

Меня попросили в разделе с заказами написать статью на тему ftp сервера. Я планировал ее написать ранее, но все откладывал, а теперь решил сделать это. В качестве примера я возьму ftp сервер - vsftpd, систему - CentOS 7, 8. Рассмотрю несколько разных вариантов настройки - простой и более сложный.

Если у вас есть желание научиться строить и поддерживать высокодоступные и надежные системы, рекомендую познакомиться с онлайн-курсом "**Administrator Linux. Professional**" в OTUS. Курс не для новичков, для поступления нужно пройти .

Содержание:

- 1 Введение
- 2 Простая настройка ftp в CentOS 7
- 3 Настройка vsftpd с виртуальными пользователями
- 4 Хранение vsftpd пользователей в mysql
- 5 Настройка vsftpd в Centos 8
- 6 Настройка SSL / TLS
- 7 Заключение
- 8 Видео

## Введение

По моим наблюдениям, популярность ftp серверов с каждым годом падает. Сам я очень редко им пользуюсь и на свои сервера чаще всего не устанавливаю. Так как я работаю с серверами, к которым имею доступ по ssh, необходимости в ftp нет, использую SFTP.

Серьезную конкуренцию ftp создали облачные сервисы для хранения файлов. Чтобы кому-то передать файл, достаточно его расшарить на облачном хранилище и отправить человеку ссылку.


Тем не менее, в некоторых случаях может понадобиться ftp сервер. Доступ к такому серверу часто дают тогда, когда нужно, чтобы к вам кто-то что-то закачал большого объема. Так же ftp доступ дают некоторые хостинг компании, когда вы у них заказываете сервер для бэкапов. Например, iHor, которым я раньше пользовался, предлагает услугу пространство для бэкапа именно по протоколу ftp.

Так что хоть ftp и потерял популярность, но не перестал быть актуальным. Рассмотрим настройку ftp сервера на примере **vsftpd**. Сделаем несколько конфигураций:

1. Самая быстрая и простая. В качестве пользователей ftp будут системные пользователи.
2. Более сложная конфигурация, где пользователи будут виртуальные, хранить их будем в отдельном файле.
3. Вариант с виртуальными пользователями, но хранить мы их будем в mysql базе.

Работать будем на следующем сервере:

```
[root@zabbix ~]# uname -a
Linux zabbix.office.local 3.10.0-327.10.1.el7.x86_64 #1 SMP Tue Feb 16 17:03:50 UTC 2016 x86_64 x86_64 x86_64 GNU/Linux
[root@zabbix ~]# cat /etc/redhat-release
CentOS Linux release 7.2.1511 (Core)
[root@zabbix ~]#
```



Настройка vsftpd для Centos 8 будет немного отличаться, но в целом, все то же самое. Я не буду дублировать информацию, просто прочитайте отдельный раздел в самом конце, где указаны замечания именно для 8-й версии.

## Простая настройка ftp в CentOS 7

Если у вас уже есть готовый сервер, то сразу приступайте к установке vsftpd. Если нет, то рекомендую две подходящие статьи для предварительной подготовки: подробная установка centos 7 и базовая настройка сервера.

Начинаем традиционно с обновления системы:

```
# yum update
```

Устанавливаем **vsftpd**:

```
# yum install vsftpd
```

Переходим к настройке. Я использую следующую схему работы ftp сервера с системными пользователями. Пользователю root разрешаю ходить по всему серверу. Всем остальным пользователям только в свои домашние директории. Анонимных пользователей отключаю.

Очистим каталог `/etc/vsftpd`, нам ничего не нужно из того, что там есть по-умолчанию. Можете сохранить куда-нибудь оригинальный конфиг, нам он не понадобится.

```
# rm -rf /etc/vsftpd/*
```

Открываем конфиг сервера `/etc/vsftpd/vsftpd.conf`

```
# mcedit /etc/vsftpd/vsftpd.conf
```

и приводим его к следующему виду `vsftpd.conf`. Не вывожу файл сюда, чтобы не растягивать страницу, конфигов в статье будет много. В файле даны подробные комментарии к каждому параметру.

