

Introduction to SSH

SSH (Secure Shell) is a network protocol that provides a secure connection between two computers over an unsecured network. It is used to securely transfer files, execute commands, and manage networks. It is an important tool for system administrators and other users who need to be able to access remote systems securely.

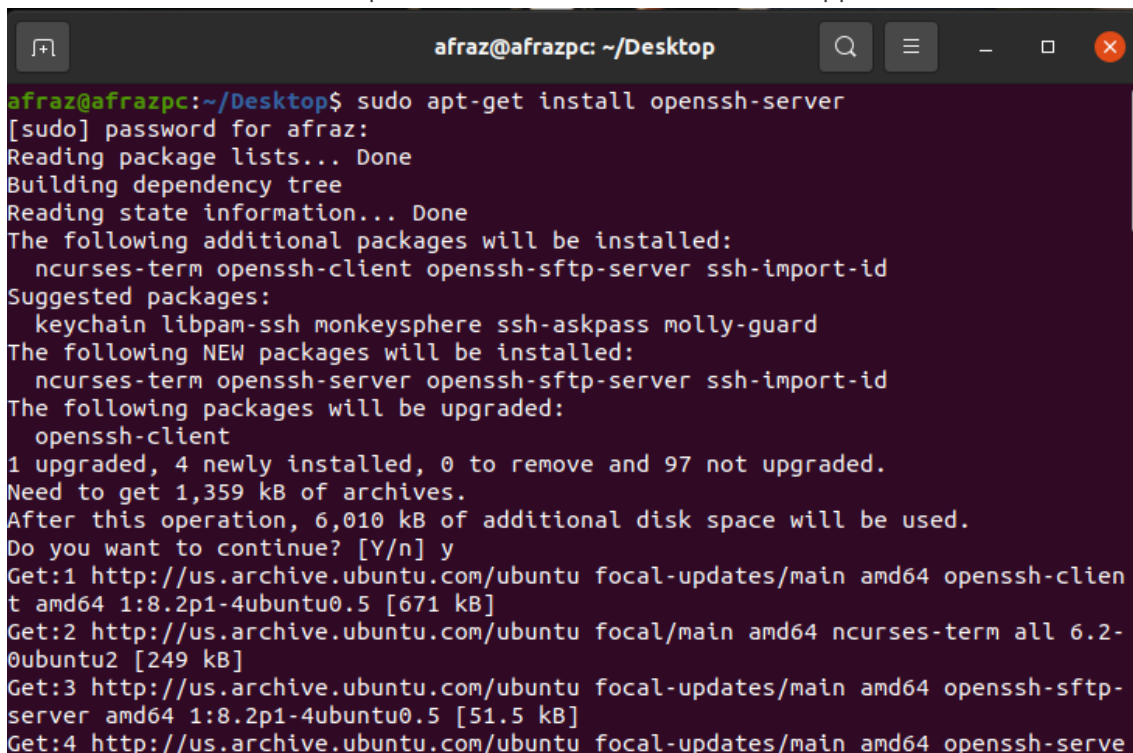
This is an important concept in cloud and server communication, because while deploying a website maybe our server is in some remote location. So the USB model fails to fulfill our purpose.

How is it accomplished between Windows and Linux?

It is possible to access a linux machine from windows, and vice versa. But first, some configuration needs to be done on the windows and linux machines.

Part 1 - Configuring the Linux Machine:

1. Run the command **sudo apt-get install openssh-server** on the terminal of your linux machine. This will install the openssh server. A screen like this will appear.

A terminal window titled 'afraz@afrazpc: ~/Desktop' with search, menu, and window control icons in the title bar. The terminal shows the command 'sudo apt-get install openssh-server' being executed. It prompts for a password, then shows the package lists, dependency tree, and state information. It lists additional packages to be installed (ncurses-term, openssh-client, openssh-sftp-server, ssh-import-id), suggested packages (keychain, libpam-ssh, monkeysphere, ssh-askpass, molly-guard), and NEW packages to be installed (ncurses-term, openssh-server, openssh-sftp-server, ssh-import-id). It also shows the packages to be upgraded (openssh-client) and the disk space requirements. Finally, it asks for confirmation to continue, which is answered with 'y'. The terminal then shows the progress of downloading the packages from the Ubuntu repository.

```
afraz@afrazpc:~/Desktop$ sudo apt-get install openssh-server
[sudo] password for afraz:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  ncurses-term openssh-client openssh-sftp-server ssh-import-id
Suggested packages:
  keychain libpam-ssh monkeysphere ssh-askpass molly-guard
The following NEW packages will be installed:
  ncurses-term openssh-server openssh-sftp-server ssh-import-id
The following packages will be upgraded:
  openssh-client
1 upgraded, 4 newly installed, 0 to remove and 97 not upgraded.
Need to get 1,359 kB of archives.
After this operation, 6,010 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 openssh-clien
t amd64 1:8.2p1-4ubuntu0.5 [671 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu focal/main amd64 ncurses-term all 6.2-
0ubuntu2 [249 kB]
Get:3 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 openssh-sftp-
server amd64 1:8.2p1-4ubuntu0.5 [51.5 kB]
Get:4 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 openssh-serve
```

