version: '3.6'

services:

vsftpd:

image: fauria/vsftpd

environment:

- FTP\_USER=myuser

- FTP\_PASS=mYp4sS

ports:

- 20:20

- 21:21

volumes:

- /var/app/ftp:/home/vsftpd

- /var/log/ftp:/var/log/vsftpd

curl -T backup-file.tar.gz ftp://username:password@ftp.server.com/backup/

curl -T backup-file.tar.gz ftp://username:password@ftp.server.com/backup/backup.tar.gz

curl --upload-file "{file1,file2}" [http://www.example.com](http://www.example.com/)

curl ftp://myftpsite.com --user myname:mypassword

Использование curl для работы с ftp-серверами

MooSE ( 2007-10-06 )

Есть множество различных программ для доступа к FTP-серверам. Они удобны. Однако большинство из них не пригодны для использования в скриптах. Если же вы пишите скрипт, который должен работать с FTP - вам скорее всего подойдёт curl. Его основные команды очень просты, и их описание можно разместить на одной странице.

Ниже следует что-то вроде небольшой памятки по основным командам curl.

Самый простой способ для обращения к ftp-серверу:

curl ftp://myftpsite.com --user myname:mypassword

Эта команда подключается к серверу, показывает список файлов и отключается.

Для скачивания файла с сервера можно воспользоваться командой:

curl ftp://myftpsite.com/mp3/mozart\_piano\_sonata.zip --user myname:mypassword -o mozart\_piano\_sonata.zip

Для загрузки файла на сервер нужна следующая команда:

curl -T koc\_dance.mp3 ftp://myftpsite.com/mp3/ --user myname:mypassword

Получение списка файлов в поддиректории:

curl ftp://myftpsite.com/mp3/ --user myname:mypassword

Получение списка директорий. В качестве фильтра используется grep:

curl ftp://myftpsite.com --user myname:mypassword -s | grep ^d

Удаление файлов с сервера:

curl ftp://myftpsite.com/ -X 'DELE mp3/koc\_dance.mp3' --user myname:mypassword

Для получений подробной справки используйте команду:

man curl

Или:

curl --manual | less

SFTP

curl -k -u virt03:virt03 -T file4 sftp://192.168.56.110/home/virt03/

Syntax :

curl -k -u username:password -T filename sftp://IP\_Addreess OR Hostname:/path/to/upload

FTP

curl -k -u virt03:virt03 -T file6 ftp://192.168.56.110

Example

[root@virt03 test]# curl -k -u virt03:virt03 -T file4 sftp://192.168.56.110/home/virt03/

**Внесение SSL в конфигурации Vsftpd**

Откройте конфигурационный файл с привилегиями root:

sudo nano /etc/vsftpd.conf

В нижней части файла найдите строку, соответствующую только что созданному сертификату SSL:

rsa\_cert\_file=/etc/ssl/private/vsftpd.pem

Под этой строкой нужно внести дополнительную информацию об SSL.

При создании сертификата ключевой файл и сертификат были помещены в один файл, поэтому можно указать строку закрытого ключа:

rsa\_private\_key\_file=/etc/ssl/private/vsftpd.pem

Затем нужно внести следующие строки, которые ограничат доступ клиентов к TLS.

ssl\_enable=YES  
allow\_anon\_ssl=NO  
force\_local\_data\_ssl=YES  
force\_local\_logins\_ssl=YES

Затем нужно настроить сервер на использование TLS (преемник SSL):

ssl\_tlsv1=YES  
ssl\_sslv2=NO  
ssl\_sslv3=NO

Чтобы расширить конфигурационный файл, необходимо внести дополнительные опции:

require\_ssl\_reuse=NO  
ssl\_ciphers=HIGH

Сохранив изменения, закройте файл.