Создаем необходимых пользователей, файлы и каталоги для установленной конфигурации. Добавим тестового пользователя ftp в систему:

```
# useradd -s /sbin/nologin ftp-user  
# passwd ftp-user
```

Пользователя создаем без оболочки. Тут сразу можно указать в качестве домашней директории необходимый каталог, в котором будет работать пользователь. Я специально этого не делаю, чтобы продемонстрировать работу пользовательских настроек в отдельном файле. Пользователь будет создан со стандартным домашним каталогом в `/home`, но при работе по ftp он будет направлен в другой каталог, который мы ему укажем через файл

пользовательских настроек vsftpd.

Здесь стоит обратить внимание на один момент. Начиная с какой-то версии то ли vsftpd, то ли это касается самой системы Centos, но пользователь с оболочкой /sbin/nologin не может подключаться по ftp. Связано это с тем, что идет проверка оболочки, а ее нет в файле */etc/shells*. Я не пробовал ее туда добавлять, так как не понимаю до конца назначение этого файла. Я предлагаю просто отключить проверку оболочки в настройках pam для vsftpd. Делается это в файле */etc/pam.d/vsftpd*. Нужно закомментировать следующую строку:

```
#auth required pam_shells.so
```

Продолжаем настройку ftp сервера. Создаем каталог для персональных настроек пользователей:

```
# mkdir /etc/vsftpd/users
```

В каталоге можно будет создать отдельно файлы с именами пользователей и передать им персональные параметры. Для примера создадим файл с именем пользователя ftp-user и укажем его домашний каталог:

```
# touch /etc/vsftpd/users/ftp-user  
# echo 'local_root=/ftp/ftp-user/' >> /etc/vsftpd/users/ftp-user
```

Не забываем создать этот каталог и назначить ему владельца:

```
# mkdir /ftp && chmod 0777 /ftp  
# mkdir /ftp/ftp-user && chown ftp-user. /ftp/ftp-user/
```

Создаем файл с пользователями, которым можно выходить за пределы домашнего каталога:

```
# touch /etc/vsftpd/chroot_list
```

Добавляем туда рута:

```
# echo 'root' >> /etc/vsftpd/chroot_list
```

Создаем файл со списком пользователей ftp, которым разрешен доступ к серверу:

```
# touch /etc/vsftpd/user_list  
# echo 'root' >> /etc/vsftpd/user_list && echo 'ftp-user' >> /etc/vsftpd/user_list
```

Этим списком мы можем ограничить доступ к ftp серверу системных пользователей, которым туда ходить не нужно. Осталось только создать файл для логов:

```
# touch /var/log/vsftpd.log && chmod 600 /var/log/vsftpd.log
```

Все готово для работы. Добавляем vsftpd в автозагрузку и запускаем:

```
# systemctl enable vsftpd  
# systemctl start vsftpd
```

Проверяем, запустился ли он:

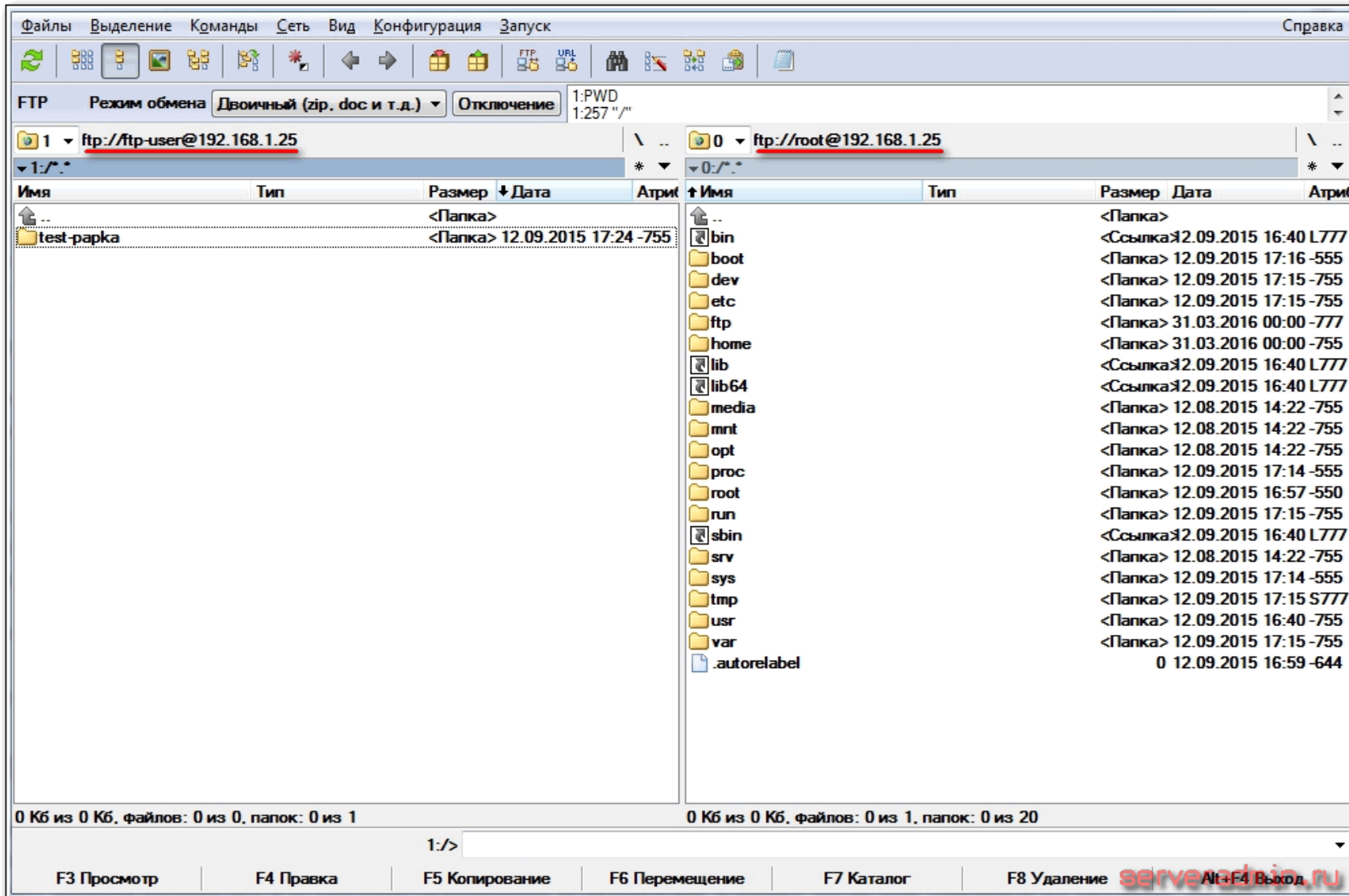
```
# netstat -tulnp | grep vsftpd
```

```
tcp 0 0 0.0.0.0:21 0.0.0.0:* LISTEN 19505/vsftpd
```

Все в порядке, можно подключаться клиентом.

Прежде чем начнете проверять подключение, убедитесь, что у вас настроен firewall. Либо выключите его, либо настройте для разрешения подключений по ftp. Узнать как правильно настроить iptables для работы ftp сервера можно в отдельной статье, посвященной настройке iptables. Там есть все необходимые пояснения и примеры.

Я зашел пользователем root и смог прогуляться по всей файловой системе.



Пользователь ftp-user ограничен своим домашним каталогом, указанным в индивидуальном файле с параметрами. Выйти за его пределы он не может. В лог файле отражены все действия пользователей на сервере: загрузка и скачивание файлов.

Обращаю внимание, что это мой опыт работы с vsftpd. Давать доступ системному руту по ftp ко всей файловой системе весьма опасно, нужно пользоваться осторожно. Я уже точно не помню, зачем я так делал, остались заметки с давних времен. Рассматривайте это просто как демонстрацию возможностей и используйте эти функции так, как вам нужно.

## Настройка vsftpd с виртуальными пользователями

Рассмотрим вариант, когда пользователи ftp сервера не должны пересекаться с локальными. В данном примере будут работать только виртуальные пользователи. Я мельком проверил, можно ли настроить и тех и других, оказалось, что можно. Но там надо аккуратно с правами разбираться и со списками разрешенных пользователей. Я решил, что не буду описывать эту ситуацию, так как не очень представлю, когда она может пригодиться. Если кому-то надо, то на базе этой статьи он сам сможет разобраться.