2. Now, when it is done, run this command **sudo apt install net-tools**. This is because by default, linux is an extremely secure operating system and we can not get the IP address of the machine normally. A screen like this will appear.

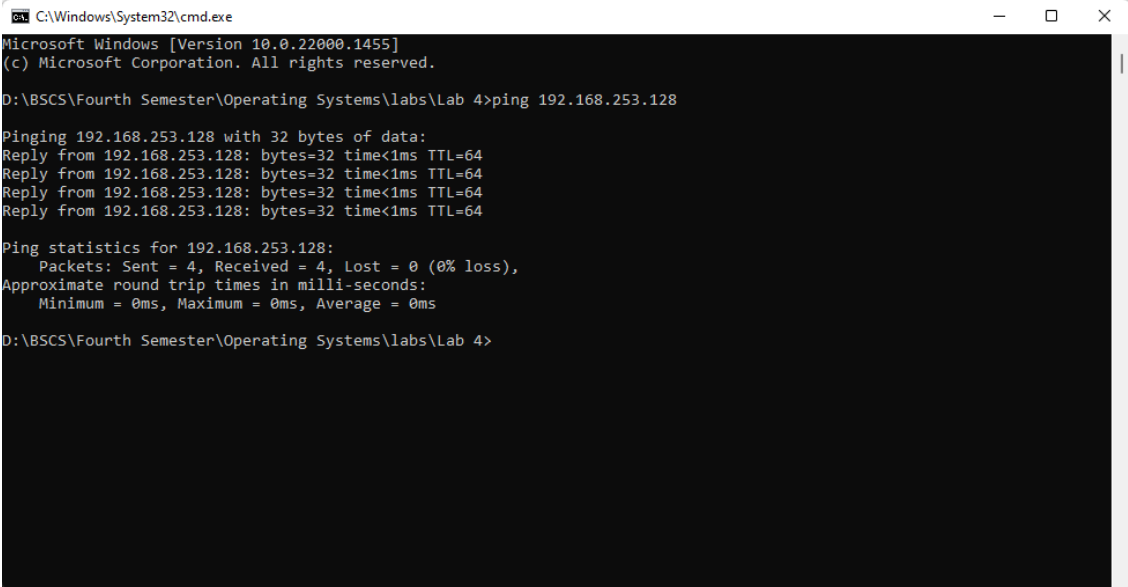
```
afraz@afrazpc:~/Desktop$ sudo apt install net-tools
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  net-tools
0 upgraded, 1 newly installed, 0 to remove and 97 not upgraded.
Need to get 196 kB of archives.
After this operation, 864 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu focal/main amd64 net-tools amd64 1.60+
git20180626.aebd88e-1ubuntu1 [196 kB]
Fetched 196 kB in 3s (70.8 kB/s)
Selecting previously unselected package net-tools.
(Reading database ... 186491 files and directories currently installed.)
Preparing to unpack .../net-tools_1.60+git20180626.aebd88e-1ubuntu1_amd64.deb ..
.
Unpacking net-tools (1.60+git20180626.aebd88e-1ubuntu1) ...
Setting up net-tools (1.60+git20180626.aebd88e-1ubuntu1) ...
Processing triggers for man-db (2.9.1-1) ...
```

3. Now, we have all the configuration on the linux machine. We only need to get the IP address now. For this purpose, run the **ifconfig** command. What it does is that, it churns out the IP address for clusters as well as individual users too (*recall the concept of groups and users in linux*). The highlighted part is the IP of our linux machine.

```
afraz@afrazpc:~/Desktop$ ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.253.128 netmask 255.255.255.0 broadcast 192.168.253.255
    inet6 fe80::dfbf:b4d4:2a00:7ab5 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:fb:f1:61 txqueuelen 1000 (Ethernet)
    RX packets 1904 bytes 2104488 (2.1 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1034 bytes 141819 (141.8 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 371 bytes 37320 (37.3 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 371 bytes 37320 (37.3 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

You can also check if the IP exists from windows too by the ping command.



