How to Secure a FTP Server Using SSL/TLS for Secure File Transfer in CentOS 7

By its original design, **FTP** (**File Transfer Protocol**) is not secure, meaning it doesn't encrypt data being transmitted between two machines, along with user's credentials. This poses a massive threat to data as well as server security.

In this tutorial, we will explain how to manually enable data encryption services in a FTP server in CentOS/RHEL 7 and Fedora; we will go through various steps of securing **VSFTPD** (**Very Secure FTP Daemon**) services using **SSL/TLS** certificates.

Prerequisites:

1. You must have installed and configured a FTP server in CentOS 7

Before we start, note that all the commands in this tutorial will be run as **root**, otherwise, use the sudo command to gain root privileges if you are not controlling the server using the root account.

Step 1. Generating SSL/TLS Certificate and Private Key

1. We need to start by creating a subdirectory under: `/etc/ssl/` where we will store the **SSL/TLS** certificate and key files:

```
# mkdir /etc/ssl/private
```

2. Then run the command below to create the certificate and key for **vsftpd** in a single file, here is the explanation of each flag used.

1. **req** – is a command for X.509 Certificate Signing Request (CSR) management.
2. **x509** – means X.509 certificate data management.
3. **days** – defines number of days certificate is valid for.
4. **newkey** – specifies certificate key processor.
5. **rsa:2048** – RSA key processor, will generate a 2048 bit private key.
6. **keyout** – sets the key storage file.
7. **out** – sets the certificate storage file, note that both certificate and key are stored in the same file: **/etc/ssl/private/vsftpd.pem**.

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

The above command will ask you to answer the questions below, remember to use values that apply to your scenario.

```
Country Name (2 letter code) [XX]:IN
State or Province Name (full name) []:Lower Parel
Locality Name (eg, city) [Default City]:Mumbai
Organization Name (eg, company) [Default Company Ltd]:TecMint.com
Organizational Unit Name (eg, section) []:Linux and Open Source
Common Name (eg, your name or your server's hostname) []:tecmint
Email Address []:admin@tecmint.com
```

Step 2. Configuring VSFTPD To Use SSL/TLS

3. Before we perform any VSFTPD configurations, let's open the ports **990** and **40000-50000** to allow TLS connections and the port range of passive ports to define in the VSFTPD configuration file respectively:

```
# firewall-cmd --zone=public --permanent --add-port=990/tcp
# firewall-cmd --zone=public --permanent --add-port=40000-50000/tcp
# firewall-cmd --reload
```

4. Now, open the VSFTPD config file and specify the SSL details in it:

```
# vi /etc/vsftpd/vsftpd.conf
```

Look for the option **ssl_enable** and set its value to `YES` to activate the use of SSL, in addition, since TSL is more secure than SSL, we will restrict VSFTPD to employ TLS instead, using the **ssl_tlsv1_2** option:

```
ssl_enable=YES
ssl_tlsv1_2=YES
ssl_sslv2=NO
ssl_sslv3=NO
```

5. Then, add the lines below to define the location of the SSL certificate and key file:

```
rsa_cert_file=/etc/ssl/private/vsftpd.pem
rsa_private_key_file=/etc/ssl/private/vsftpd.pem
```

6. Next, we have to prevent anonymous users from using SSL, then force all non-anonymous logins to use a secure SSL connection for data transfer and to send the password during login:

```
allow_anon_ssl=NO
force_local_data_ssl=YES
force_local_logins_ssl=YES
```

7. In addition, we can add the options below to boost up FTP server security. When option **require_ssl_reuse** is set to YES, then, all SSL data connections are required to exhibit SSL session reuse; proving that they know the same master secret as the control channel.