Затем перезапустите сервер, чтоб активировать внесенные изменения:

sudo service vsftpd restart

#### **Как настроить SFTP VSFTPD с SSL / TLS на Ubuntu 18.04**

### **Настройка SSL для VSFTPD**

Поскольку данные (даже учетные данные), переданные по FTP, не зашифрованы, мы можем включить TLS / SSL для обеспечения другого уровня безопасности для нашего FTP-сервера.

Чтобы создать сертификат с помощью openssl, выполните следующие действия:

[?](https://itsecforu.ru/2018/05/30/как-настроить-sftp-vsftpd-с-ssl-tls-на-ubuntu-18-04/)

|  |
| --- |
| # sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/ssl/private/vsftpd.pem -out /etc/ssl/private/vsftpd.pem |

где 365 – для добавления сертификата в течение одного года и добавления того же значения для флажков -out, -keyout, чтобы найти закрытый ключ и сертификат в том же файле.

Вам будет предложено добавить всю необходимую информацию для создания сертификата, как показано ниже

[?](https://itsecforu.ru/2018/05/30/как-настроить-sftp-vsftpd-с-ssl-tls-на-ubuntu-18-04/)

|  |
| --- |
| Generating a 2048 bit RSA private key  .................+++  ..............................................................................................+++  writing new private key to '/etc/ssl/private/vsftpd.pem'  -----  You are about to be asked to enter information that will be incorporated  into your certificate request.  What you are about to enter is what is called a Distinguished Name or a DN.  There are quite a few fields but you can leave some blank  For some fields there will be a default value,  If you enter '.', the field will be left blank.  -----  Country Name (2 letter code) [AU]:AU  State or Province Name (full name) [Some-State]:SY  Locality Name (eg, city) []:Sydney  Organization Name (eg, company) [Internet Widgits Pty Ltd]:Linoxide  Organizational Unit Name (eg, section) []:  Common Name (e.g. server FQDN or YOUR name) []:ubuntu  Email Address []:test@email.com |

После создания сертификата нам нужно добавить его в файл конфигурации vsftpd и включить SSL.

Для этого добавьте следующую строку в указанный файл:

[?](https://itsecforu.ru/2018/05/30/как-настроить-sftp-vsftpd-с-ssl-tls-на-ubuntu-18-04/)

|  |
| --- |
| rsa\_cert\_file=/etc/ssl/private/vsftpd.pem  rsa\_private\_key\_file=/etc/ssl/private/vsftpd.pem  ssl\_enable=YES |

осле того, как нам нужно отклонить анонимные подключения через SSL и потребовать SSL для передачи данных и входа в систему.

Для этого добавьте следующие строки в файл /etc/vsftpd.conf:

[?](https://itsecforu.ru/2018/05/30/как-настроить-sftp-vsftpd-с-ssl-tls-на-ubuntu-18-04/)

|  |
| --- |
| allow\_anon\_ssl=NO  force\_local\_data\_ssl=YES  force\_local\_logins\_ssl=YES |

Как только добавленные строки добавлены, мы настроим сервер для использования TLS, который является предпочтительным преемником SSL.

Для этого добавьте следующие строки в один и тот же файл:

[?](https://itsecforu.ru/2018/05/30/как-настроить-sftp-vsftpd-с-ssl-tls-на-ubuntu-18-04/)

|  |
| --- |
| ssl\_tlsv1=YES  ssl\_sslv2=NO  ssl\_sslv3=NO |

После всех шагов наш файл конфигурации vsftpd будет выглядеть так:

[?](https://itsecforu.ru/2018/05/30/как-настроить-sftp-vsftpd-с-ssl-tls-на-ubuntu-18-04/)

|  |
| --- |
| listen=NO  listen\_ipv6=YES  anonymous\_enable=NO  local\_enable=YES  write\_enable=YES  local\_umask=022  dirmessage\_enable=YES  use\_localtime=YES  xferlog\_enable=YES  connect\_from\_port\_20=YES  chroot\_local\_user=YES  secure\_chroot\_dir=/var/run/vsftpd/empty  pam\_service\_name=vsftpd  pasv\_enable=Yes  pasv\_min\_port=10000  pasv\_max\_port=11000  user\_sub\_token=$USER  local\_root=/home/$USER/ftp  userlist\_enable=YES  userlist\_file=/etc/vsftpd.userlist  userlist\_deny=NO  rsa\_cert\_file=/etc/ssl/private/vsftpd.pem  rsa\_private\_key\_file=/etc/ssl/private/vsftpd.pem  ssl\_enable=YES  allow\_anon\_ssl=NO  force\_local\_data\_ssl=YES  force\_local\_logins\_ssl=YES  ssl\_tlsv1=YES  ssl\_sslv2=NO  ssl\_sslv3=NO |

После любых изменений в файле конфигурации vsftpd нам необходимо перезапустить службу следующей командой:

[?](https://itsecforu.ru/2018/05/30/как-настроить-sftp-vsftpd-с-ssl-tls-на-ubuntu-18-04/)

|  |
| --- |
| # sudo service vsftpd restart |

Вы также можете настроить vsftpd для использования letencrypt сертификата для sftp, но убедитесь, что у вас есть домен для проверки.

Вы можете добавить переменную пути в конфигурацию vsftpd, где хранится ключ ssl и private.

rsa\_cert\_file=/ssl/letsencrypt/ftpdomain.com/chain-bundle.pem  
rsa\_private\_key\_file=/ssl/letsencrypt/ftpdomain.com/private-key.pem

VSFTPD configuration Primer:

**Если сервер предназначен для работы в интернет, полезно добавить некоторые ограничения.**  
  
 **anonymous\_enable=YES**  
 **no\_anon\_password=YES**  
 **anon\_world\_readable\_only=YES**  
 **connect\_from\_port\_20=YES**  
 **no\_anon\_password=YES**  
 **hide\_ids=YES**  
 **pasv\_min\_port=50000**  
 **pasv\_max\_port=60000**  
 **xferlog\_enable=YES**  
 **ascii\_download\_enable=NO**  
 **async\_abor\_enable=YES**  
 **one\_process\_model=YES**  
 **idle\_session\_timeout=120**  
 **data\_connection\_timeout=300**  
 **accept\_timeout=60**  
 **connect\_timeout=60**  
 **anon\_max\_rate=15000**  
  
**Для работы в интернет в независимом от inetd/xinetd режиме**  
  
 **listen=YES**  
 **max\_clients=200**  
 **max\_per\_ip=4**  
 **anonymous\_enable=YES**  
 **no\_anon\_password=YES**  
 **anon\_world\_readable\_only=YES**  
 **connect\_from\_port\_20=YES**  
 **hide\_ids=YES**  
 **pasv\_min\_port=50000**  
 **pasv\_max\_port=60000**  
 **xferlog\_enable=YES**  
 **async\_abor\_enable=YES**  
 **one\_process\_model=YES**  
 **idle\_session\_timeout=120**  
 **data\_connection\_timeout=300**  
 **accept\_timeout=60**  
 **connect\_timeout=60**  
 **anon\_max\_rate=15000**

## Настраиваем FTP-сервер с виртуальными пользователями на базе vsftpd

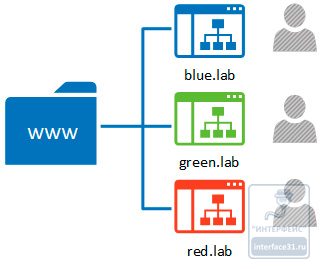
Протокол FTP, несмотря на преклонный возраст и серьезные недостатки продолжает широко применяться, во многом благодаря простоте и поддержке со стороны самого широкого спектра устройств и ПО. FTP-сервер является неотъемлемой частью любого веб-сервера, применяется в системах автоматизированного обмена данными. Наиболее часто встающей перед администраторами таких систем задачей является создание сервера с виртуальными пользователями. Сегодня мы расскажем, как это сделать.

