

## Part 1: Connect to BC's Web Lab Linux Server using SSH.

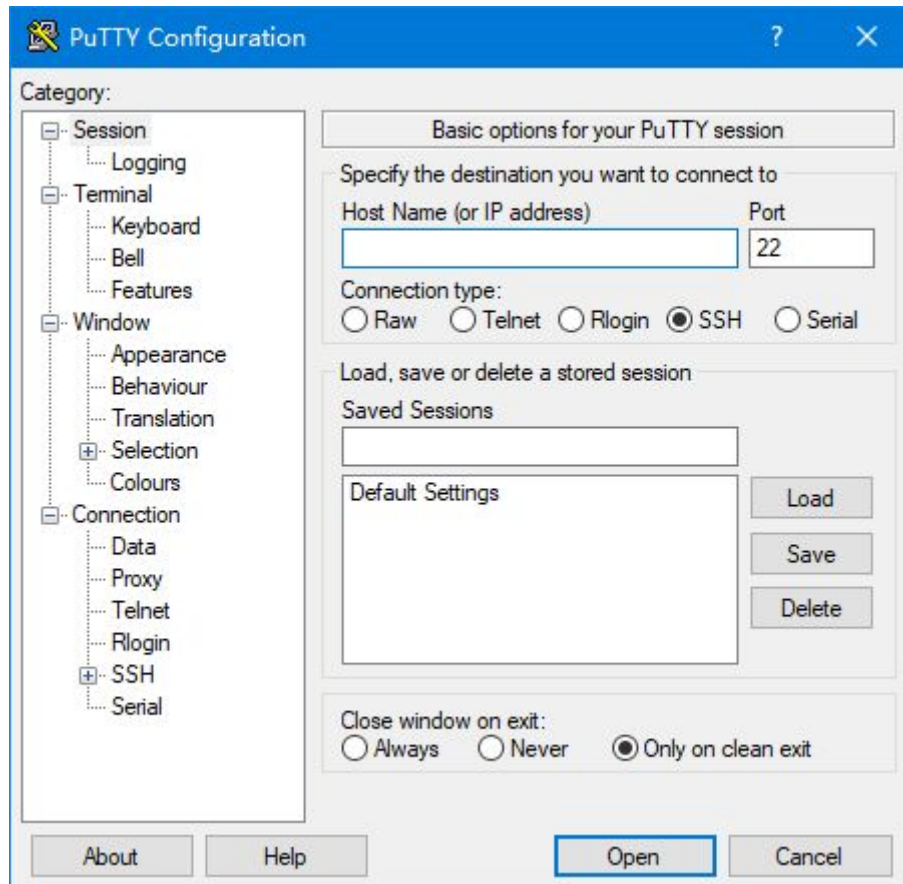
Secure Shell (SSH) is a UNIX-based command interface and protocol that is used by nowadays people for securely getting access to a remote computer. SSH is actually a suite of three utilities - slogin, ssh, and scp. slogin is short for secure login and is a Linux and Unix secure shell SSH (secure shell) remote login program. For slogin to work, you must log into a machine that supports a secure login and that in most cases, a computer or Internet service provider has multiple login addresses. While scp is short for secure copy. It is a command-line utility that allows you to securely copy files and directories between two locations. With the usage of scp, you can copy a file or directory in three ways: from your local system to a remote system, from a remote system to your local system, and between two remote systems from your local system. When transferring data with scp, both the files and password are encrypted so that anyone snooping on the traffic doesn't get anything sensitive information. SSH commands are encrypted and secure in several ways. Both ends of the client and the server connection are authenticated using a digital certificate, and passwords are protected by being encrypted. In short, SSH allows you to connect to your server securely to perform Linux command-line operations. And although you will be running Linux commands, SSH can still be performed from a Windows, Mac, or Linux computer.

Today I would be using PuTTY from Window to make an SSH connection to the BC's Web Lab Linux server.

