iGNITE Technologies

FTP Penetration TESTING

WWW.HACKINGARTICLES.IN

Contents

Introduction to FTP

FTP is a file transfer protocol, used to transfer files between a network using TCP/IP connections via Port 20/21. It is basically a client-server protocol. As it works on TCP, it requires two communication channels between client and server: a command channel and a data channel. The command channel is for controlling the conversation between client and server, whereas the data connection is initiated by the server to transfer data.

Uses of FTP

- An FTP site is a web site where users can easily upload or download specific files.
- FTP by mail allows users without access to the Internet to access and copy files using anonymous FTP by sending an email message to ftpmail@decwrl.dec.com and putting the word help in the body of the text.
- FTP Explorer is an FTP client based on Windows 95 file manager (Windows 95 Explorer).
- An FTP server is a dedicated computer which provides FTP service. This invites hackers and necessitates security hardware or software such as utilizing usernames, passwords, and file access control.
- An FTP client is a computer application which accesses an FTP server. While doing so, users should block incoming FTP connection attempts using passive mode and should check for viruses on all downloaded files.

Working of FTP

FTP works just like HTTP and SMB protocols. When the FTP server is configured on a network, then a specific folder is defined as a shared folder in order to share files. Users can access this file server via FTP. FTP is often authenticated by a sign-in protocol. However, an FTP server may be configured to accept anonymous login credentials as well. But now, it's mostly FTP with SSL/TLS.

When transferring files through FTP, the user's machine is called the "local host machine" and is connected to the internet. Another machine is called the remote host, which has FTP running on it and is also connected to the internet. Now, in order to transfer the files, the local host machine connects to the remote host's IP. Then the user must enter the username and password. FTP always provides a GUI which makes file transfer user-friendly. Here, you can transfer files by the drag-and-drop method. Otherwise, you can simply use FTP commands for the desired transfer.

Penetration Testing on FTP

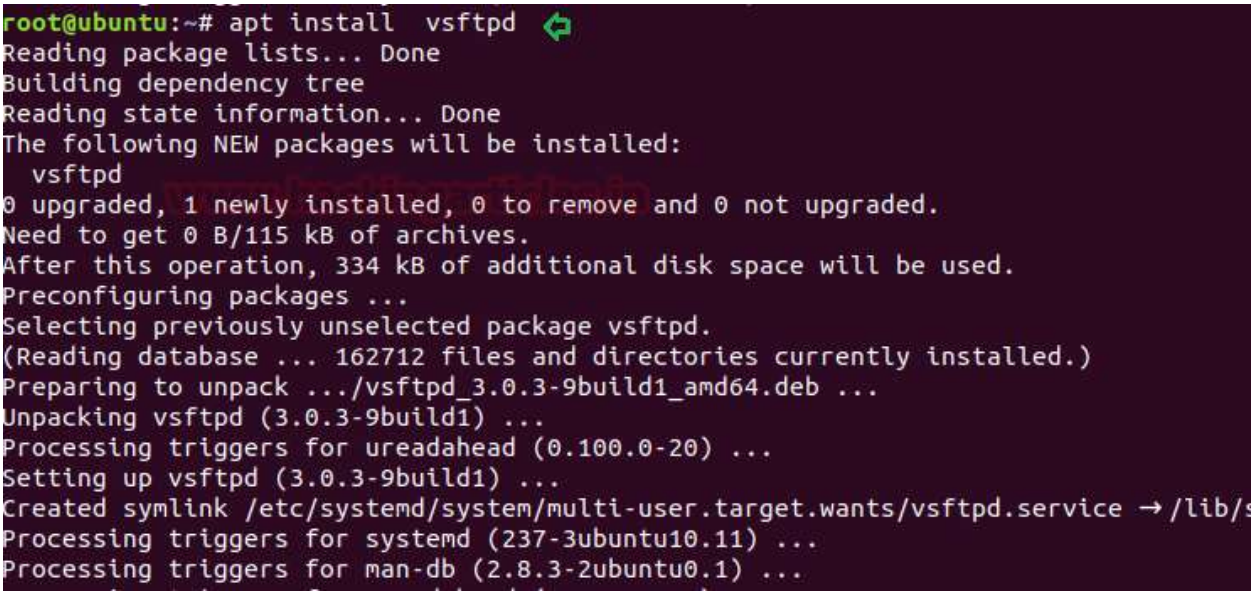
Requirements:

- FTP server: Ubuntu
- Attacking machine: Kali
- Client machine: Windows

Installation of FTP

Installation FTP is quite easy. To install FTP, open the terminal in ubuntu as root user and type:

```
apt install vsftpd
```



Once FTP is installed use nmap to confirm and to do so, type the following command:

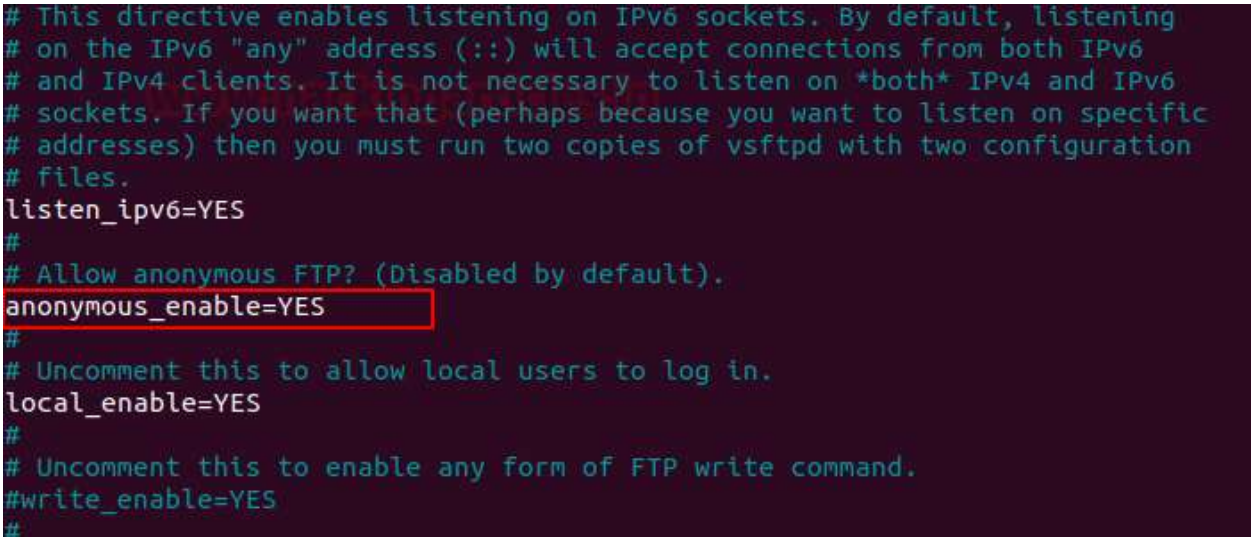
```
nmap -p21 192.168.1.102
```

As you can see that FTP is working on port 21.