Therefore, we have to turn it off.

```
require_ssl_reuse=NO
```

Again, we need to select which SSL ciphers VSFTPD will permit for encrypted SSL connections with the **ssl_ciphers** option. This can greatly limit efforts of attackers who try to force a particular cipher which they probably discovered vulnerabilities in:

```
ssl_ciphers=HIGH
```

8. Now, set the port range (min and max port) of passive ports.

```
pasv_min_port=40000
pasv_max_port=50000
```

9. Optionally, allow SSL debugging, meaning openSSL connection diagnostics are recorded to the VSFTPD log file with the **debug_ssl** option:

```
debug_ssl=YES
```

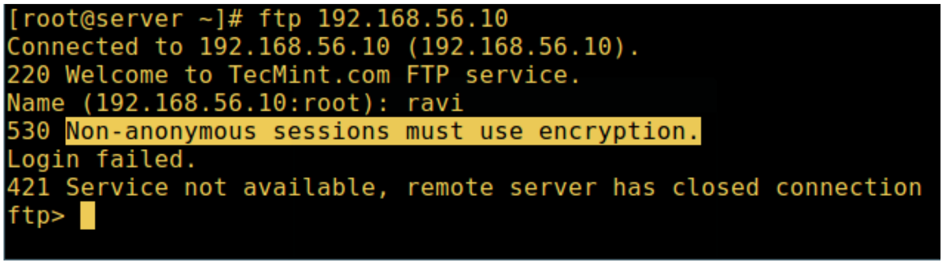
Save all the changes and close the file. Then let’s restart VSFTPD service:

```
# systemctl restart vsftpd
```

Step 3: Testing FTP server With SSL/TLS Connections

10. After doing all the above configurations, test if VSFTPD is using SSL/TLS connections by attempting to use FTP from the command line as follows:

```
# ftp 192.168.56.10
Connected to 192.168.56.10  (192.168.56.10).
220 Welcome to TecMint.com FTP service.
Name (192.168.56.10:root) : ravi
530 Non-anonymous sessions must use encryption.
Login failed.
421 Service not available, remote server has closed connection
ftp>
```



```
[root@server ~]# ftp 192.168.56.10
Connected to 192.168.56.10 (192.168.56.10).
220 Welcome to TecMint.com FTP service.
Name (192.168.56.10:root): ravi
530 Non-anonymous sessions must use encryption.
Login failed.
421 Service not available, remote server has closed connection
ftp>
```

Verify FTP SSL Secure Connection

From the screen shot above, we can see that there is an error informing us that VSFTPD can only allow user to login from clients that support encryption services.

The command line does not offer encryption services thus producing the error. So, to securely connect to the server, we need a FTP client that supports SSL/TLS connections such as **FileZilla**.

Step 4: Install FileZilla to Securely Connect to a FTP Server

11. FileZilla is a modern, popular and importantly cross-platform FTP client that supports SSL/TLS connections by default.

To install FileZilla in Linux, run the command below:

--------- On CentOS/RHEL/Fedora ---------

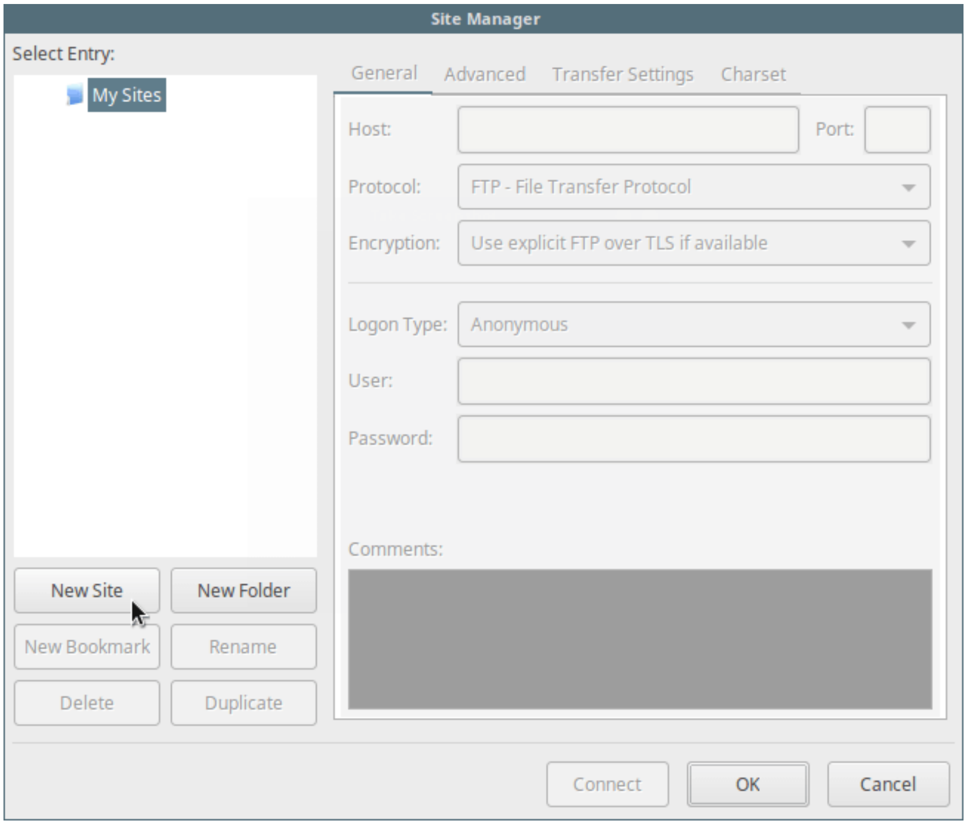
```
# yum install epel-release filezilla
```