Чтобы авторизовать виртуальных пользователей, установим дополнительный пакет **compat-db**:

```
# yum install compat-db
```

На всякий случай сохраните оригинальный pam.d файл, если захотите снова вернуться к системным пользователям:

```
# cp /etc/pam.d/vsftpd /etc/pam.d/vsftpd.orig
```

Нужно изменить pam файл */etc/pam.d/vsftpd*, приведя его к следующему виду:

```
# mcedit /etc/pam.d/vsftpd
```



```
auth required pam_userdb.so db=/etc/vsftpd/virt_users
account required pam_userdb.so db=/etc/vsftpd/virt_users
session required pam_loginuid.so
```

```
# mcedit /etc/vsftpd/vsftpd.conf
```

Рисуем следующий конфиг для vsftpd `vsftpd.conf` Создаем файл с виртуальными пользователями:

```
# touch /etc/vsftpd/virt_users
```

Добавляем туда в первую строку имя пользователя, во вторую его пароль. В конце не забудьте перейти на новую строку, это важно. Файл должен быть примерно таким:

```
ftp-virt1
password1
ftp-virt2
password2
```

Сохраняем файл и генерируем локальное хранилище учеток:

```
# db_load -T -t hash -f /etc/vsftpd/virt_users /etc/vsftpd/virt_users.db
```

У вас должен появиться файл `virt_users.db`.

Нужно создать каталоги для этих пользователей:

```
# mkdir /ftp/ftp-virt1 /ftp/ftp-virt2
```

Для папки /ftp надо назначить соответствующего владельца, от которого ftp сервер будет пускать виртуальных пользователей:

```
# chown -R ftp. /ftp
```

На этом настройка виртуальных пользователей ftp закончена. Перезапускаем vsftpd и пробуем залогиниться:

```
# systemctl restart vsftpd
```

Я набросал небольшой скрипт, чтобы было удобно добавлять новых пользователей, предлагаю воспользоваться:

```
# mcedit /etc/vsftpd/add_virt_user.sh
```

```
#!/bin/sh

echo -n "Enter name of virtual user: "
read virtuser

echo -n "Enter password: "
read virtpass

mkdir /ftp/$virtuser
```

```
chown ftp. /ftp/$virtuser
touch /etc/vsftpd/users/$virtuser

echo "$virtuser" >> /etc/vsftpd/virt_users
echo "$virtpass" >> /etc/vsftpd/virt_users

db_load -T -t hash -f /etc/vsftpd/virt_users /etc/vsftpd/virt_users.db
```

Делаете файл исполняемым и запускаете:

```
# chmod 0700 /etc/vsftpd/add_virt_user.sh
# /etc/vsftpd/add_virt_user.sh
Enter name of virtual user: ftp-virt2
Enter password: 123456
```

Все, пользователь добавлен, можно сразу им заходить. Вот так легко и просто настраиваются виртуальные пользователи в vsftpd. Рассмотрим теперь третий вариант, когда эти пользователи хранятся в **mysql базе**.

## Хранение vsftpd пользователей в mysql

Настройка не отличается от предыдущего раздела. Конфиг один в один совпадает. Изменится только файл в `ram.d`. Но перед тем, как его изменить, выполним подготовительные настройки. Первым делом нам понадобится `mariadb`, устанавливаем ее:

```
# yum install mariadb mariadb-server
```

Запускаем и добавляем ее в автозагрузку:

```
# systemctl start mariadb
```

```
# systemctl enable mariadb.service
```

Выполняем скрипт первоначальной настройки mysql:

```
# /usr/bin/mysql_secure_installation
```

Подробнее о настройке mysql рассказано в статье про настройку web сервера на centos. Если вам в будущем понадобится веб сервер, можете сразу его настроить, а заодно поставить phpmyadmin, дальнейшие действия проще будет выполнить там. Но я сейчас не буду на этом останавливаться, сделаем все в консоли.