Anonymous Login

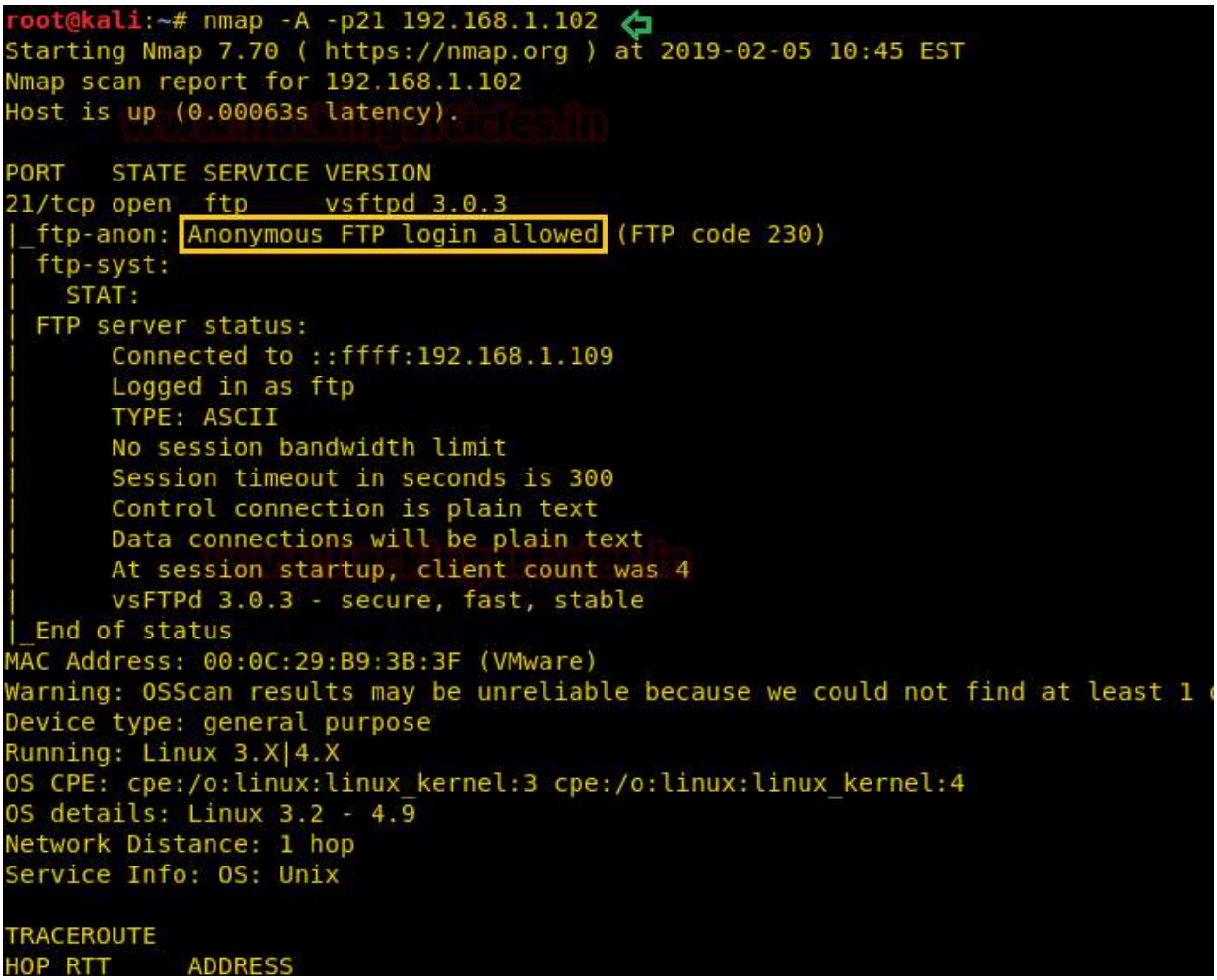
As I have mentioned before, FTP credentials can be set to anonymous and this is often found on many FTP servers. FTP users may authenticate themselves with a **clear-text sign-in protocol**, normally in the form of a username and password, but can connect **anonymously** if the server is configured to allow it. So, let's see how it will be done by first configuring it to be anonymous. Use nano or another text editor to open vsftpd.conf.Find the "anonymous_enable=NO" statement as shown in the image below.

Change NO to YES to enable anonymous as shown here:



Now let's check it from nmap by using the following command:

nmap -A -p21 192.168.1.102



As the result shown by nmap, you can see that port 21 is open and you some details about it to like its version. Now, let's try and log in FTP using anonymous as our credentials. Now, let's try and login:

ftp 192.168.1.102

Enter anonymous as username and password as shown in the image below as you will find you in the ftp server.

```
root@kali:~# ftp 192.168.1.102
Connected to 192.168.1.102.
220 (vsFTPd 3.0.3)
Name (192.168.1.102:root): Anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
226 Directory send OK.
ftp> help
Commands may be abbreviated.  Commands are:

!               dir             mdelete         qc              site
$               disconnect      mdir            sendport        size
account         exit            mget            put             status
append          form            mkdir           pwd             struct
ascii           get             mls             quit            system
bell            glob            mode            quote           sunique
binary          hash            modtime         recv            tenex
bye             help            mput            reget           tick
case            idle            newer           rstatus         trace
cd              image           nmap            rhelp           type
cdup            ipany           nlist           rename          user
chmod           ipv4            ntrans          reset           umask
close           ipv6            open            restart         verbose
cr              lcd             prompt          rmdir           ?
delete          ls              passive         runique
debug           macdef          proxy           send
ftp>
```

Disable FTP_banner

Now if you scan ftp from nmap you will its version:

nmap -sV -p21 192.168.1.102

```
root@kali:~# nmap -sV -p21 192.168.1.102
Starting Nmap 7.70 ( https://nmap.org ) at 2019-02-05 10:50 EST
Nmap scan report for 192.168.1.102
Host is up (0.00047s latency).

PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.3
MAC Address: 00:0C:29:B9:3B:3F (VMware)
Service Info: OS: Unix
```

