

Вебсервер Apache2 на ALT Linux

Материал из Lblss.ru

Имеем ОС ALT Linux 4.1.1, обновлённую до бранча.

Содержание

- 1 Установка
- 2 Запуск
- 3 Рабочая директория и юзеры
- 4 Разрешаем имена хостов
- 5 Конфигурирование веб-сервера
- 6 Облегчаем себе жизнь

Установка

Устанавливаем apache2 из пакета:

```
$ sudo apt-get install apache2
Reading Package Lists... Done
Building Dependency Tree... Done
The following extra packages will be installed:
  apache2-common apache2-httpd-worker
The following NEW packages will be installed:
  apache2 apache2-common apache2-httpd-worker
0 upgraded, 3 newly installed, 0 removed and 0 not upgraded.
Need to get 0B/961kB of archives.
After unpacking 2173kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Committing changes...
Preparing... ##### [100%]
1: apache2-common ##### [ 33%]
2: apache2-httpd-worker ##### [ 66%]
3: apache2 ##### [100%]
Running /usr/lib/rpm/posttrans-filetriggers
Done.
$
```

Собственно, вебсервер установлен.

В процессе установки создаётся директория `/etc/httpd2/conf`, где по дефолту располагаются конфиги. Если имеется предыдущая неудачная установка с кривыми конфигами и кучей мусора, то перед установкой директорию необходимо переименовать или удалить:

```
$ sudo rm -rf /etc/httpd2
```

Запуск

Запустить Apache2 можно непосредственно из управления системой или из командной строки:

```
$ sudo /etc/init.d/httpd2 start
Starting httpd2 service: [ DONE ]
```

или так:

```
$ sudo service httpd2 start
Starting httpd2 service: [ DONE ]
```

Теперь можно набрать в строке адреса веб-браузер заветное `http://localhost/` и получить дежурную фразу: **It works!**

Если вызвать команду без параметров (или с неправильным параметром), то в консоли будет выведен список доступных опций, таких как остановка и перезапуск.

Чтоб вебсервер запускался каждый раз при загрузке ОС, можно дать соответствующие указания через центр управления системой (ищите в главном меню) в разделе системные службы. Если доступа к гуям нет, можно из командной строки:

```
$ chkconfig httpd2 on
```

или набросать симлинков на запуск/остановку в /etc/rcX.d в соответствующих runlevel, по полной должно получиться примерно так:

```
$ ls -l /etc/rc[0-6].d/*httpd2
lrwxrwxrwx 1 root root 16 Map 17 07:29 /etc/rc0.d/K15httpd2 -> ../init.d/httpd2
lrwxrwxrwx 1 root root 16 Map 17 07:29 /etc/rc1.d/K15httpd2 -> ../init.d/httpd2
lrwxrwxrwx 1 root root 16 Map 20 23:06 /etc/rc2.d/S85httpd2 -> ../init.d/httpd2
lrwxrwxrwx 1 root root 16 Map 20 23:06 /etc/rc3.d/S85httpd2 -> ../init.d/httpd2
lrwxrwxrwx 1 root root 16 Map 20 23:06 /etc/rc4.d/S85httpd2 -> ../init.d/httpd2
lrwxrwxrwx 1 root root 16 Map 20 23:06 /etc/rc5.d/S85httpd2 -> ../init.d/httpd2
lrwxrwxrwx 1 root root 16 Map 17 07:29 /etc/rc6.d/K15httpd2 -> ../init.d/httpd2
```

Напомню, что дефолтовый запуск ALT Linux с графическим интерфейсом идёт на 5 уровне, остановка на 0 и перезагрузка на 6. Первая буква в симлинках означает соответственно Start и Kill, а число – порядок запуска/остановки.

Рабочая директория и юзеры

В этом разделе предположим, что веб-разработчик имеет имя webdev и входит в группу webdev. Сразу условимся, что это **единственный** пользователь (точней, веб-разработчик), имеющий права на изменение файлов тестового "хостинга" на данном компьютере.

Рабочую директорию создадим в /home. Нет никакого желания пихать её внутрь пользовательской директории, и ещё больше не хочется оставлять в системной.

Определимся со структурой, что где будет жить:

```
/home/www/.log           - тут будут логи веб-сервера по виртхостам
/home/www/.sites         - тут будут конфиги, и, возможно, вспомогательные скрипты
/home/www/.sites/sites-enabled - конфиги виртуальных хостов (серверов)
/home/www/test0.local    - эта и следующие папки соответствуют DocumentRoot виртуальных хостов
/home/www/test1.local
/home/www/test2.local
```

К всем файлам внутри /home/www наш юзер webdev должен иметь доступ на запись. Так как это единственный веб-разработчик на этом компьютере, сделаем его **владельцем** директории /home/www и всего что в ней будет лежать.

Вебсервер (пользователь apache2) должен как минимум, читать директорию /home/www. Для этого введём apache2 в группу webdev, для чего правим файл /etc/group. В итоге должно получиться примерно следующее:

```
$ grep apache2 /etc/group
webdev:x:500:apache2
apache2:x:37:apache2
```

Рисуем папки:

```
$ sudo mkdir -m 0750 /home/www
$ sudo chown webdev:webdev /home/www
$ mkdir -p -m 0750 .log .sites/sites-enabled test0.local test1.local test2.local
```

Разрешаем имена хостов

Вебсервер у нас один. Однако, планируется поддержка нескольких сайтов. Естественно, у каждого должно быть своё имя.