Подключаемся к mysql:

```
# mysql -u root -p
```

Создаем пользователя и базу данных для хранения учетных записей пользователей ftp:

```
> CREATE DATABASE vsftpd;  
> GRANT SELECT ON vsftpd.* TO 'vsftpd'@'localhost' IDENTIFIED BY '12345';  
> FLUSH PRIVILEGES;
```

vsftpd имя пользователя и базы  
12345 пароль пользователя

Создаем таблицу учеток:

```
> USE vsftpd;
```

```
> CREATE TABLE `accounts` (  
  `id` INT NOT NULL AUTO_INCREMENT PRIMARY KEY ,  
  `username` VARCHAR( 30 ) NOT NULL ,  
  `pass` VARCHAR( 50 ) NOT NULL ,  
  UNIQUE ( `username` )  
 ) ENGINE = MYISAM ;
```

Обратите внимание на кавычки при копировании. Если будут ошибки в синтаксисе mysql, вам нужно будет их удалить и проставить заново.

Создадим сразу одного пользователя:

```
> INSERT INTO accounts (username, pass) VALUES('ftp-mysql', md5('123'));
```

ftp-mysql имя пользователя

123      пароль

Список пользователей можно посмотреть с помощью команды:

```
> select * from accounts;  
+-----+-----+-----+  
| id | username | pass |  
+-----+-----+-----+  
| 1 | ftp-mysql | 202cb962ac59075b964b07152d234b70 |  
+-----+-----+-----+
```

```
1 row in set (0.01 sec)
```

Выходим из консоли mysql и создаем каталог для нового пользователя:

```
# mkdir /ftp/ftp-mysql && chown ftp. /ftp/ftp-mysql
```

Для авторизации пользователей vsftpd через mysql нам необходимо установить **pam\_mysql**. Тут я столкнулся серьезной проблемой. Во-первых, в стандартном репозитории его нет. Раньше он был в репозитории epel, но начиная с 7-й версии его оттуда убрали. Причина в том, что он очень старый, 2008-го года или еще раньше, если я не ошибаюсь. С тех пор он не обновляется. В centos решили, что этот модуль уже не нужен, он морально устарел, а хранить учетные записи в mysql плохая идея. Сейчас в тренде ldap, надо там хранить учетки. В принципе, это разумно. Но отказываться от привычной mysql все равно не хочется.

Мне пришлось прилично загуглиться в буржунет и найти там информацию. Я узнал, что для CentOS 7 подойдет вот эта версия **pam\_mysql-0.7-0.16.rc1.fc20.x86\_64.rpm**. Было очень много битых ссылок, но все же я нашел рабочую. На всякий случай сохраняю ее у себя. Так что ставим модуль pam\_mysql для centos 7 с моего сайта:

```
# rpm -Uvh https://serveradmin.ru/files/pam_mysql-0.7-0.16.rc1.fc20.x86_64.rpm
```

Теперь редактируем файл `/etc/pam.d/vsftpd`. В интернете много примеров конфигурации. Опытным путем собрал рабочий вариант:

```
# mcedit /etc/pam.d/vsftpd

session optional pam_keyinit.so force revoke
auth required pam_mysql.so user=vsftpd passwd=12345 host=localhost db=vsftpd table=accounts usercolumn=username
passwdcolumn=pass crypt=3
account required pam_mysql.so user=vsftpd passwd=12345 host=localhost db=vsftpd table=accounts usercolumn=username
passwdcolumn=pass crypt=3
```

Сохраняем файл, перезапускаем на всякий случай vsftpd и пробуем зайти свежесозданным пользователем.

```
# systemctl restart vsftpd
```

У меня все получилось. Если будут какие-то проблемы, то смотрите в первую очередь на работу `mysql_pam` в логе `/var/log/audit/auditlog`. У меня были такие ошибки, когда я настраивал авторизацию и она не работала:

```
type=USER_AUTH msg=audit(1459442408.756:724): pid=21483 uid=0 auid=4294967295 ses=4294967295 msg='op=PAM:authentication grantors=? acct="ftp-mysql" exe="/usr/sbin/vsftpd" hostname=192.168.1.100 addr=192.168.1.100 terminal=ftp res=failed'
```

После правильной настройки, появились такие записи:

```
type=CRED_ACQ msg=audit(1459442810.698:735): pid=21564 uid=0 auid=4294967295 ses=4294967295 msg='op=PAM:setcred grantors=pam_mysql acct="ftp-mysql" exe="/usr/sbin/vsftpd" hostname=192.168.1.100 addr=192.168.1.100 terminal=ftp res=success'
```