Before starting, there would be three things that we need to make an SSH connection to a server, that is:

- Server IP Address
- SSH Username
- SSH Password

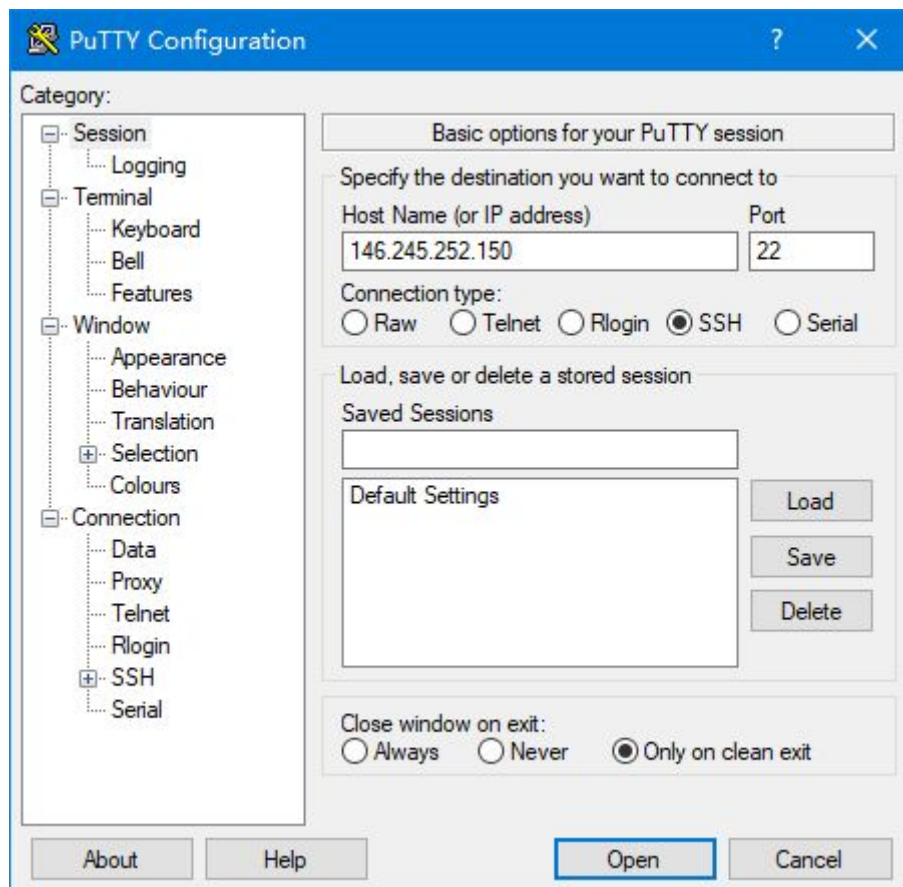
Then download PuTTY and launch the program. It would lead you to this page:



This is the main page of the PuTTY program. The information we needed in this page would be the Host Name/IP address of the server that we are connecting to. Also, we have to make sure that the connection type is SSH. The IP address that we can use to connect to the Brooklyn College WEB Lab Linux server are:

- 146.245.252.25
- 146.245.252.26
- 146.245.252.28
- 146.245.252.29
- 146.245.252.30
- 146.245.252.31
- 146.245.252.150