--------- On Debian/Ubuntu ---------

```
$ sudo apt-get install  filezilla
```

12. When the installation completes (or else if you already have it installed), open it and go to **File=>Sites Manager** or (press `Ctrl+S`) to get the **Site Manager** interface below.

Click on **New Site** button to add a new site/host connection details.



Protocol: FTP - File Transfer Protocol

Encryption: Use explicit FTP over TLS if available

Logon Type: Anonymous

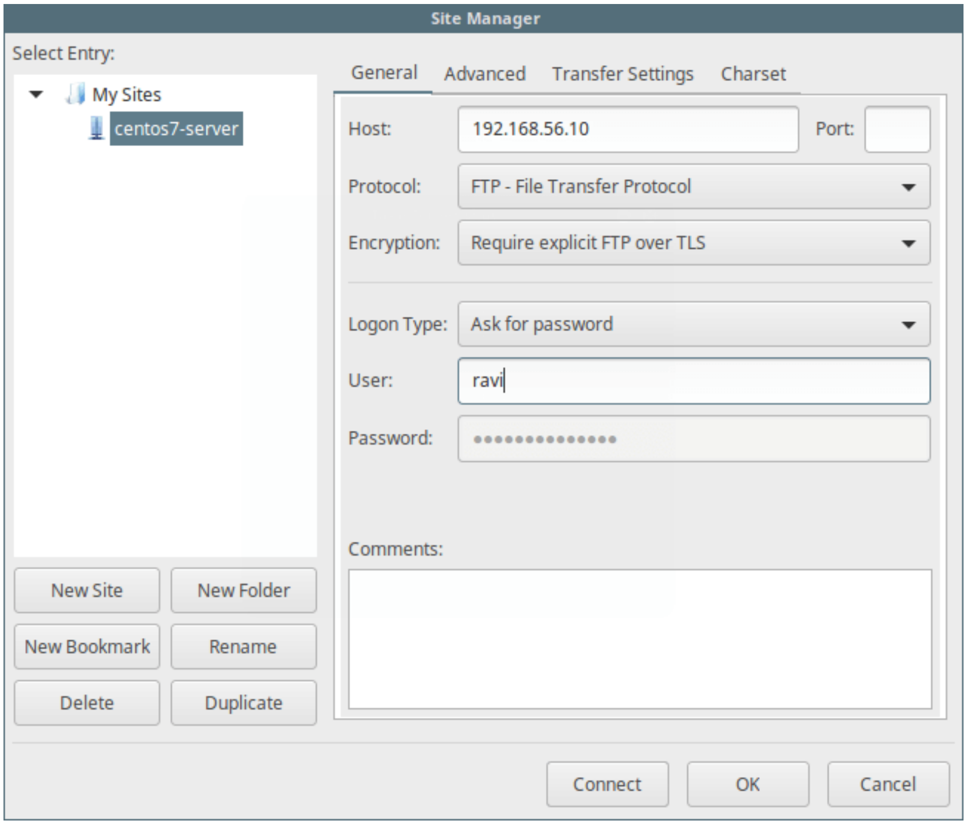
User:

Password:

Add New FTP Site in Filezilla

13. Next, set the host/site name, add the IP address, define the protocol to use, encryption and logon type as in the screen shot below (use values that apply to your scenario):

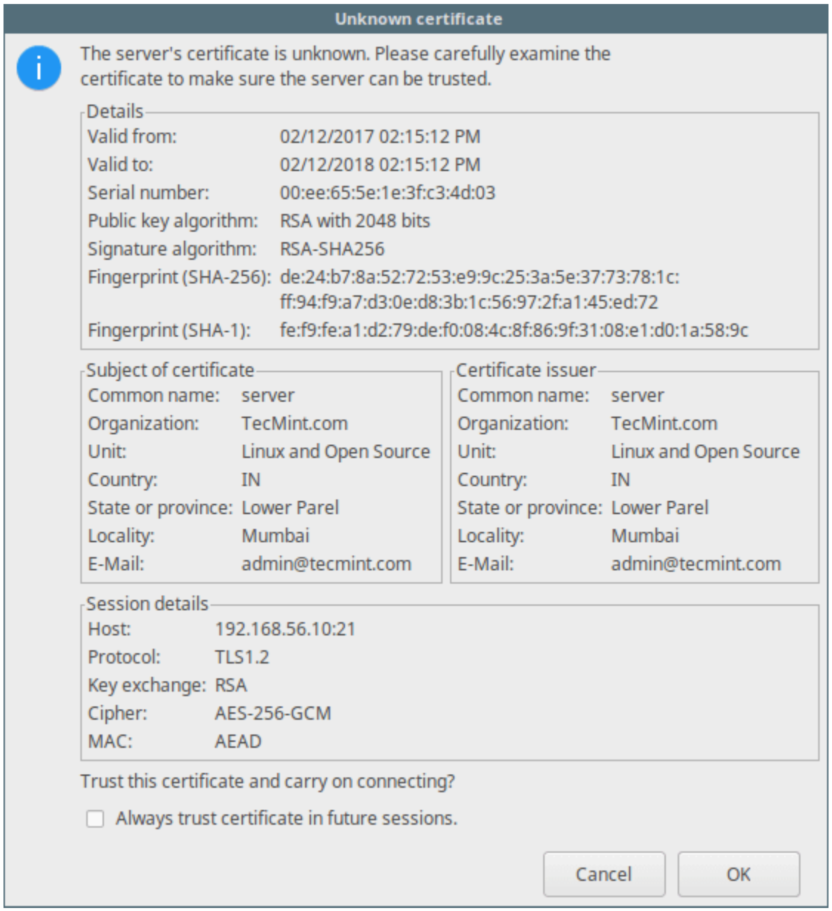
Host: **192.168.56.10**
Protocol: **FTP – File Transfer Protocol**
Encryption: **Require explicit FTP over**   #recommended
Logon Type: **Ask for password**   #recommended
User: **username**



Host: 192.168.56.10
Port:
Protocol: FTP - File Transfer Protocol
Encryption: Require explicit FTP over TLS
Logon Type: Ask for password
User: ravi
Password: ••••••••••••••
Comments:

Add FTP Server Details in Filezilla

14. Then click on **Connect** to enter the password again, and then verify the certificate being used for the SSL/TLS connection and click `OK` once more to connect to the FTP server:



The server's certificate is unknown. Please carefully examine the certificate to make sure the server can be trusted.