В логе `/var/log/secure` тоже может быть полезная информация. У меня было так:

```
Mar 31 19:39:44 vsftpd[21475]: pam_mysql - MySQL error(Unknown column 'passwd' in 'field list')
```

С пробелами в строках была путаница, пока не исправил, не работало ничего. Но потом все заработало. Если ошибетесь с пользователем `mysql`, от которого подключается `vsftpd`, получите ошибку:

```
pam_mysql - MySQL error (Access denied for user 'vsftpd'@'localhost' (using password: YES))
```

Проверяйте логин с паролем в рат файле. На этом у меня сегодня все. Можно скрипт набросать для удобного добавления пользователей в базу, но мне уже лениво. Может кто-нибудь другой в комментариях поделится.

## Настройка vsftpd в Centos 8

Для настройки ftp сервера в Centos 8, необходимо выполнить несколько изменений по сравнению с основной статьей:

1. Параметр **tcp\_wrappers** необходимо отключить, т.е. указать **NO**. В Centos 8 нет его поддержки. Это касается всех приведенных в статье конфигов.
2. Если вы настраиваете виртуальных пользователей vsftpd в Centos 8, то ставить пакет compat-db нет необходимости. Утилита **db\_load** уже есть в системе.
3. Модуль **pam\_mysql** для Centos 8 брать по ссылке - [https://serveradmin.ru/files/pam\\_mysql-0.8.1-0.6.el8.x86\\_64.rpm](https://serveradmin.ru/files/pam_mysql-0.8.1-0.6.el8.x86_64.rpm).

Все остальное настраивается точно так же, как и в предыдущей версии системы.

## Настройка SSL / TLS

Для того, чтобы ваши пароли и данные, которые ftp сервер передает в открытом виде, не стали легкой добычей sniffеров, рекомендую настроить ssl или tls сертификаты для работы ftp через ssl (ftps, не путать с sftp). Сделать это не сложно. Настройка одинаковая для обеих систем.

Генерируем самоподписный сертификат. Если есть валидный, или есть возможность получить бесплатный сертификат от let's encrypt, можно использовать их.

```
# openssl req -x509 -nodes -days 3650 -newkey rsa:2048 -keyout /etc/vsftpd/vsftpd.pem -out /etc/vsftpd/vsftpd.pem
```



```
[root@centos8 vsftpd]# openssl req -x509 -nodes -days 3650 -newkey rsa:2048 -keyout /etc/vsftpd/vsftpd.pem -out /etc/vsftpd/vsftpd.pem
Generating a RSA private key
.....+++++
.....+++++
writing new private key to '/etc/vsftpd/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) [XX]:RU
State or Province Name (full name) []:MSK
Locality Name (eg, city) [Default City]:MSK
Organization Name (eg, company) [Default Company Ltd]:IT
Organizational Unit Name (eg, section) []:it
Common Name (eg, your name or your server's hostname) []:vsftpd
Email Address []:
```

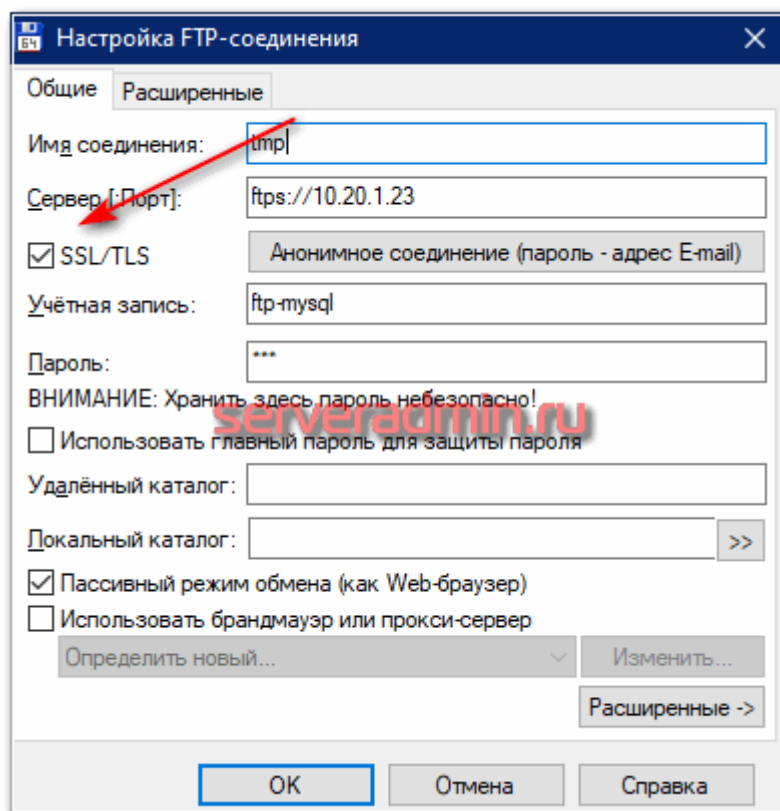
Данные можете любые указывать, это не имеет значения. Добавляете в конфиг vsftpd.conf дополнительные параметры:

```
rsa_cert_file=/etc/vsftpd/vsftpd.pem
rsa_private_key_file=/etc/vsftpd/vsftpd.pem
ssl_enable=YES
```