As this visibility of the version can leave you vulnerable to various exploits, let's now learn how you will protect yourself by hiding the banner of FTP. For this, again open the vsftpd.conf file using any desired text editor.

```
# raw file.
# ASCII mangling is a horrible feature of the protocol.
#ascii_upload_enable=YES
#ascii_download_enable=YES
#
# You may fully customise the login banner string:
#ftpd_banner=Welcome to blah FTP service.
#
# You may specify a file of disallowed anonymous e-mail addresses. Apparently
# useful for combatting certain DoS attacks.
#deny_email_enable=YES
# (default follows)
#banned_email_file=/etc/vsftpd.banned_emails
```

Find the line "ftpd_banner=welcome to blah FTP service" in the conf file.From this statement, remove the # symbol as shown in the image below:

```
# raw file.
# ASCII mangling is a horrible feature of the protocol.
#ascii_upload_enable=YES
#ascii_download_enable=YES
#
# You may fully customise the login banner string:
ftpd_banner=Welcome to blah FTP service.
#
# You may specify a file of disallowed anonymous e-mail addresses. Apparently
# useful for combatting certain DoS attacks.
#deny_email_enable=YES
# (default follows)
#banned_email_file=/etc/vsftpd.banned_emails
#
```

Now if you scan again with nmap, it will hide the banner. Try it by using the following command:

nmap -sV -p21 192.168.1.102

```
root@kali:~# nmap -sV -p21 192.168.1.102
Starting Nmap 7.70 ( https://nmap.org ) at 2019-02-05 10:52 EST
Nmap scan report for 192.168.1.102
Host is up (0.00046s latency).

PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd
MAC Address: 00:0C:29:B9:3B:3F (VMware)
Service Info: OS: Unix
```

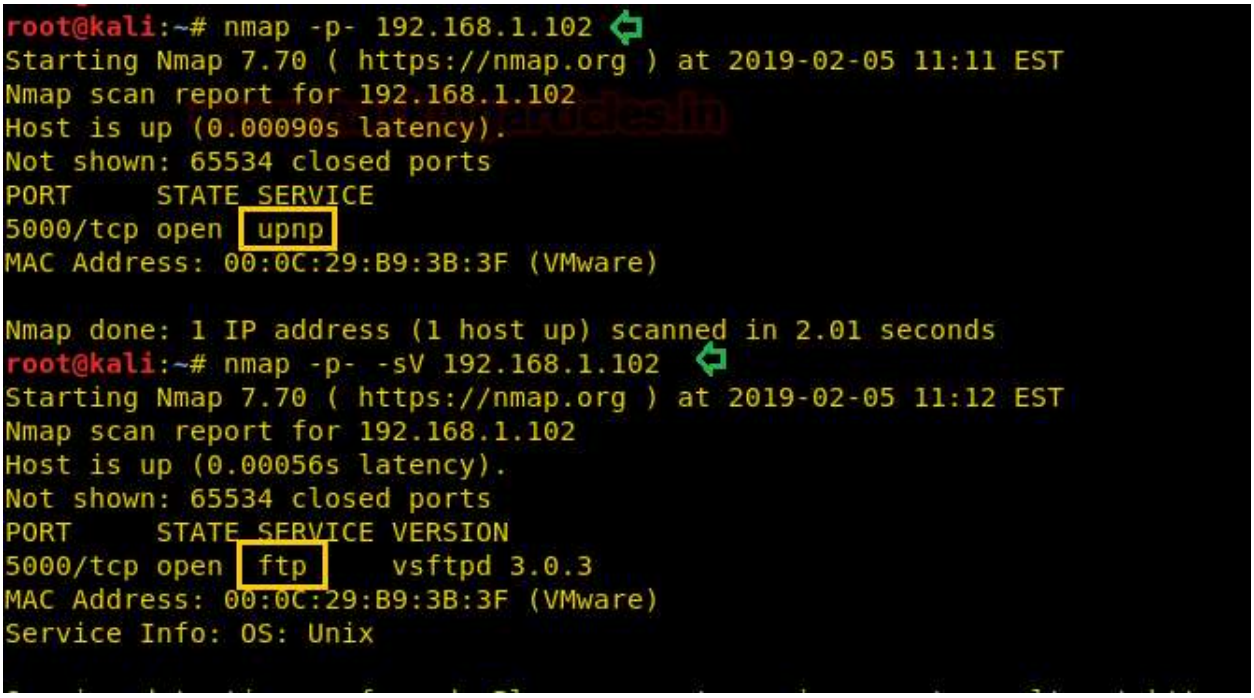
Switch Port for FTP Service

Like this, you can add another security layer by changing the port of ftp. You can start the service of FTP on any port you like. Here, we have shifted the FTP port to 5000. Find the line "listen_port=21" in the ftp conf file to do this.Change the port number to 5000, or any other number as you desire, as shown in the image below:

```
# in  your  local  time  zone.  The default is to display GMT. The
# times returned by the MDTM FTP command are also affected by this
# option.
use_localtime=YES
#
# Activate logging of uploads/downloads.
xferlog_enable=YES
#
# Make sure PORT transfer connections originate from port 20 (ftp-data).
connect_from_port_20=NO
listen_port=5000
#
# If you want, you can arrange for uploaded anonymous files to be owned by
# a different user. Note! Using "root" for uploaded files is not
# recommended!
#chown_uploads=YES
#chown_username=whoever
#
# You may override where the log file goes if you like. The default is shown
# below.
#xferlog_file=/var/log/vsftpd.log
```

Save the file and restart the FTP service. If you scan from nmap, you will find the port is now at 5000. Applying such a layer of security helps to confuse attackers.

nmap -p- 192.168.1.102



Sniffing FTP Login Credential

By default, the traffic sent to and received from FTP is not encrypted. An attacker can use sniffing tools to intercept data packets travelling between a server and a client in a network in order to obtain credentials. then use them for unauthorised access. As we have discussed above, FTP users may authenticate themselves with a **clear-text sign-in protocol** for username and password.