```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.22000.1455]
(c) Microsoft Corporation. All rights reserved.

D:\BSCS\Fourth Semester\Operating Systems\labs\Lab 4>ping 192.168.253.128

Pinging 192.168.253.128 with 32 bytes of data:
Reply from 192.168.253.128: bytes=32 time<1ms TTL=64
Reply from 192.168.253.128: bytes=32 time<1ms TTL=64
Reply from 192.168.253.128: bytes=32 time<1ms TTL=64
Reply from 192.168.253.128: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.253.128:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

D:\BSCS\Fourth Semester\Operating Systems\labs\Lab 4>
```

This shows that the IP exists.

Part 2: Configuring the Windows machine:

Now here is where things get a little bit tricky. There is an important thing to keep in mind from here.

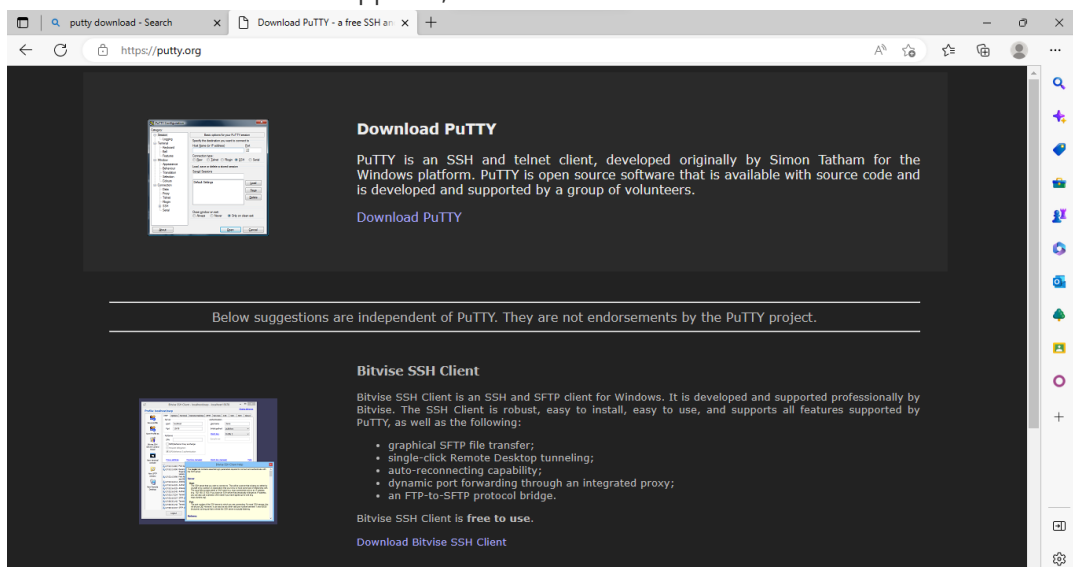
1) windows uses back-slash for file separation e.g "C:\\\\so on. Linux uses forward slash e.g home/user/so on".

Let's start configuring the windows machine.

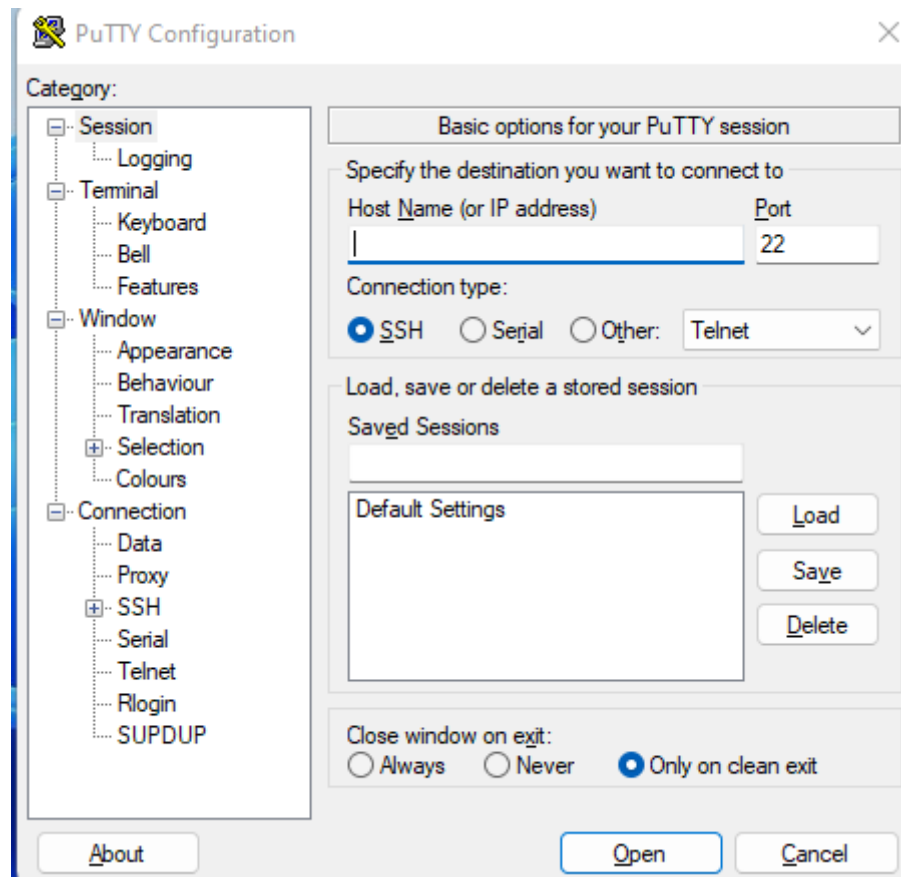
Downloading PuTTY

PuTTY is a free and open source terminal emulator, serial console, and network file transfer application. It is primarily used to connect to remote computers and manage networks, systems, and configurations. PuTTY supports a wide variety of protocols, including SSH, telnet, rlogin, and raw TCP protocols. It can also be used to securely transfer files using SCP and SFTP.

1. Go to this website [Download PuTTY - a free SSH and telnet client for Windows](https://putty.org).
2. Download the first link that appears, and install it.



3. After it is installed, open it.
4. You will see this kind of window.



Here, the Host_Name is the IP of the linux machine we found in step 3 of Part 1. Put that here. The port number (*specified here as 22*) is also important because it tells our machine where to connect with the other machine. (*This concept is analogous to connecting two computers with a wire*)

5. Once You press open, the putty console will ask for your linux username and password. Enter them, and *voila*. You have entered your linux prompt from within your windows. You can check for your directories using the ls command.