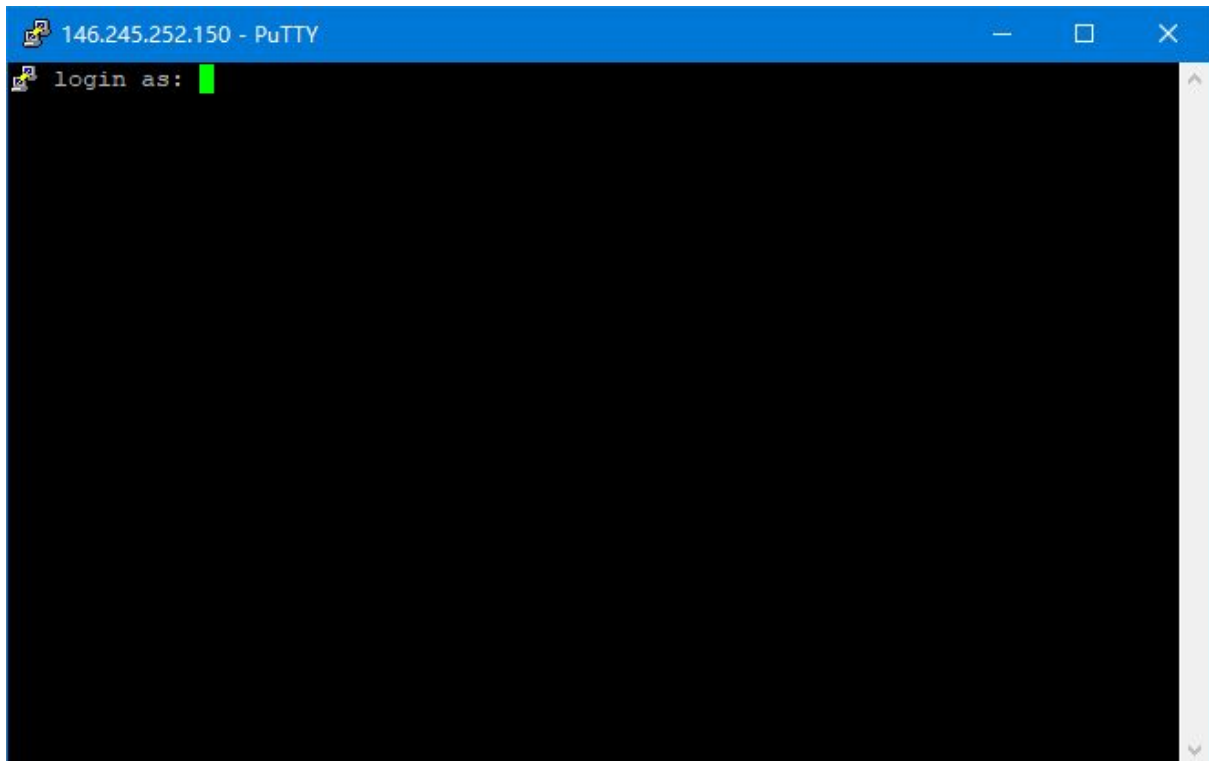
Since either one would be fine, I just chose the last IP address and input it into the program. The port would be set as default to 22 and also the connection type would be set to default of SSH.



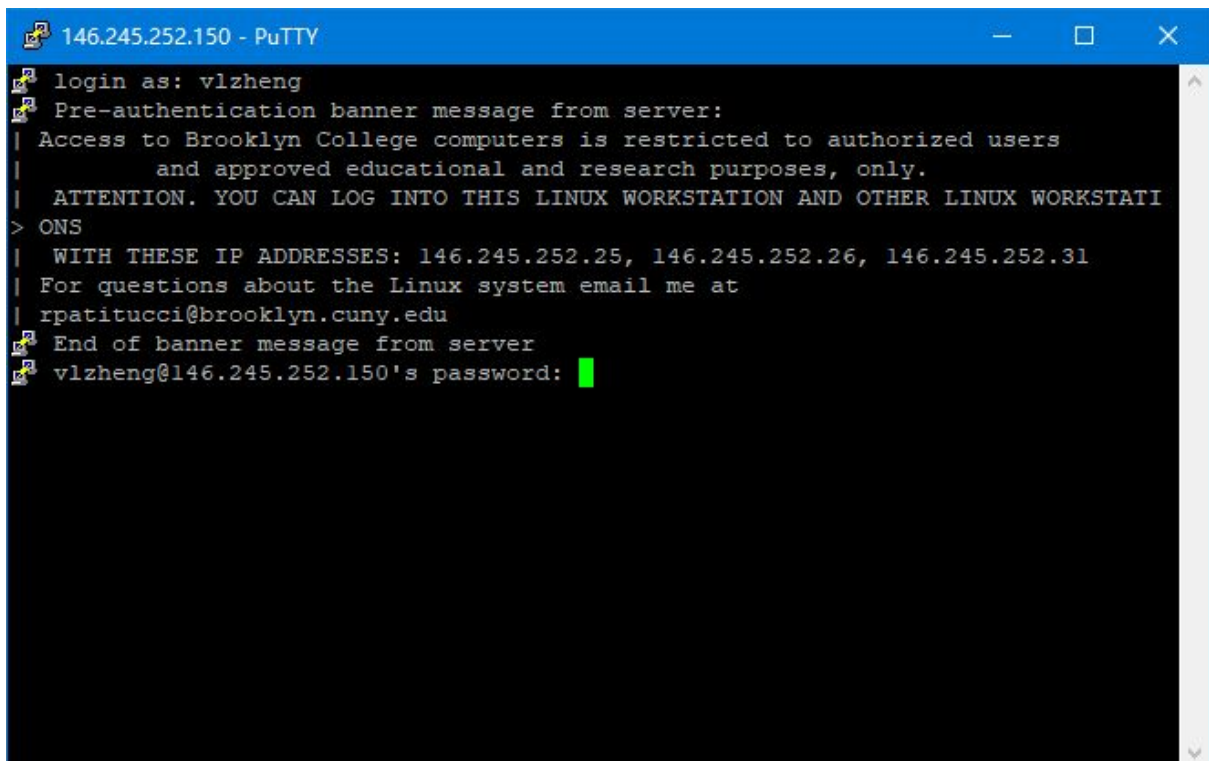
Click open to start the SSH session. If this is your first time connecting to the server from this computer, you will see the following output. Just simply accept the connection by clicking Yes.

