx

```
sudo systemctl restart nginx
```

Откроем адрес

<http://192.168.1.100>

Где 192.168.1.100 ip адрес настраиваемого сервера.

Откроется страница с надписью, большими буквами:

```
NGINX INSTALLED
```

Проверяем работу PHP

Создадим файл:

```
echo "<?php echo phpinfo(); ?>" | sudo tee /var/www/html/t.php
```

Откроем адрес:

<http://192.168.1.100/t.php>

Должна открыться страница `phpinfo()`

Настройка MariaDB

Запустим:

```
sudo mysql_secure_installation
```

Запустим настройку MariaDB:

```
sudo mysql_secure_installation
```

```
NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!
```

```
In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
you haven't set the root password yet, the password will be blank,
so you should just press enter here.
```

Enter current password for root (enter for none):

Просто нажмите Enter

OK, successfully used password, moving on...

Setting the root password ensures that nobody can log into the MariaDB root user without the proper authorisation.

Enable unix_socket authentication? [Y/n]

Нажмите y

Set root password? [Y/n]

Введите пароль, в рамках моих статей я буду использовать пароль:

Pa\$\$w0rd

Пароль нужно ввести 2 раза!

New password:

Re-enter new password:

Password updated successfully!

Reloading privilege tables..

... Success!

By default, a MariaDB installation has an anonymous user, allowing anyone to log into MariaDB without having to have a user account created for them. This is intended only for testing, and to make the installation go a bit smoother. You should remove them before moving into a production environment.

Remove anonymous users? [Y/n]

Нажмите y

... Success!

Normally, root should only be allowed to connect from 'localhost'. This ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n]

Так как мы настраиваем машину для разработки нажмите n

... skipping.

By default, MariaDB comes with a database named 'test' that anyone can access. This is also intended only for testing, and should be removed before moving into a production environment.

Remove test database and access to it? [Y/n]

Нажмите y

```
- Dropping test database...
... Success!
- Removing privileges on test database...
... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n]
```

Нажмите y

```
... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.

Thanks for using MariaDB!
```

Вот и всё, осталось проверить доступность MariaDB, введем в консоли:

```
mysql -u root --password='Pa$$w0rd'
```

```
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 59
Server version: 10.3.27-MariaDB-0+deb10u1 Debian 10

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

Введите:

```
select user, host from mysql.user;
+-----+-----+
| User      | Host      |
+-----+-----+
| root      | 127.0.0.1 |
| root      | ::1       |
| mariadb.sys | localhost |
| root      | localhost |
| root      | srvdev-1  |
+-----+-----+
5 rows in set (0.002 sec)
```

Разрешаем удаленный доступ к Базе Данных

В консоль MariaDB, которую, мы открыли на предыдущем шаге введем:

```
CREATE USER 'root'@'192.168.1.10' IDENTIFIED BY 'Pa$$w0rd';
```

Где 192.168.1.10 – IP адрес ПК, с которого будет осуществляться удаленный доступ к БД.

Добавим права администратора:

```
GRANT ALL ON *.* to 'root'@'192.168.1.39' IDENTIFIED BY 'Pa$$w0rd' WITH GRANT OPTION;
```

Введем:

```
exit;
```

Теперь нам нужно настроить сервер MariaDB таким образом, чтобы он разрешал удаленный доступ.

Для этого откроем файл /etc/my.cnf.d/server.cnf:

```
sudo mcedit /etc/my.cnf.d/server.cnf
```

Найдем строку:

```
skip-networking
```

и поставим перед ней знак # таким образом закомментировав её:

```
#skip-networking
```

Перезапустим службу MariaDB

```
sudo systemctl restart mysqld
```

Теперь мы сможем удаленно соединиться с базой данных с помощью, например, HeidiSQL.

Проверка работоспособности PHP+MariaDB

Создадим файл:

```
sudo mcedit /var/www/html/m.php
```

С содержимым:

```
<?php
$dbname = "mysql";
$dbuser = "root";
$dbpass = 'Pa$$w0rd';
$dbhost = "127.0.0.1";

$connect = mysqli_connect($dbhost, $dbuser, $dbpass, "mysql") or die("Не могу подключиться к БД '$dbho:
mysqli_select_db($connect,'mysql') or die("Не могу открыть БД '$dbname'");

$test_query = "SHOW TABLES FROM mysql";
$result = mysqli_query($connect,$test_query);

$tblCnt = 0;
```

```
while($tbl = mysqli_fetch_array($result)) {  
    $tblCnt++;  
    echo $tbl[0]."<br />\n";  
}  
  
if (!$tblCnt) {  
    echo "Таблиц не найдено<br />\n";  
} else {  
    echo "Найдено таблиц: $tblCnt<br />\n";  
}
```

Откроем страницу по адресу:

<http://192.168.1.100/m.php>

Результат работы скрипта:

```
column_stats  
columns_priv  
db  
event  
func  
general_log  
global_priv  
gtid_slave_pos  
help_category  
help_keyword  
help_relation  
help_topic  
index_stats  
plugin  
proc  
procs_priv  
proxies_priv  
roles_mapping  
servers  
slow_log  
table_stats  
tables_priv  
time_zone  
time_zone_leap_second  
time_zone_name  
time_zone_transition  
time_zone_transition_type  
user  
Найдено таблиц: 28
```

Настройка PHP и MariaDB проведена корректно!

Заключение

Сегодня мы установили LEMP-сервер на базе серверного образа Starterkit Alt Linux от Базальт СПО.

Обновили систему.

Установили требуемые пакеты.

Настроили автоматический запуск серверных служб.

Настроили Nginx и проверили доступность тестового сайта.

Проверили работу phpinfo().

Настроили MariaDB:

- задали пароль для root;
- проверили подключение к БД из консоли;
- настроили удаленное подключение к серверу;

Написали php-скрипт для тестирования работоспособности PHP+Mysql.

0

Категория: Simply Linux Просмотров: 1762
eltex # Astra_Linux # импортозамещение # информационные_технологии # alt_linux # rosa # рикор

Поиск

Введите текст для поиска...

звук импортозамещение Office scale Программирование рабочий_стол Удаленный_доступ GPU tor Minify неявная_инициализация
ShapeSheet Joomla39 PHP caption

Популярные

- > Установка GNS3 – 2.2.11 в Hyper-V
- > Используем репозитории Яндекса для apt в Debian и Ubuntu. Используем "зеркала" в РФ для установки ПО. Часть 2.
- > Установка поддержки C++ в Visual Studio Code (VSCode)
- > Реальное импортозамещение или что такое ТОРП?
- > Настраииваем Windows для работы с VLAN

Языки

C++
VSCode
GULP
PHP
Vue.js

Qt5

Модели
Виджеты
WebAPI
LDAP

OS

Windows
Linux
Simply Linux
Astra Linux

Сеть

GNS3
Zabbix

Qt6

Подпишитесь

R

VK

Telegram

Discord

Я

Telegram

Контакты

Политика конфиденциальности

Copyright © 2021–2023 АлтунинВВ.Блог

Перепечатка или воспроизведение материалов этого сайта, без разрешения автора, запрещены!