Details

Valid from:	02/12/2017 02:15:12 PM
Valid to:	02/12/2018 02:15:12 PM
Serial number:	00:ee:65:5e:1e:3f:c3:4d:03
Public key algorithm:	RSA with 2048 bits
Signature algorithm:	RSA-SHA256
Fingerprint (SHA-256):	de:24:b7:8a:52:72:53:e9:9c:25:3a:5e:37:73:78:1c: ff:94:f9:a7:d3:0e:d8:3b:1c:56:97:2f:a1:45:ed:72
Fingerprint (SHA-1):	fe:f9:fe:a1:d2:79:de:f0:08:4c:8f:86:9f:31:08:e1:d0:1a:58:9c

Subject of certificate

Common name:	server
Organization:	TecMint.com
Unit:	Linux and Open Source
Country:	IN
State or province:	Lower Parel
Locality:	Mumbai
E-Mail:	admin@tecmint.com

Certificate issuer

Common name:	server
Organization:	TecMint.com
Unit:	Linux and Open Source
Country:	IN
State or province:	Lower Parel
Locality:	Mumbai
E-Mail:	admin@tecmint.com

Session details

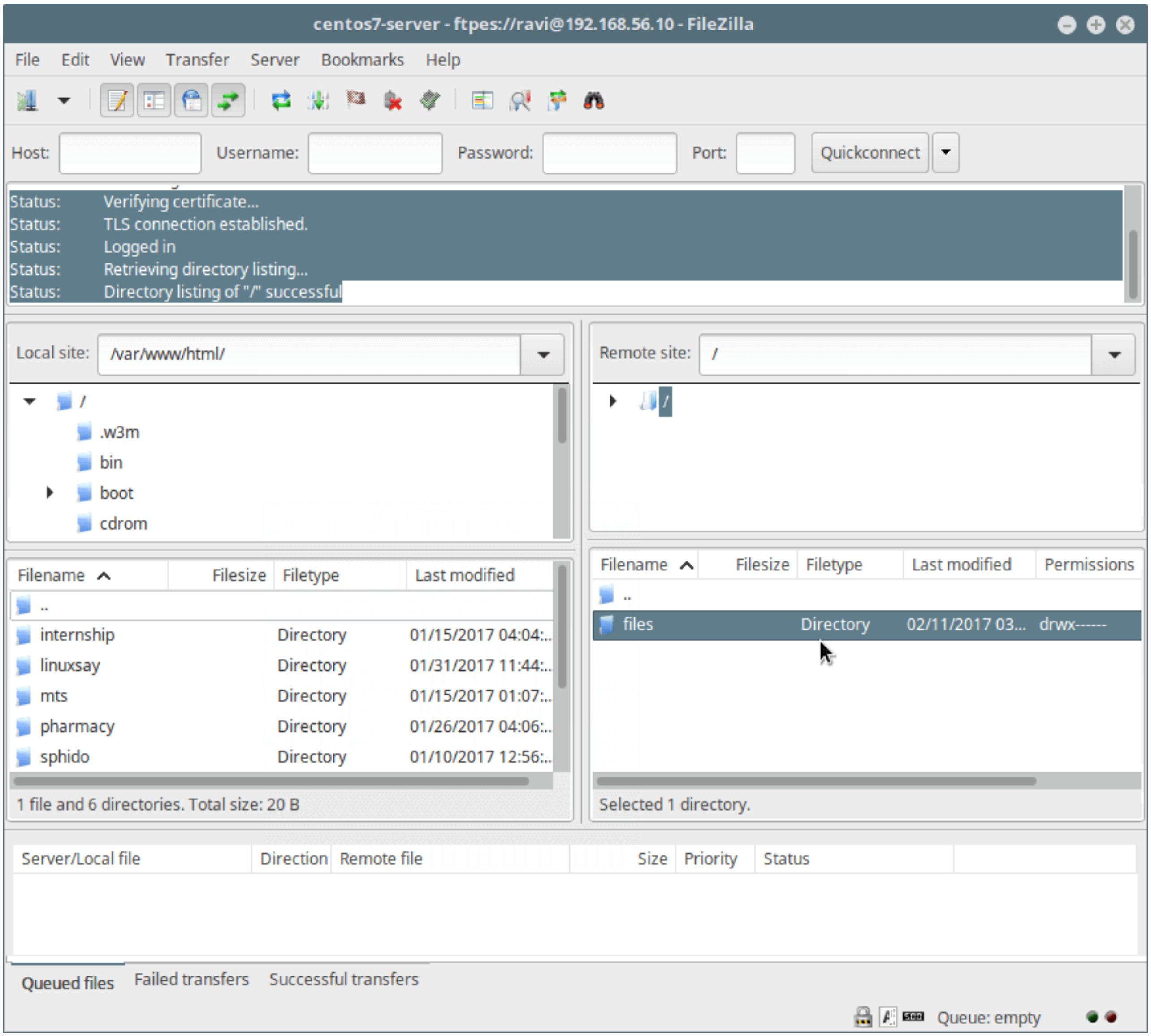
Host:	192.168.56.10:21
Protocol:	TLS1.2
Key exchange:	RSA
Cipher:	AES-256-GCM
MAC:	AEAD