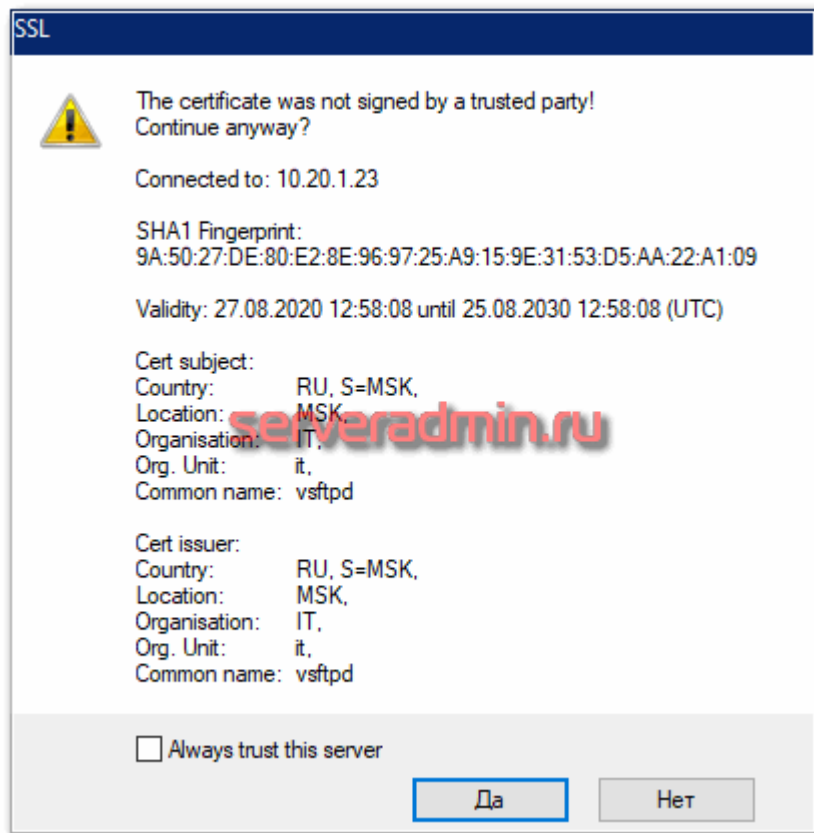
И перезапускаете ftp сервер, чтобы проверить работу по ftps.

```
# systemctl restart vsftpd
```

Теперь можно подключаться и проверять.



При подключении, должно выйти предупреждение о том, что нет возможности проверить сертификат. Надо вручную подтвердить доверие.



Вот и все. Теперь ftp сервером можно безопасно пользоваться, так как данные не передаются в открытом виде. Так что настоятельно рекомендую ssl в ftp настраивать.

