Bofin Babu | 2013H313085H 17/June/2015

# Lab Demo - Notes

## FTP

```
$ sudo apt-get install vsftpd
```

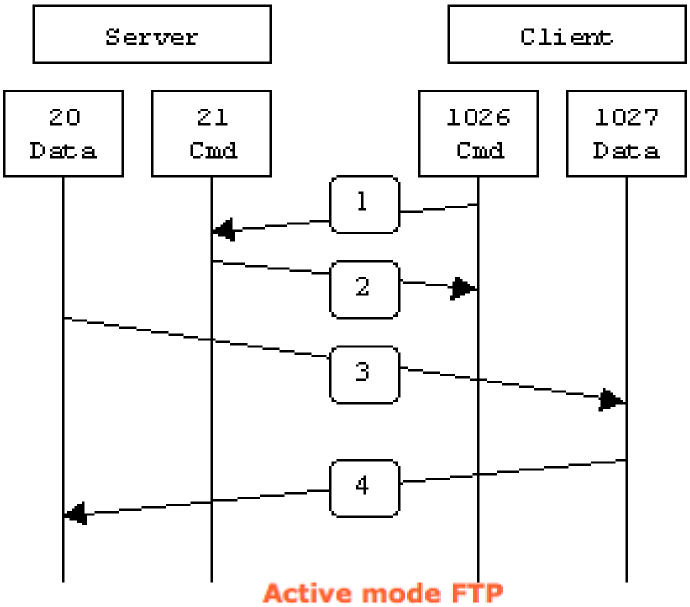
Configuration- /etc/vsftpd.conf
29: write_enable=YES
33: local_umask=022
120: chroot_local_user=YES - access to other folders outside home directory

To enable passive mode, we add:

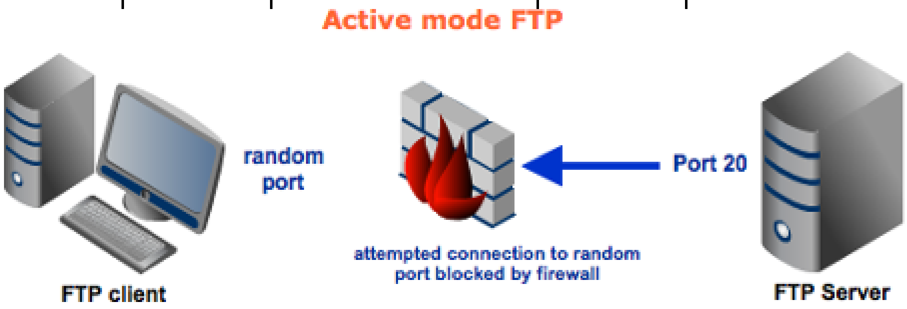
```
pasv_enable=Yes
pasv_min_port=40000
pasv_max_port=40100
```