Про настройку FTP-сервера на базе **vsftpd** мы уже рассказывали на страницах нашего сайта, недавно существенно дополнив и обновив статью [Простой FTP-сервер на базе Ubuntu (vsftpd)](https://interface31.ru/tech_it/2012/02/prostoy-ftp-server-na-baze-ubuntu.html). В ней мы рассказали каким образом можно ограничить список пользователей, имеющих возможность подключаться к серверу, и их права. Однако локальные пользователи остаются при этом локальными пользователями и несмотря на ограниченные права могут оказаться объектом атаки, направленной на повышение прав, например, через уязвимости в каком-либо ПО.

Кроме того, ручное редактирование списков и настроек пользователей может оказаться весьма трудоемким делом если их не один-два, а существенно больше. Также иногда возникает ситуация, что светить данные реальных пользователей просто нежелательно, например, FTP используется движком сайта (CMS), связка пользователь-пароль явно прописана в конфиге и теоретически доступна любому работающему с сайтом сотруднику.

В таких ситуациях следует использовать виртуального пользователя, который будет иметь доступ исключительно к данному FTP-ресурсу и учетные данные которого можно смело сообщить всем принимающих участие в работе реальным сотрудникам.

В нашей статье мы настроим FTP-сервер для реализации следующей схемы:



Пусть в нашем распоряжении имеется некий веб-сервер под управлением которого находятся три сайта: **blue.lab**, **green.lab** и **red.lab**, физически расположенные в одноименных директориях. Наша задача создать FTP-сервер с виртуальными пользователями, каждый из которых имеет доступ только к своему сайту.

Важное замечание, в данной схеме все сайты должны работать от имени веб-сервера (www-data), если вы используете запуск каждого сайта от имени своего (реального) пользователя, то вам лучше настроить FTP-сервер для работы с реальными пользователями или дополнительно обеспечить нужные права доступа для виртуальных пользователей, что может представлять из себя нетривиальную задачу.

Прежде всего установим соглашение: имя виртуального пользователя должно совпадать и именем корневой FTP-папки, в нашем случае папки сайта, а все такие папки должны располагаться в одном месте, т.е. подпадать под шаблон **/var/www/$USER**. Это позволит в дальнейшем с легкостью манипулировать сайтами и виртуальными пользователями, а также избежать путаницы в именах и настройках.

Определившись с моделью размещения данных можно переходить к настройкам ПО. Кроме **vsftpd** нам понадобятся пакеты **libpam-pwdfile** и **apache2-utils**. Последний входит в состав одноименного веб-сервера и если у вас уже работает Apache, то устанавливать его не надо.

Сама установка предельно проста:

apt-get install vsftpd libpam-pwdfile apache2-utils

После чего перейдем к настройкам FTP-сервера, которые расположены в файле **/etc/vsftpd.conf**, мы перечислим опции в порядке их нахождения в файле.

Данная опция запускает FTP-сервер как службу

listen=YES

ниже имеется аналогичная взаимоисключающая опция:

listen\_ipv6=NO

Первая из них обеспечивает поддержку IPv4, вторая сразу IPv6 и IPv4, поэтому должна быть включена только одна из них.

Запрещаем анонимных пользователей:

anonymous\_enable=NO

И разрешаем локальных (и виртуальных) пользователей:

local\_enable=YES

Также разрешаем запись:

write\_enable=YES

и задаем маску для вновь создаваемых файлов и папок:

local\_umask=022

Это обеспечит установку прав 755 на папки и 644 на файлы, т.е. полный доступ только владельцу и чтение для группы и остальных.