Для разрешения доменных имён сайтов можно поднять какой нибудь DNS-сервер, однако, это выходит за рамки данной статьи. Добавляем по простому, в `/etc/hosts`.

```
$ su
# cat >>/etc/hosts

127.0.0.1 test0.local www.test0.local
127.0.0.1 test1.local www.test1.local
127.0.0.1 test2.local www.test2.local

Ctrl+D
```

Здесь **Ctrl+D** завершает ввод текста. Не забудьте оставить пустую строку или две после. Так как неизвестно, есть ли пустая строка в конце файла, перед вводом своих записей я добавил её на всякий случай.

Не следует давать имена, соответствующие именам сайтов, размещённым в инете, иначе просто на них не попадёте :~)

Конфигурирование веб-сервера

Мы уже условились, что часто востребованные конфиги отдадим нашему webdev. Для начала поправим строчку, описывающую расположение конфигов виртхостов в `httpd.conf`, в самом конце. Существующую строку я закомментировал. Должно получиться так:

```
$ grep sites-enabled /etc/httpd2/conf/httpd2.conf
#Include conf/sites-enabled/*.conf
Include /home/www/.sites/sites-enabled/*.conf
```

Однако, нам нужно сохранить пару дефолтовых конфигов. Сделаем на них симлинки:

```
$ cd /home/www/.sites/sites-enabled/
$ ln -s /etc/httpd2/conf/sites-available/default.conf 00000-default.conf
$ ln -s /etc/httpd2/conf/sites-available/ports_all.conf ports_all.conf
```

Впрочем, из второго нужна только одна строчка `NameVirtualHost *`. А первый описывает дефолтовый виртхост, тот самый, который отдаёт **"It works!"**. Можете вместо симлинков нарисовать свой файл. В любом случае, следует учесть, что имя конфига дефолтового сайта должно быть ранее всех других по алфавиту. Потому оно начинается с нескольких нулей.

Теперь конфиги виртхостов:

```
$ cat >/home/www/.sites/sites-enabled/test0.local.conf
<VirtualHost *>
    ServerAdmin webmaster@webserver.local
    DocumentRoot /home/www/test0.local
    ServerName test0.local
    ServerAlias www.test0.local
    ErrorLog /home/www/.log/test0.local-error_log
    CustomLog /home/www/.log/test0.local-access_log common
    AddDefaultCharset utf-8
</VirtualHost>

Ctrl+D
```

Аналогично делаем и для остальных. Для мегаприверженцев русских кодировок: чтоб не спрашивать про крякозябрики, укажите нужный `DefaultCharset`!

Посмотреть настройки, с которыми был скомпилирован `suexec` можно командой:

```
$ sudo /usr/sbin/suexec2 -V
-D AP_DOC_ROOT="/home"
-D AP_GID_MIN=500
-D AP_HTTPD_USER="apache2"
-D AP_LOG_EXEC="/var/log/httpd2/suexec.log"
-D AP_SAFE_PATH="/usr/local/bin:/usr/bin:/bin"
-D AP_UID_MIN=500
-D AP_USERDIR_SUFFIX="public_html"
```

Тестовый файл для test0.local:

```
$ cat >/home/www/test0.local/index.html
<p>Это тестовый сайт <b>test0.local</b></p>
Ctrl+D
```

Перезапускаем апача:

```
$ sudo /etc/init.d/httpd2 restart
Stopping httpd2 service:
Checking configuration sanity for httpd2: Syntax OK
Starting httpd2 service:
```

```
[ DONE ]
[ DONE ]
[ DONE ]
```

Открываем веб-браузер, набираем в адресной строке `http://test0.local/` и видим на страничке:

Это тестовый сайт **test0.local**

Облегчаем себе жизнь

Рекомендую шаблон конфига виртхоста положить куда то недалеко, например, в папку `/home/www/.sites`, тогда при создании нового виртхоста, достаточно будет его скопировать и указать имя/алиас(ы) сервера и документрут. Не забудьте, что имя файла конфига обязательно должно заканчиваться на **.conf**, иначе Апач его проигнорирует.

Итак, файлы сайтов и конфиги мы отдали нашему пользователю. Теперь он может их править и создавать новые виртхосты. Чтобы наш пользователь `webdev` смог без проблем перезагрузить вебсервер, дадим ему необходимые права на рестарт вебсервера и релоад без ввода пароля, для этого добавим пару строк. Должно получиться примерно следующее:

```
$ sudo grep webdev /etc/sudoers
webdev    ALL=(ALL) ALL
webdev    ALL=(ALL) NOPASSWD: /etc/init.d/httpd2 restart
webdev    ALL=(ALL) NOPASSWD: /etc/init.d/httpd2 reload
```

Осталось бросить на десктоп ярлык с командой

```
sudo /etc/init.d/httpd2 restart
```

или

```
sudo service httpd2 restart
```

Впрочем, более правильным будет вместо `restart` дать команду `reload`

По желанию (по кнопке "Дополнительно") отключить отклик запуска и поставить галку "Выполнять в терминале"

Продолжение: для желающих прикрутить к вебсерверу PHP5

Источник – «[https://wiki.lblss.ru/index.php?](https://wiki.lblss.ru/index.php?title=%D0%92%D0%B5%D0%B1%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D1%80_Apache2_%D0%BD%D0%B0_ALT_Linux&oldid=2471)

[title=%D0%92%D0%B5%D0%B1%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D1%80_Apache2_%D0%BD%D0%B0_ALT_Linux&oldid=2471](https://wiki.lblss.ru/index.php?title=%D0%92%D0%B5%D0%B1%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D1%80_Apache2_%D0%BD%D0%B0_ALT_Linux&oldid=2471)»

Категории: *nix | Apache

Поделиться...