ftp 192.168.1.102 5000

```
root@kali:~# ftp 192.168.1.102 5000
Connected to 192.168.1.102.
220 (vsFTPd 3.0.3)
Name (192.168.1.102:root): raj
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
drwxr-xr-x    3 1000     1000         4096 Feb 05 07:32 Desktop
drwxr-xr-x    2 1000     1000         4096 Feb 05 07:16 Documents
drwxr-xr-x    2 1000     1000         4096 Feb 05 07:16 Downloads
drwxr-xr-x    2 1000     1000         4096 Feb 05 07:16 Music
drwxr-xr-x    2 1000     1000         4096 Feb 05 07:16 Pictures
drwxr-xr-x    2 1000     1000         4096 Feb 05 07:16 Public
drwxr-xr-x    2 1000     1000         4096 Feb 05 07:16 Templates
drwxr-xr-x    2 1000     1000         4096 Feb 05 07:16 Videos
-rw-r--r--    1 1000     1000         8980 Feb 05 07:04 examples.desktop
226 Directory send OK.
ftp>
```

Similarly, if we capture TCP packet through **Wireshark** for sniffing FTP credential. So, now try and log in to ftp using the following commands:

Give the username and password.

Capture the traffic using Wireshark. Now, in Wireshark, if you follow the TCP stream of the packet, you can see the login credentials in clear text as shown in the following image:

Use SSL Certificate against Sniffing

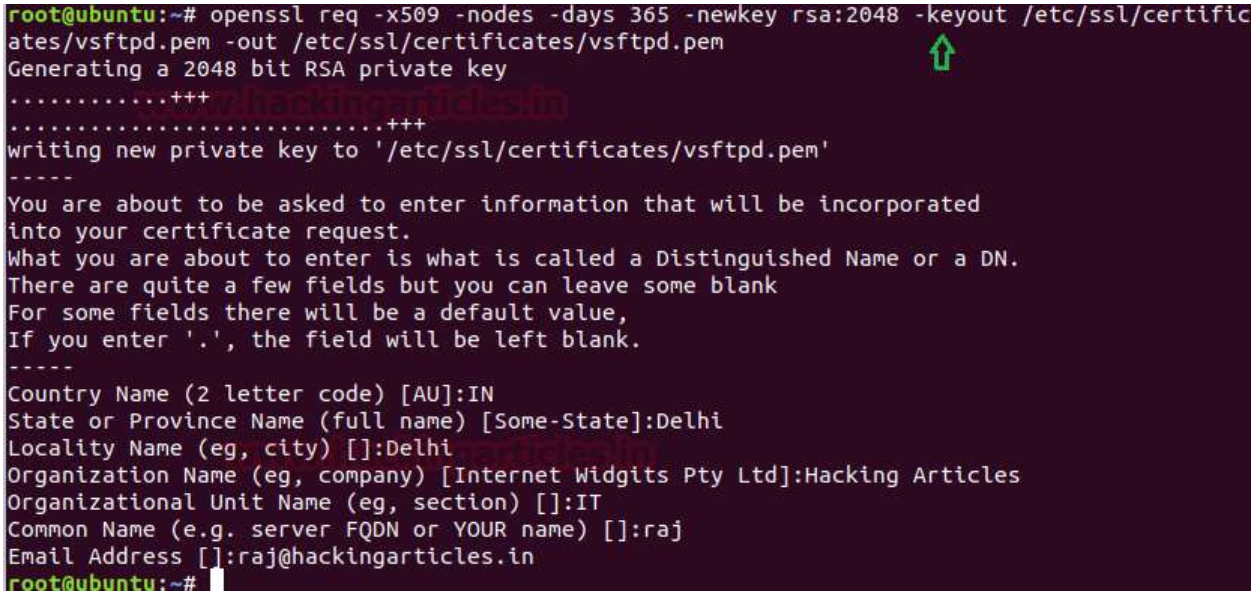
So, let's add another security layer to this for the above-mentioned problem.The solution to this is to create an SSL certificate. This encrypted data packet travels between server and client networks. SSL stands for Secure Sockets Layer, the protocol which provides secure, encrypted communications between server and client.

Although an attacker can sniff network data packet but will be not able to read fetched information because entire data will show in the form of **ciphertext**.

Here, administrations need to generate their own SSL certificate for secure authentication. Make a directory where the SSL certificate keys will be stored.

Use the following command to create a certificate:

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



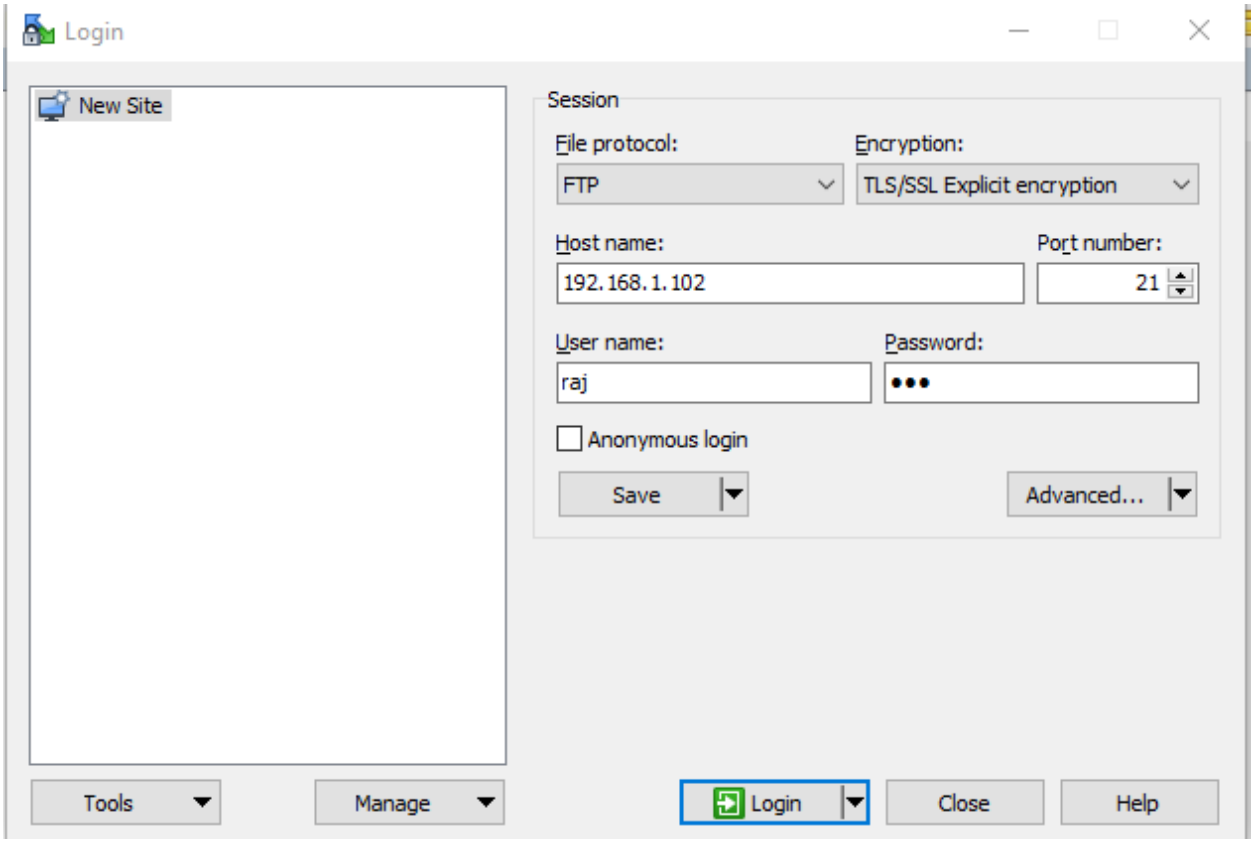
Once the above command is executed, open **the vsftpd.conf** file to change the default settings by adding a few lines at the end of the file. The following are the lines to be added:

rsa_cert_file=/etc/ssl/certificates/vsftpd.pem
rsa_private_key_file=/etc/ssl/certificates/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
require_ssl_reuse=NO
ssl_ciphers=HIGH

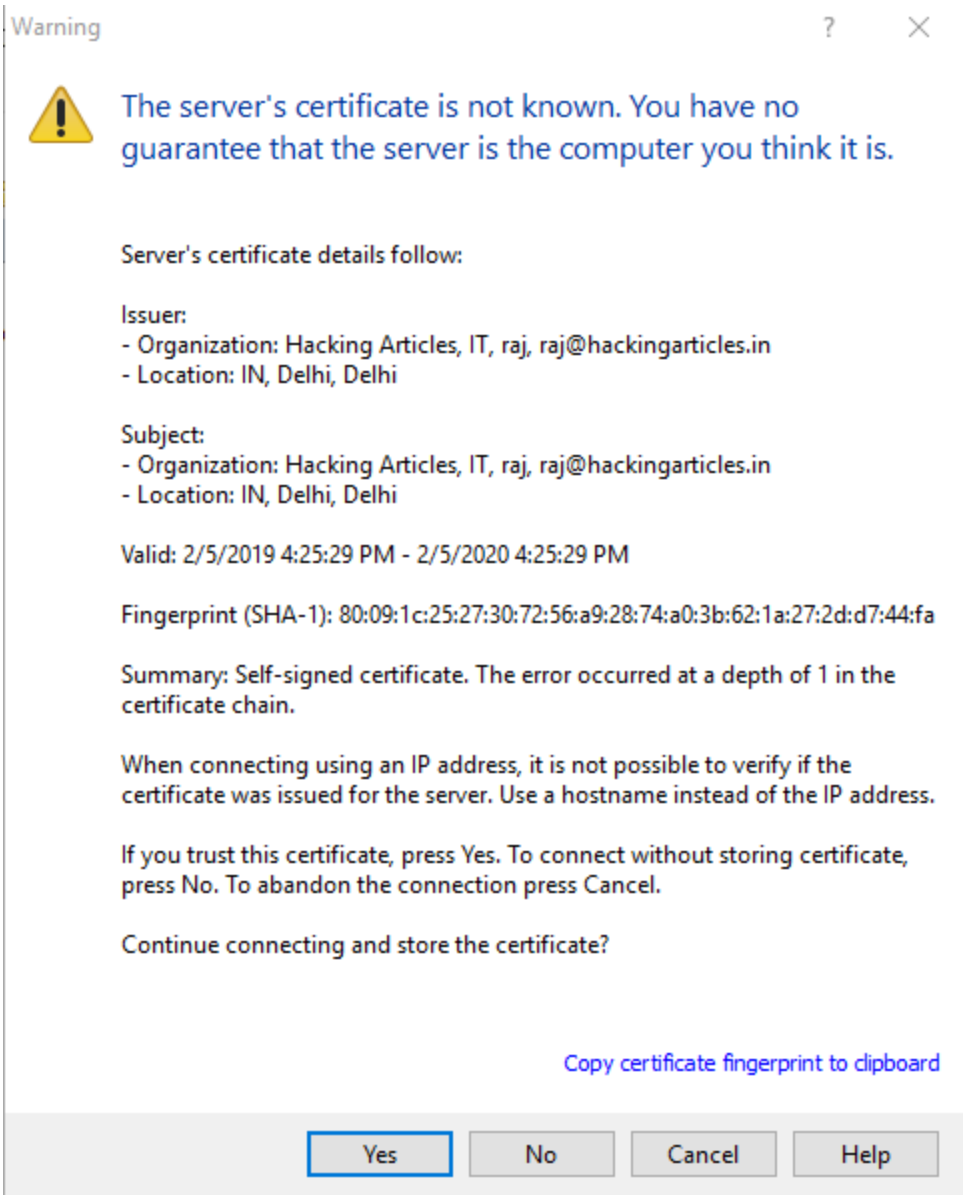
```
rsa_cert_file=/etc/ssl/certs/ssl-cert-snakeoil.pem
rsa_private_key_file=/etc/ssl/private/ssl-cert-snakeoil.key
ssl_enable=NO

#
# Uncomment this to indicate that vsftpd use a utf8 filesystem.
#utf8_filesystem=YES
rsa_cert_file=/etc/ssl/certificates/vsftpd.pem
rsa_private_key_file=/etc/ssl/certificates/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
require_ssl_reuse=NO
ssl_ciphers=HIGH
```

Now let's ensure whether we can connect to FTP server.