Чтобы FTP использовал локальное время сервера, а не GMT установим:

use\_localtime=YES

И включим лог загружаемых и скачиваемых файлов:

xferlog\_enable=YES

Разрешаем передачу данных через порт 20, требуется для лучшей совместимости с некоторыми клиентами:

connect\_from\_port\_20=YES

Задаем путь к логу и его формат:

xferlog\_file=/var/log/vsftpd.log

xferlog\_std\_format=YES

Укажем таймауты:

idle\_session\_timeout=600

data\_connection\_timeout=120

Первый задает время отключения бездействующего пользователя, а второй время ожидания возобновления неоконченной передачи.

Для корректной работы с текстовыми данными, в частности с символами переноса строки в разных ОС (CR+LF в Windows, LF в Linux) включим поддержку ASCII, это полезно, если вы скачиваете скрипты или файлы с Linux сервера, правите их в среде Windows и заливаете обратно. В этом случае FTP автоматически будет менять символы переноса строки для соответствия стандартам системы.

ascii\_upload\_enable=YES

ascii\_download\_enable=YES

Обязательно изолируем пользователя в корне его домашней директории и рядом впишем опцию, разрешающую запись в корень:

chroot\_local\_user=YES

allow\_writeable\_chroot=YES

Остальные опции оставляем без изменения.

Ниже добавим необходимые нам настройки. Разрешим пассивный режим и явно укажем диапазон портов:

pasv\_enable=YES

pasv\_min\_port=62000

pasv\_max\_port=62999

Включим виртуальных пользователей:

guest\_enable=YES

Укажем от имени какого реального пользователя они будут работать, так как в нашем случае нам надо работать с содержимым веб-сервера, то указываем соответствующего пользователя **www-data**.

guest\_username=www-data

Применяем к виртуальным пользователям настройки локальных, иначе к ним будут применены ограничения анонимных.

virtual\_use\_local\_privs=YES

Укажем шаблон для автоматической генерации домашнего каталога виртуального пользователя:

user\_sub\_token=$USER

Теперь можно задать сами домашние директории

local\_root=/var/www/$USER

Также иногда неплохо будет скрыть реальных владельцев содержимого, для этого добавьте:

hide\_ids=YES

Если вы используете Ubuntu, то добавьте следующую недокументированную опцию:

seccomp\_sandbox=NO

Это позволит избежать ошибки **500 OOPS: prctl PR\_SET\_SECCOMP failed**, которая возникает из некорректной совместной работы **vsftpd** и системы безопасности **seccomp**, в Debian данная ошибка не проявляется.

На этом настройка FTP-сервера закончена, сохраняем файл и переходим к настройке правил аутентификации. Откроем **/etc/pam.d/vsftpd**, закомментируем в нем все строки и добавим две свои:

auth required pam\_pwdfile.so pwdfile /etc/vsftpwd

account required pam\_permit.so

После этого аутентификация локальных пользователей на FTP-сервере будет невозможна, работать будут только виртуальные. Опция **pwdfile /etc/vsftpwd** указывает путь к файлу паролей, его нам необходимо будет создать на следующем шаге.

Для создания пользователей будем использовать утилиту **htpasswd** из состава **apache2-utils**, для создания файла паролей и первого пользователя введите:

htpasswd -c -d /etc/vsftpwd blue.lab

Ключ **-c** предписывает создать файл паролей, если он существует, то все данные в нем будут очищены, а файл перезаписан, **-d** задает необходимый для vsftpd алгоритм шифрования хэша пароля.

Следующие пользователи создаются командой (в нашем примере это пользователь green.lab):

htpasswd -d /etc/vsftpwd green.lab

Для удаления пользователя используйте (удаляем red.lab):

htpasswd -D /etc/vsftpwd red.lab

После чего можно перезапустить службу FTP-сервера и проверить его работу:

service vsftpd restart

Как видим, настроить FTP-сервер с виртуальными пользователями по нашей инструкции (в отличие от многих иных "инструкций" в сети) абсолютно несложно и недолго. Если вы будете внимательно следовать всем нашим рекомендациям, то сложностей у вас возникнуть не должно.