Trust this certificate and carry on connecting?

☐ Always trust certificate in future sessions.

Verify FTP SSL Certificate

At this stage, we should have logged successfully into the FTP server over a TLS connection, check the connection status section for more information from the interface below.



```
Status:    Verifying certificate...
Status:    TLS connection established.
Status:    Logged in
Status:    Retrieving directory listing...
Status:    Directory listing of "/" successful
```

Local site: /var/www/html/

- /
  - .w3m
  - bin
  - boot
  - cdrom

Filename	Filesize	Filetype	Last modified
..			
internship		Directory	01/15/2017 04:04:...
linuxsay		Directory	01/31/2017 11:44:...
mts		Directory	01/15/2017 01:07:...
pharmacy		Directory	01/26/2017 04:06:...
sphido		Directory	01/10/2017 12:56:...

1 file and 6 directories. Total size: 20 B

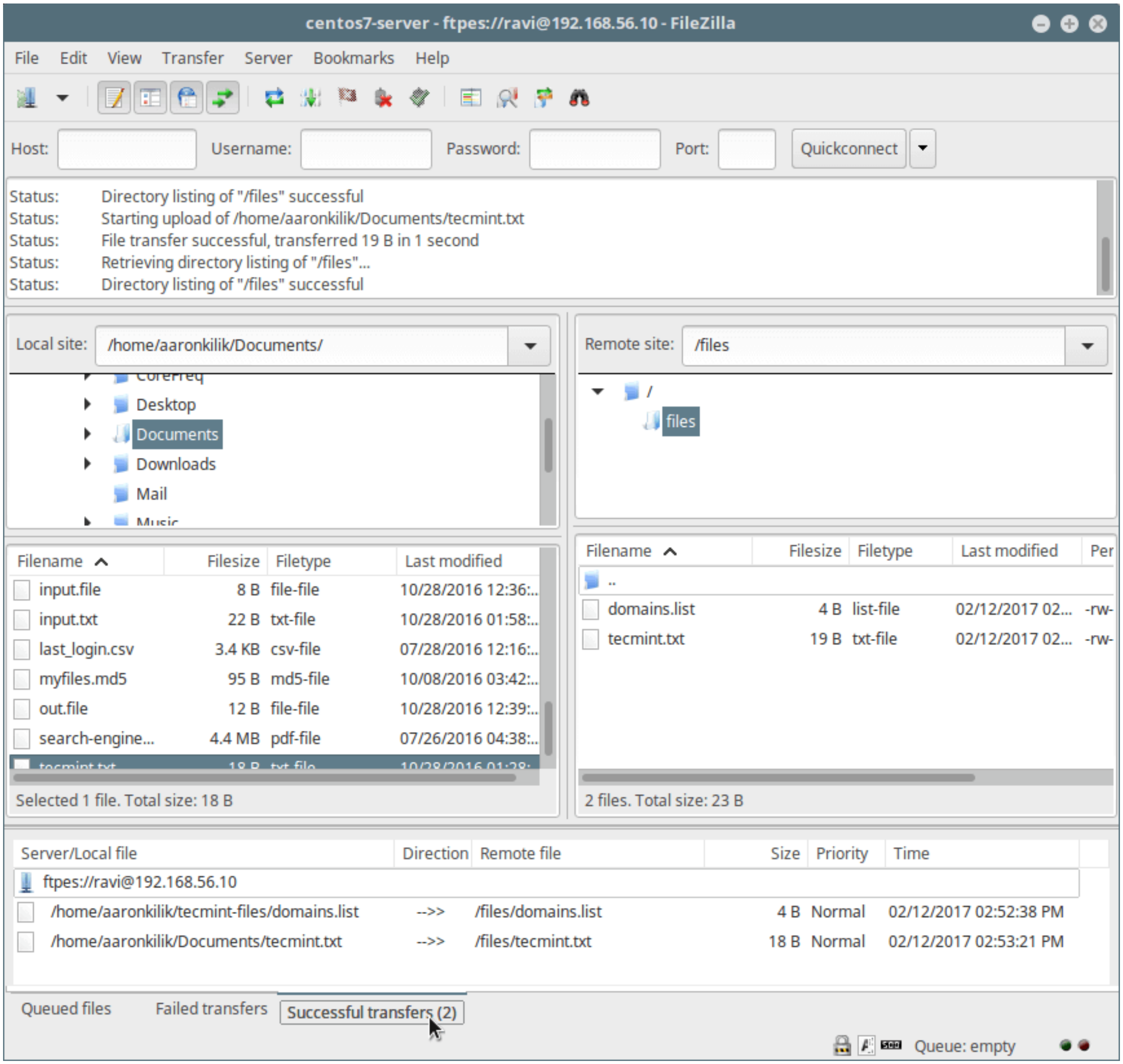
Remote site: /

Filename	Filesize	Filetype	Last modified	Permissions
..				
files		Directory	02/11/2017 03...	drwx------

Selected 1 directory.

Connected to FTP Server Over TLS/SSL

15. Last but not least, try transferring files from the local machine to the FTP sever in the files folder, take a look at the lower end of the **FileZilla** interface to view reports concerning file transfers.



```
Status:    Directory listing of "/files" successful
Status:    Starting upload of /home/aaronkilik/Documents/tecmint.txt
Status:    File transfer successful, transferred 19 B in 1 second
Status:    Retrieving directory listing of "/files"...
Status:    Directory listing of "/files" successful
```

Local site: /home/aaronkilik/Documents/

- Corerreq
- Desktop
- Documents