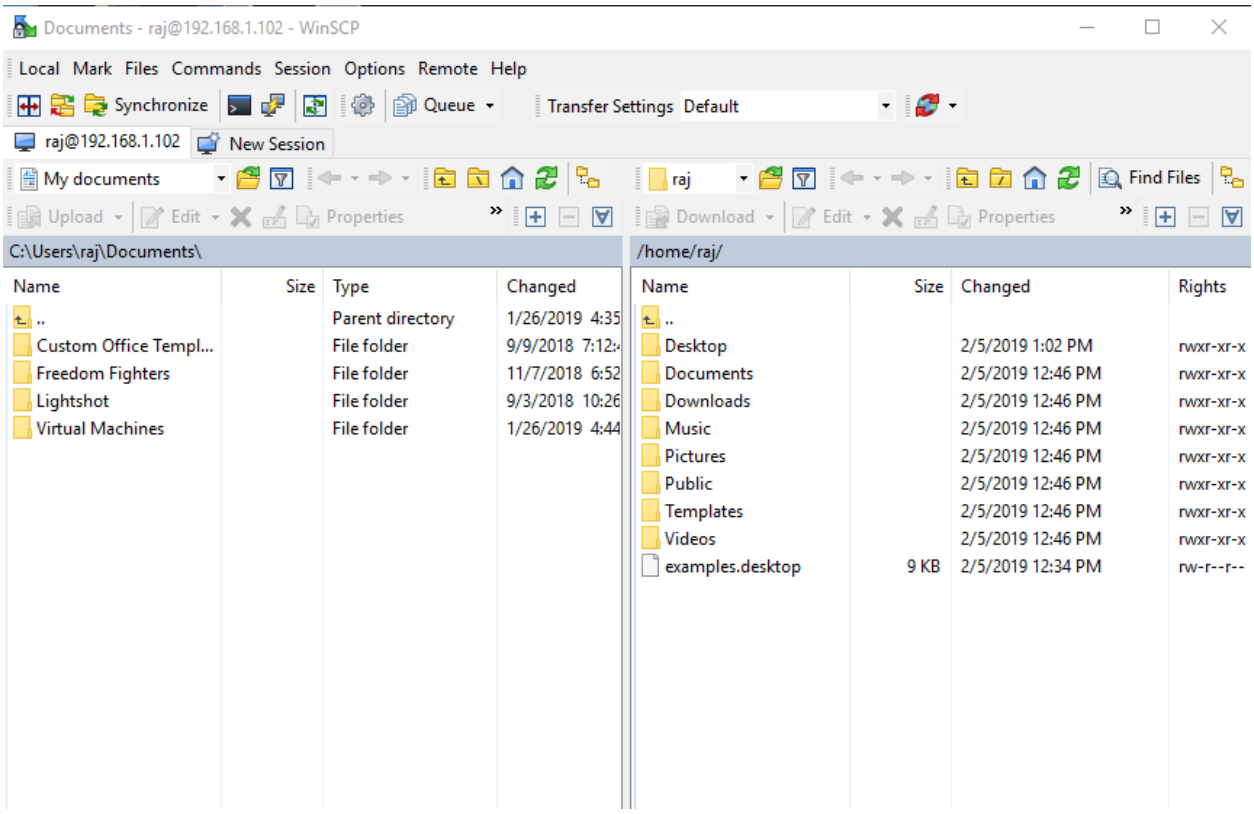
Protocol to: FTP
Encryption To: TSL/SSL Explicit encryption
Hostname: IP of the FTP Server
Port: 21
Username and Password: raj: 123



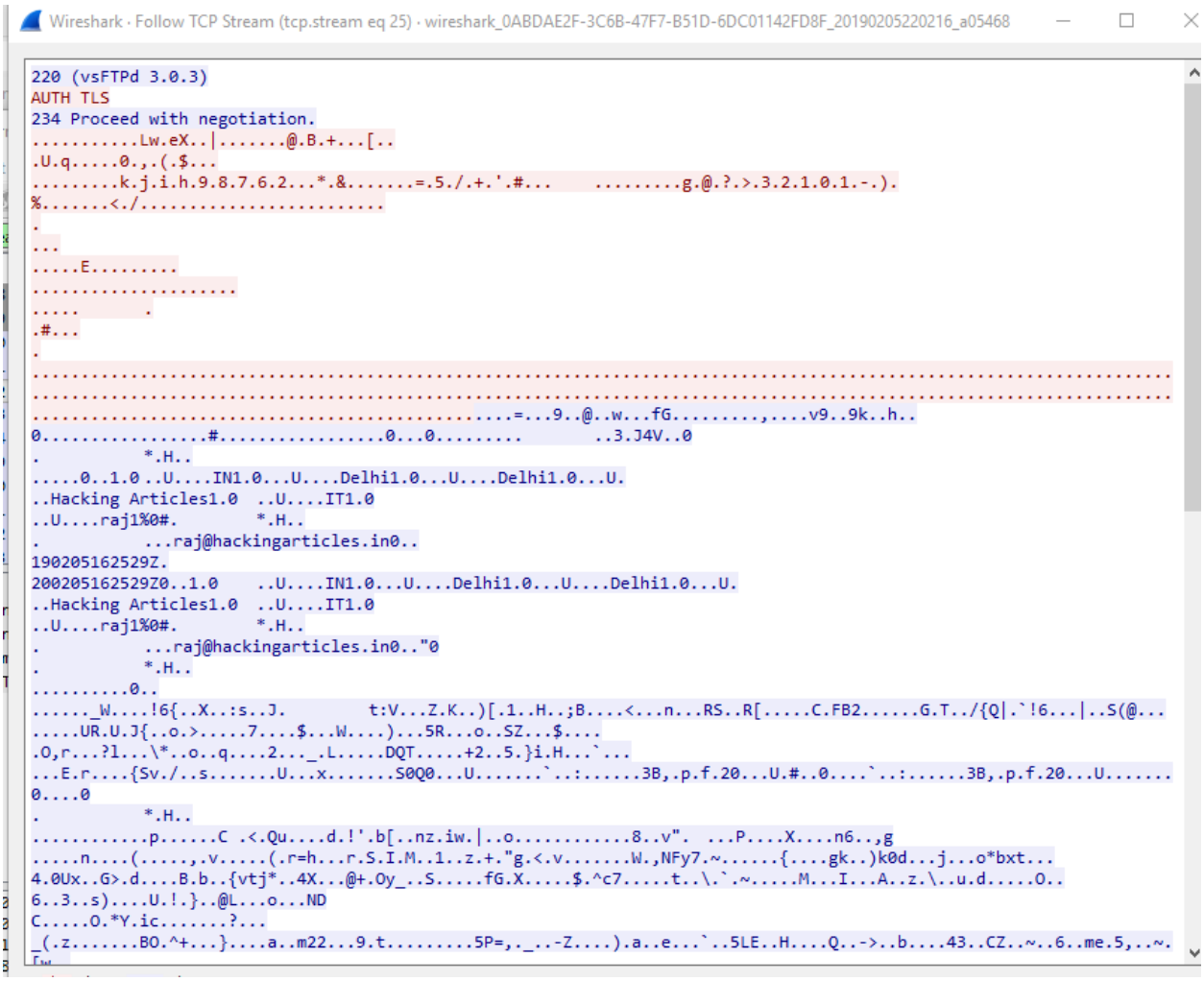
Now the server will send the certificate to an authorized user click on yes to store certificate and continue the encrypted connecting.