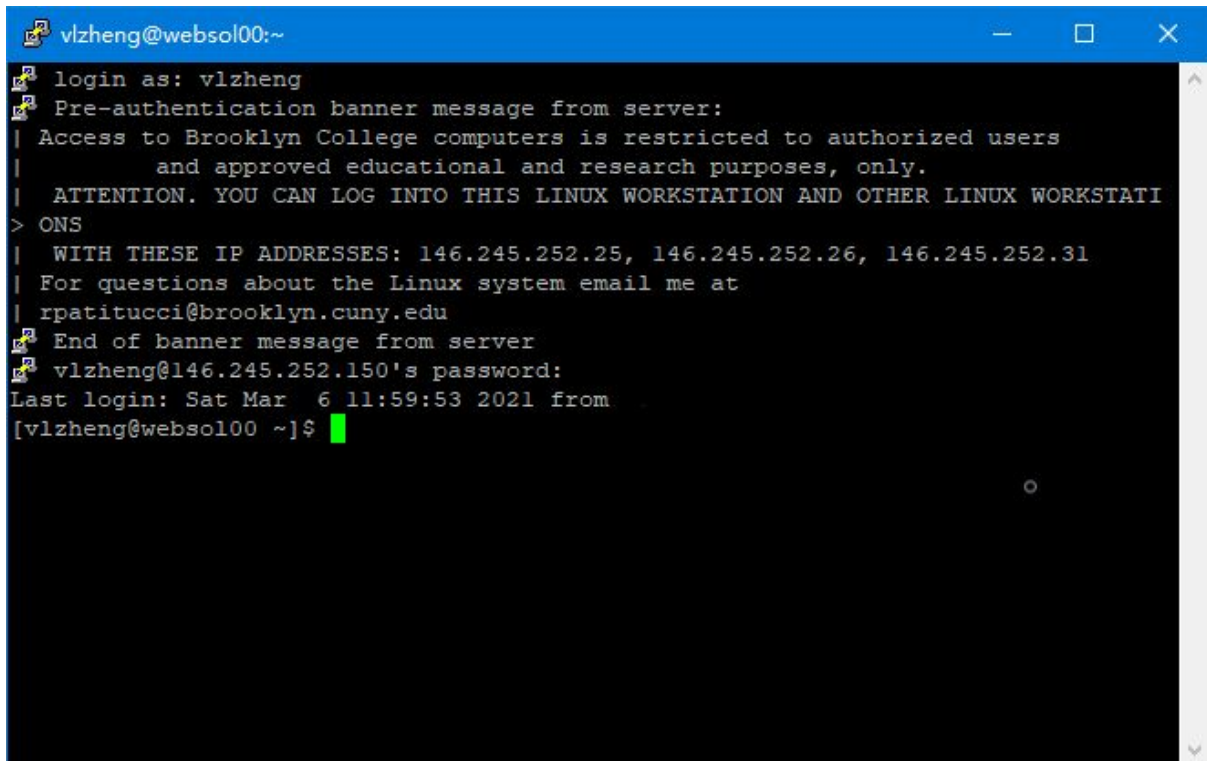


Once the SSH Connection is open, you should see a terminal prompt asking for your username and would look like this:



In this case type your login username. After inputting your username and click enter, it would prompt and ask for your password. Enter the password and click enter.