- Downloads
- Mail
- Music

Filename	Filesize	Filetype	Last modified
input.file	8 B	file-file	10/28/2016 12:36:...
input.txt	22 B	txt-file	10/28/2016 01:58:...
last_login.csv	3.4 KB	csv-file	07/28/2016 12:16:...
myfiles.md5	95 B	md5-file	10/08/2016 03:42:...
out.file	12 B	file-file	10/28/2016 12:39:...
search-engine...	4.4 MB	pdf-file	07/26/2016 04:38:...
tecmint.txt	18 B	txt-file	10/28/2016 01:28:...

Selected 1 file. Total size: 18 B

Remote site: /files

- /
  - files

Filename	Filesize	Filetype	Last modified	Per
..				
domains.list	4 B	list-file	02/12/2017 02...	-rw-
tecmint.txt	19 B	txt-file	02/12/2017 02...	-rw-

2 files. Total size: 23 B

Server/Local file	Direction	Remote file	Size	Priority	Time
ftpes://ravi@192.168.56.10					
/home/aaronkilik/tecmint-files/domains.list	-->>	/files/domains.list	4 B	Normal	02/12/2017 02:52:38 PM
/home/aaronkilik/Documents/tecmint.txt	-->>	/files/tecmint.txt	18 B	Normal	02/12/2017 02:53:21 PM

Transfer Files Securely Using FTP

That's all! Always keep in mind that FTP is not secure by default, unless we configure it to use SSL/TLS connections as we showed you in this tutorial. Do share your thoughts about this tutorial/topic via the feedback form below.

If You Appreciate What We Do Here On TecMint, You Should Consider:

TecMint is the fastest growing and most trusted community site for any kind of Linux Articles, Guides and Books on the web. Millions of people visit TecMint! to search or browse the thousands of published articles available FREELY to all.

If you like what you are reading, please consider buying us a coffee (or 2) as a token of appreciation.

We are thankful for your never ending support.