Passive vs Active FTP

- command channel and data channel



Active FTP

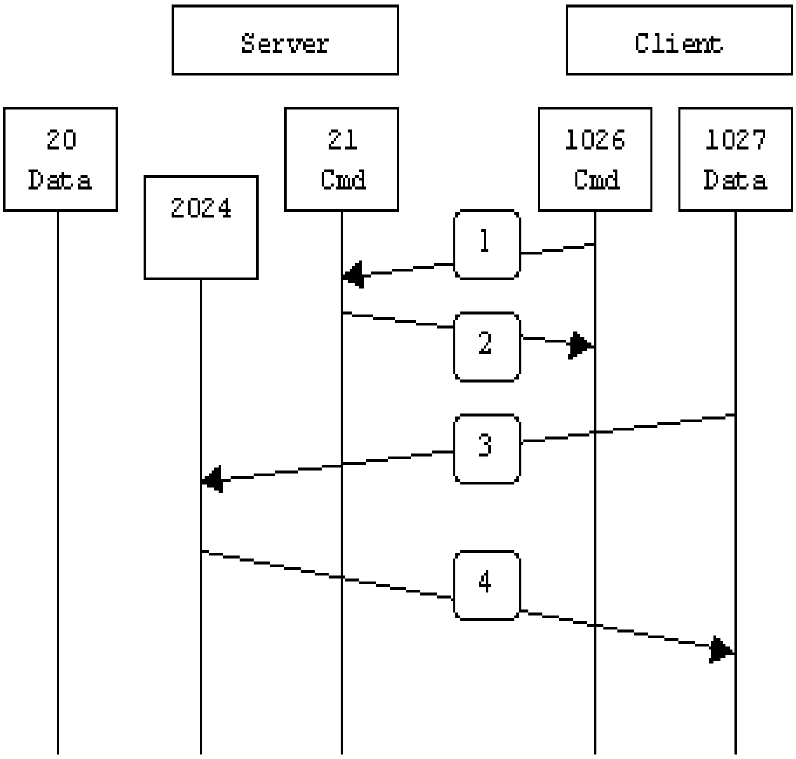


^ FTP server's port 21 from anywhere (Client initiates connection)

- FTP server's port 21 to ports > 1023 (Server responds to client's control port)
- FTP server's port 20 to ports > 1023 (Server initiates data connection to client's

data port)
- FTP server's port 20 from ports > 1023 (Client sends ACKs to server's data port)

```
 //$ sudo service vsftpd restart
// Now ftp server will listen on port 21
```



Passive FTP

^ From the server-side firewall's standpoint, to support passive mode FTP the following communication channels need to be opened:

- FTP server's port 21 from anywhere (Client initiates connection)
- FTP server's port 21 to ports > 1023 (Server responds to client's control port)
- FTP server's ports > 1023 from anywhere (Client initiates data connection to random port specified by server)
- FTP server's ports > 1023 to remote ports > 1023 (Server sends ACKs (and data) to client's data port)

Prevent access to the bash shell for the ftp users.

```
$ sudo useradd -m mohan -s /usr/sbin/nologin
$ sudo passwd mohan
```

Open /etc/shells and add /usr/sbin/nologin

Connect via Filezilla

Secure FTP

- FTP over SSH (using openSSH)

```
$ sudo apt-get install openssh-server
```

Create a new group **ftpaccess** for FTP users.

```
$ sudo groupadd ftpaccess
```

make changes in this **/etc/ssh/sshd_config** file.

Comment out - Subsystem sftp /usr/lib/openssh/sftp-server

Add,

```
Subsystem sftp internal-sftp
Match group ftpaccess
ChrootDirectory %h
X11Forwarding no
AllowTcpForwarding no
ForceCommand internal-sftp
```

```
$ sudo service ssh restart
```

Create user **mohan** with **ftpaccess** group and **/usr/bin/nologin** shell.

```
$ sudo useradd -m smohan -g ftpaccess -s /usr/sbin/nologin
$ sudo passwd smohan
```

$ sudo chown root /home/smohan – change the ownership of home dir.

Create a folder inside home directory for writing and change ownership of that folder.

```
$ sudo mkdir /home/smohan/updir
$ sudo chown mohan:ftpaccess /home/smohan/updir
```

connect server using SFTP ( port : 22 )

Now users can upload files to **updir** directory and cannot access other folders outside home directory

* Ubuntu oracle directory: /opt/oracle or /usr/local/oracle

Then -$ scp -r user@server.ip:/path/to/foo /home/user/Desktop/

or wget -r –no-parent

## MITM attack on FTP

(Victim 1) 172.16.6.185   ---------------/\/\/\/\Kali-Linux/\/\/\\----------------- 172.16.6.120 (Victim 2)

From Kali: Start ping from Victim2 to Victim1, Then
Terminal 1 :  arpspoof -t 172.16.6.120 172.16.6.185 //telling
Terminal 2 :   arpspoof -t 172.16.6.185 172.16.6.120

Now the ping started from Victim2 begins falling.

Start FTP server on victim 1

Go to Victim1 and open/connect to ftp server (via browser or Terminal)

Dsniff needs the entire session for credentials. Log out and complete a session and see the credentials.

Wireshark : tcp.port==21 || tcp.port==20

## REF

(1): http://www.krizna.com/ubuntu/setup-ftp-server-on-ubuntu-14-04-vsftpd/

(2): http://h2-exploitation.blogspot.in/2013/10/configure-pure-ftp-on-kali-linux.html

(3): http://www.windowsecurity.com/articles-tutorials/misc_network_security/Secure_FTP_Server.html

(4): https://www.owasp.org/index.php/Man-in-the-middle_attack

(5): http://www.irongeek.com/i.php?page=security/arpspoof