```
afraz@afrazpc: ~
login as: afraz
afraz@192.168.253.128's password:
Welcome to Ubuntu 20.04.4 LTS (GNU/Linux 5.15.0-58-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

97 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Your Hardware Enablement Stack (HWE) is supported until April 2025.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

afraz@afrazpc:~$ ls
Codes    Documents  Music      Pictures  snap      Videos
Desktop  Downloads  OperatingSystemsLab  Public   Templates
afraz@afrazpc:~$
```

Part 3 - Transferring Data

3-1: From Windows to Linux

Open up the location where PuTTY is installed. In Windows, this will most probably be in the Program Files Folder. Open up the cmd window from there and type this command

pscp -P 22 "FromWindowsDirectory" "ToLinuxDirectory"

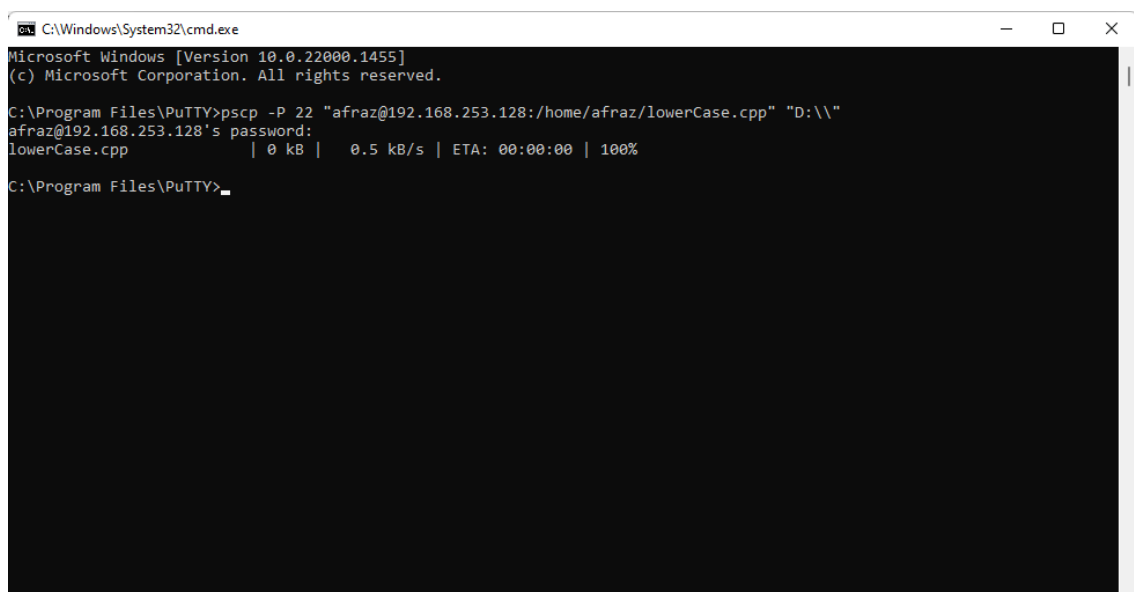
Keep in mind the rule we discussed earlier in part 2 about back and forward slashes. The working of this example is discussed below, the syntax and methodology is exactly the same.

3-2: From Linux to Windows

Open up the location where PuTTY is installed. In Windows, this will most probably be in the Program Files Folder. Open up the cmd window from there and type this command

pscp -P 22 "FromLinuxDirectory" "ToWindowsDirectory"

Keep in mind the rule we discussed earlier in part 2 about back and forward slashes.

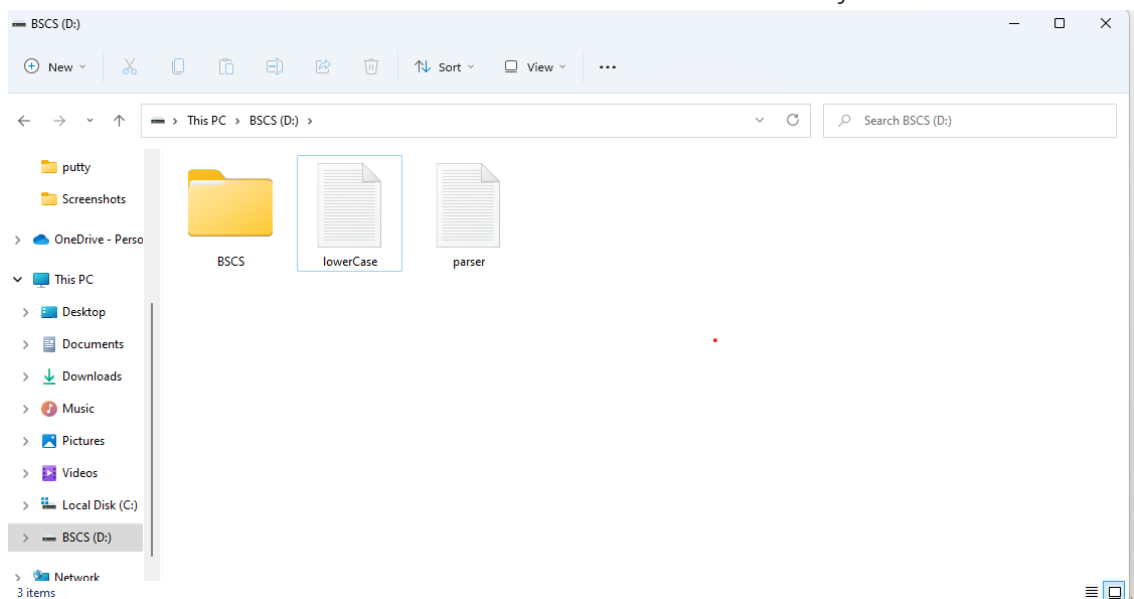


```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.22000.1455]
(c) Microsoft Corporation. All rights reserved.

C:\Program Files\PuTTY>pscp -P 22 "afraz@192.168.253.128:/home/afraz/lowerCase.cpp" "D:\""
afraz@192.168.253.128's password:
lowerCase.cpp          | 0 kB |   0.5 kB/s | ETA: 00:00:00 | 100%

C:\Program Files\PuTTY>
```

We can see that the file was transferred to windows in the D directory.



Important:

Keep the syntax of the command, exactly the same as it appears in the screenshot above.
/home/user is the linux directory that is default open to ssh transfers. And, the separator in windows is \.