Now, when you will establish the connection of FTP as shown in the image below:



All the traffic that is sent and received is encrypted, which you can check through Wireshark. It is also shown below:



Stop FTP Brute_Force Attack with Fail2ban

Hydra is often the tool of choice for bruteforce. It can perform rapid dictionary attacks against more than 50 protocols, including telnet, FTP, HTTP, HTTPS, SMB, several databases, and much more. Now, to bruteforce our FTP server, we need to choose a word list. As with any dictionary attack, the wordlist is key.

Run the following command to execute bruteforce:

```
hydra -L user -P pass 192.168.1.102 ftp
```

```
root@kali:~# hydra -L user -P pass 192.168.1.102 ftp
Hydra v8.8 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizat

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2019-02-05 11:41:02
[DATA] max 16 tasks per 1 server, overall 16 tasks, 36 login tries (l:6/p:6), ~3 tries per task
[DATA] attacking ftp://192.168.1.102:21/
[21][ftp] host: 192.168.1.102   login: raj   password: 123
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2019-02-05 11:41:12
root@kali:~#
```

As you can see, using hydra, we have logged in credentials and so a bruteforce attack is successful. But we can protect our FTP server and important files. To be secure against brute force, you can use the fail2ban tool. For a detailed guide on the fail2ban tool, read our other article **here.**

sudo fail2ban-client status vsftpd

```
root@ubuntu:~# sudo fail2ban-client status vsftpd
Status for the jail: vsftpd
|- Filter
|  |- Currently failed: 0
|  |- Total failed:     16
|  `- File list:        /var/log/vsftpd.log
`- Actions
   |- Currently banned: 1
   |- Total banned:     1
   `- Banned IP list:   192.168.1.109
root@ubuntu:~#
```

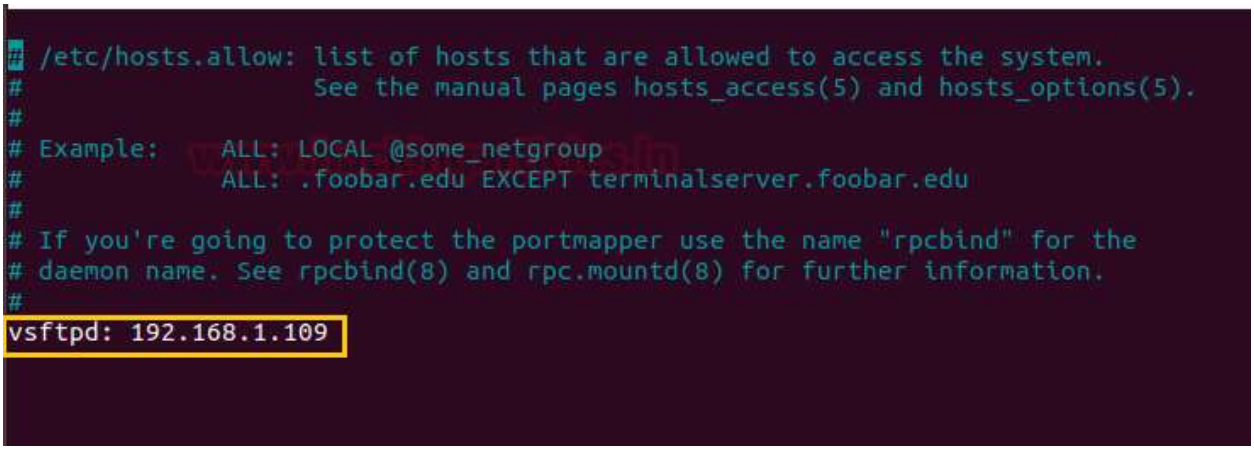
Once you have limited the bruteforce attack through fail2ban, You can try and use hydra again, but you will get a negative result, as shown in the image below:

```
root@kali:~# hydra -L user -P pass 192.168.1.102 ftp
Hydra v8.8 (c) 2019 by van Hauser/THC - Please do not use in military or secret servic

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2019-02-05 11:53:32
[DATA] max 16 tasks per 1 server, overall 16 tasks, 48 login tries (l:8/p:6), ~3 tries
[DATA] attacking ftp://192.168.1.102:21/
```

Restrict IP to connect FTP

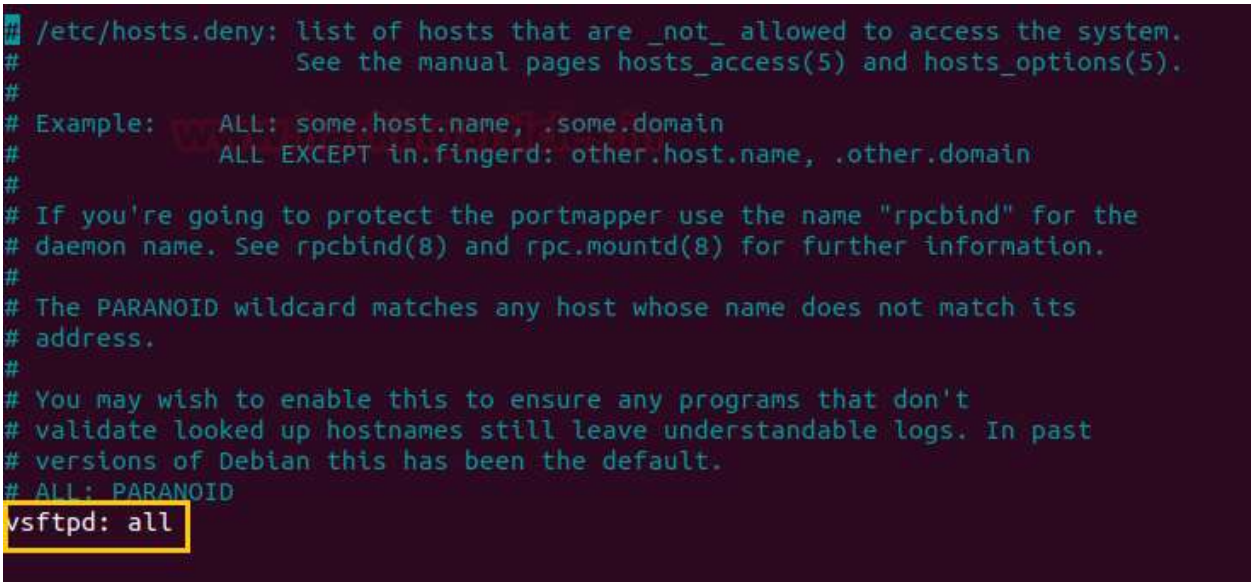
Another security layer that you can apply is blocking all other IPs but allowing your trusted ones. Now open **hosts.allow** file from inside **/etc** to allow the valid user to connect with the server securely through