## Заключение

Мы подробно рассмотрели вопрос использования vsftpd сервера с различной настройкой учетных записей. По мне удобнее всего работать с виртуальными

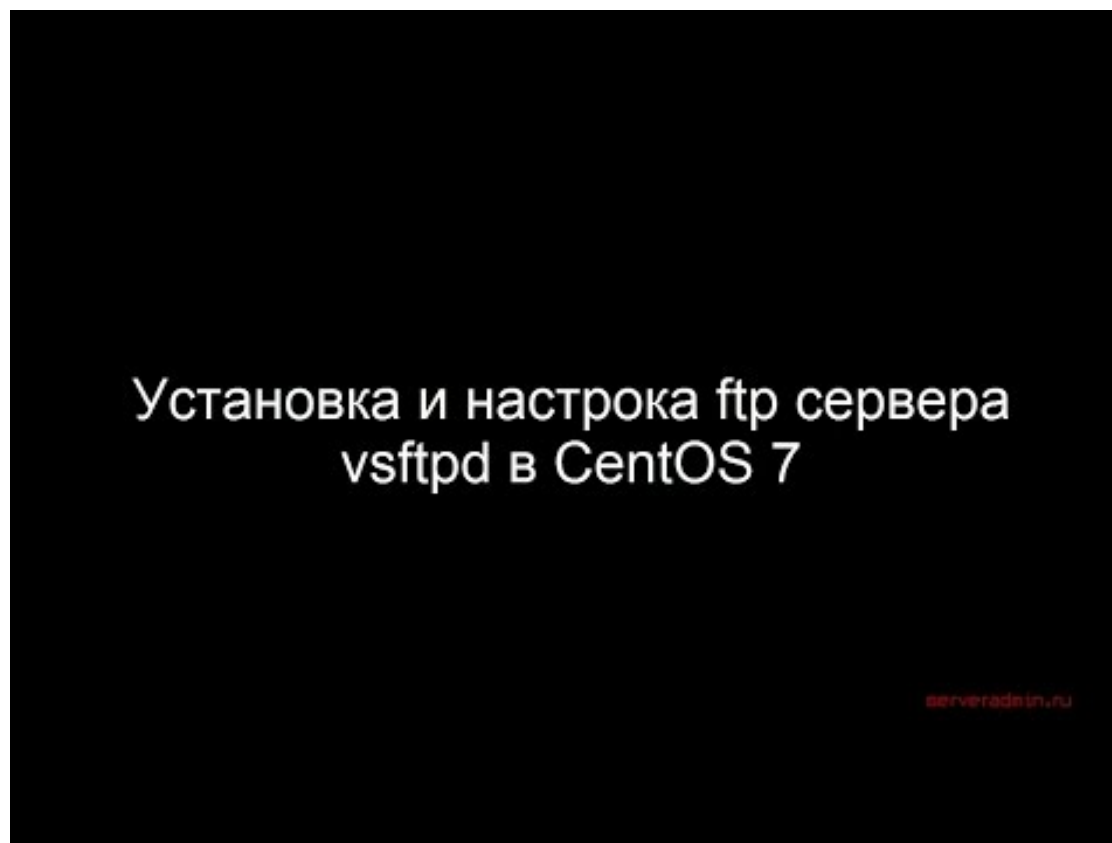
пользователями в локальной, не mysql, базе. Но если используется какая-нибудь панель управления сервером, то удобно засунуть всех пользователей в базу.

Когда я настраивал публичный хостинг, то использовал самописные скрипты для создания каталогов виртуальных хостов, добавления настроек в apache и создание пользователей ftp с доступом к своим сайтам. Там без ftp не обойтись, так как его постоянно просят веб мастера, даже в наше время.

Помогла статья? Подписывайся на telegram канал автора

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

## Видео



Watch this video on YouTube

## Онлайн курс MS SQL Server Developer

Если у вас есть желание научиться обрабатывать миллиарды данных, рекомендую познакомиться с **онлайн-курсом "MS SQL Server Developer"** в OTUS. Курс не для новичков, для поступления нужны базовые знания по программированию, работе с БД и SQL. Обучение длится 4 месяца, после чего успешные выпускники курса смогут пройти собеседования у партнеров. После обучения вы сможете:

- разрабатывать на SQL;
- проектировать БД и понимать все нюансы;
- анализировать и оптимизировать производительности запросов;
- писать сложные хранимые процедуры, функции и триггеры;
- читать план запроса.

Проверьте себя на вступительном тесте и смотрите подробнее программу по .