A terminal window titled 'vlzheng@websol00:~' showing the process of logging into a Linux server via SSH. The output includes a pre-authentication banner message from the server, which states that access to Brooklyn College computers is restricted to authorized users and approved educational and research purposes. The banner also provides IP addresses (146.245.252.25, 146.245.252.26, 146.245.252.31) and an email address (rpatitucci@brooklyn.cuny.edu) for questions. The user 'vlzheng' enters their password, and the system shows the last login time as 'Sat Mar 6 11:59:53 2021 from'. The prompt returns to '[vlzheng@websol00 ~]\$' with a green cursor.

```
login as: vlzheng
Pre-authentication banner message from server:
| Access to Brooklyn College computers is restricted to authorized users
|   and approved educational and research purposes, only.
| ATTENTION. YOU CAN LOG INTO THIS LINUX WORKSTATION AND OTHER LINUX WORKSTATI
> ONS
| WITH THESE IP ADDRESSES: 146.245.252.25, 146.245.252.26, 146.245.252.31
| For questions about the Linux system email me at
| rpatitucci@brooklyn.cuny.edu
End of banner message from server
vlzheng@146.245.252.150's password:
Last login: Sat Mar  6 11:59:53 2021 from
[vlzheng@websol00 ~]$
```

This is the page showing that you were successfully connected to the Linux Server by using SSH.

Part 2: Lookup information for linux programs and summarize the information.

Examples of commandas used in Linux:

whoami - To see which user you are using.

```
[vlzheng@websol00 ~]$ whoami
vlzheng
```

whatis - To get a one-line manual page descriptions.

```
[vlzheng@websol00 ~]$ whatis wget
wget (1) - The non-interactive network downloader.
```

whereis - To find the location of source/binary file of a command and manuals sections for a specified file in Linux system.

```
[vlzheng@websol00 ~]$ whereis git
git: /usr/bin/git /usr/share/man/man1/git.1.gz
```

man - To view a system's reference manuals

```
[vlzheng@websol00 ~]$ man git
```

```
vlzheng@websol00:~  
GIT(1) Git Manual GIT(1)  
NAME  
    git - the stupid content tracker  
SYNOPSIS  
    git [--version] [--help] [-c <name>=<value>]  
        [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]  
        [-p|--paginate|--no-pager] [--no-replace-objects] [--bare]  
        [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]  
        <command> [<args>]  
DESCRIPTION  
    Git is a fast, scalable, distributed revision control system with an unusually rich  
    command set that provides both high-level operations and full access to internals.  
  
    See gittutorial(7) to get started, then see Everyday Git[1] for a useful minimum set  
    of commands. The Git User's Manual[2] has a more in-depth introduction.  
  
    After you mastered the basic concepts, you can come back to this page to learn what  
    commands Git offers. You can learn more about individual Git commands with "git help  
    command". gitcli(7) manual page gives you an overview of the command line command  
    syntax.  
  
    Formatted and hyperlinked version of the latest Git documentation can be viewed at  
    http://git-htmldocs.googlecode.com/git/git.html.  
OPTIONS  
    --version  
        Prints the Git suite version that the git program came from.  
Manual page git(1) line 1 (press h for help or q to quit)
```

pwd - Show current directory

```
[vlzheng@websol00 ~]$ pwd  
/users1/st/vlzheng
```

application\_name -version - To check the version of the program

```
[vlzheng@websol00 ~]$ java -version  
java version "1.8.0_171"  
Java(TM) SE Runtime Environment (build 1.8.0_171-b11)  
Java HotSpot(TM) 64-Bit Server VM (build 25.171-b11, mixed mode)
```

id - Displays the details of the active user e.g. uid, gid, and groups

```
[vlzheng@websol00 ~]$ id  
uid=23807(vlzheng) gid=400(cis) groups=400(cis) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

clear - Clears the terminal