a specific IP. At the end of the text file, enter the specific IP to whom you want to give permission to establish a connection as shown in the given image.



It is quite important that the admin restrict all IPs other than the allowed IP (192.168.0.106) to protect the network from being connected by an unknown IP.

Open **/etc/hosts.deny** and specify a list of hosts whom you want to deny access to the system.



Now open configure file of vsftpd and add following lines:

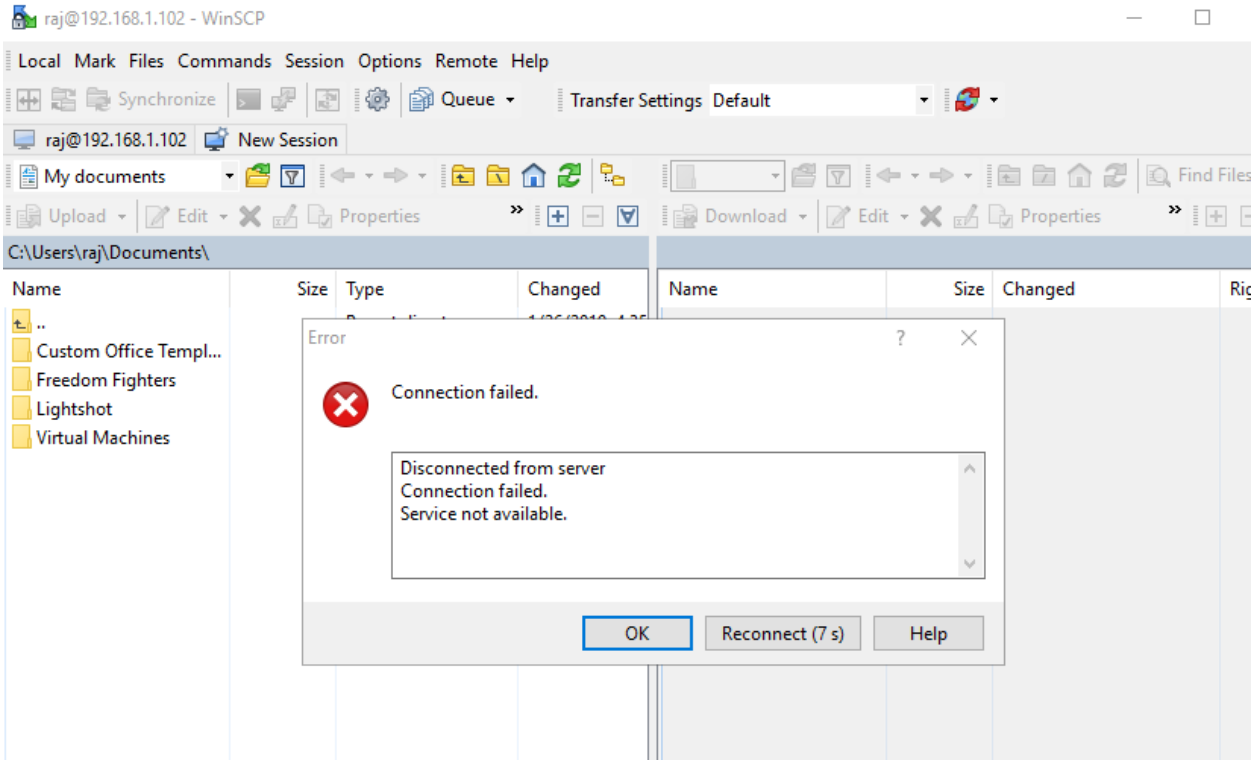
```
# TCP Wrappers
tcp_wrappers=Yes
```

```
secure_chroot_dir=/var/run/vsftpd/empty
#
# This string is the name of the PAM service vsftpd will use.
pam_service_name=vsftpd
#
# This option specifies the location of the RSA certificate to use for SSL
# encrypted connections.
rsa_cert_file=/etc/ssl/certs/ssl-cert-snakeoil.pem
rsa_private_key_file=/etc/ssl/private/ssl-cert-snakeoil.key
ssl_enable=NO

#
# Uncomment this to indicate that vsftpd use a utf8 filesystem.
#utf8_filesystem=YES
# TCP Wrappers
tcp_wrappers=YES
```

Now, if you connect to ftp from an allowed IP address, you will be logged in, as shown in the image below:

```
root@kali:~# ftp 192.168.1.102
Connected to 192.168.1.102.
220 (vsFTPd 3.0.3)
Name (192.168.1.102:root): raj
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
drwxr-xr-x    3 1000     1000         4096 Feb 05 07:32 Desktop
drwxr-xr-x    2 1000     1000         4096 Feb 05 07:16 Documents
drwxr-xr-x    2 1000     1000         4096 Feb 05 07:16 Downloads
drwxr-xr-x    2 1000     1000         4096 Feb 05 07:16 Music
drwxr-xr-x    2 1000     1000         4096 Feb 05 07:16 Pictures
drwxr-xr-x    2 1000     1000         4096 Feb 05 07:16 Public
drwxr-xr-x    2 1000     1000         4096 Feb 05 07:16 Templates
drwxr-xr-x    2 1000     1000         4096 Feb 05 07:16 Videos
-rw-r--r--    1 1000     1000         8980 Feb 05 07:04 examples.desktop
226 Directory send OK.
ftp>
```

But it will block other IPs as shown below:



Conclusion

FTP was discovered around four decades earlier. And since then, there have been substantial changes as it has developed a lot over time. These changes have been related to encryption standards and file transfer functionality.