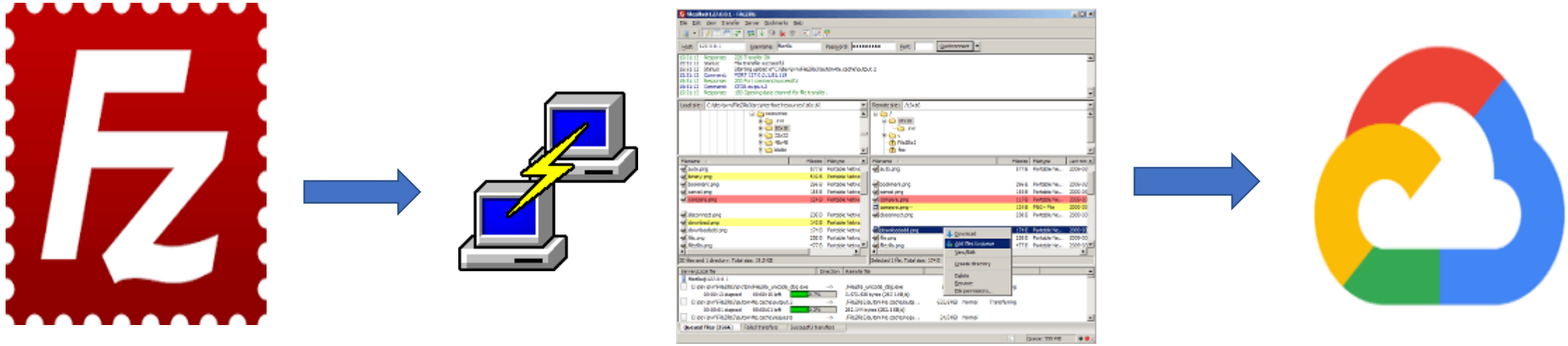
Step-by-step Filezilla and PuTTY installation to securely connect to Linux VM on the Google Cloud



Google Cloud Tutorial by Michael Weiss

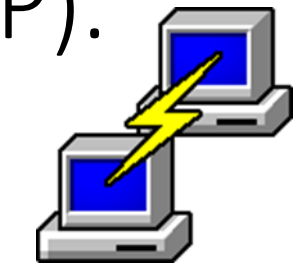
What is Filezilla?

Suppose you have a created a website on Linux Apache and you want to upload that web content to a web server. One way to do this is to use an FTP application. Filezilla is an FTP front-end that helps you do this. Here we use **Filezilla client**.

- FileZilla is a free software, cross-platform FTP application, consisting of FileZilla Client and FileZilla Server. *We will use the client*.
- Client binaries are available for Windows, Linux, and macOS, server binaries are available for Windows only.
- Both server and client support FTP and FTPS (FTP over SSL/TLS), while the client can in addition connect to SFTP servers.

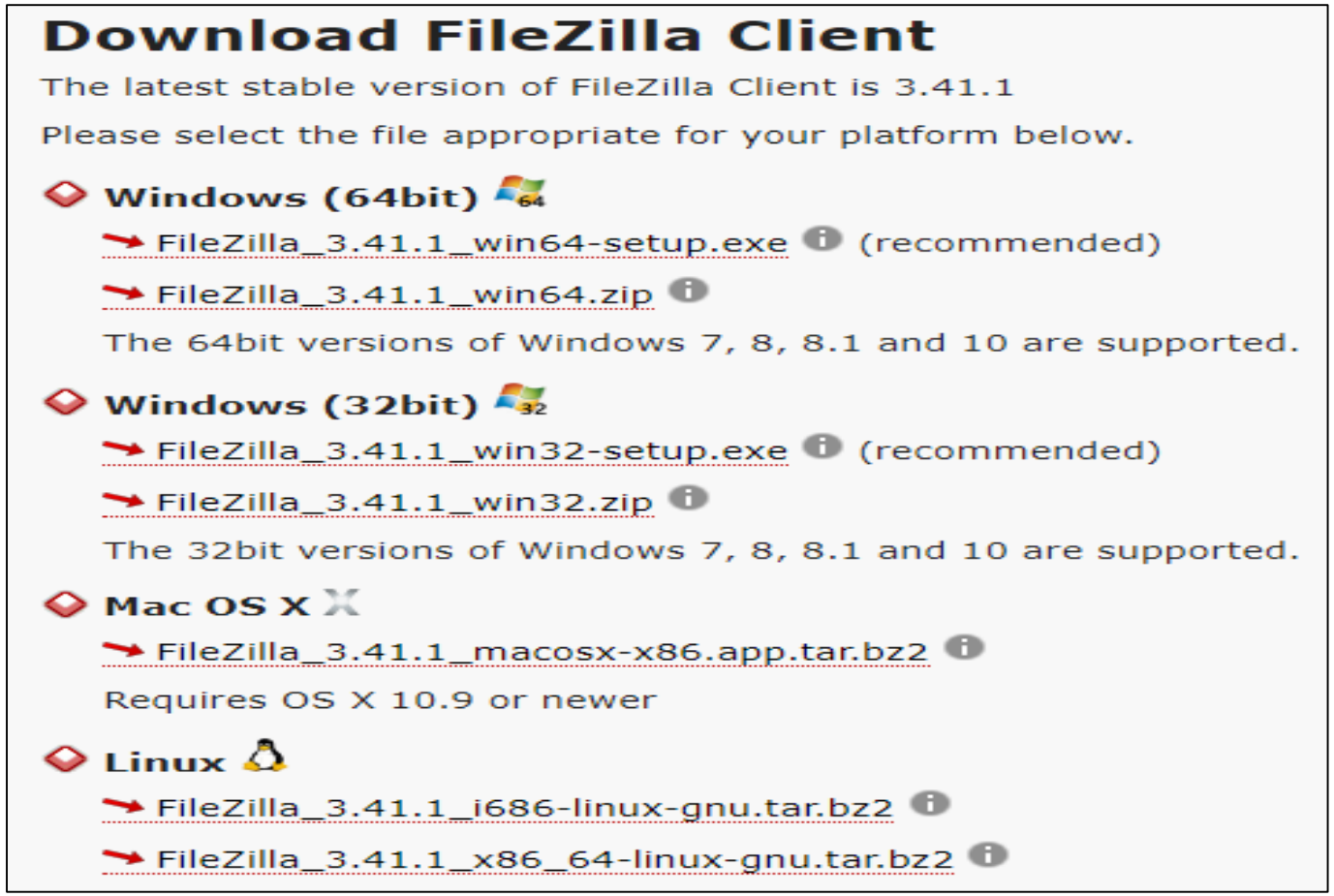
What is PuTTY?

PuTTY is an application that is used to help secure network file transfers.
We will use it secure our FTP file transfers (SFTP).

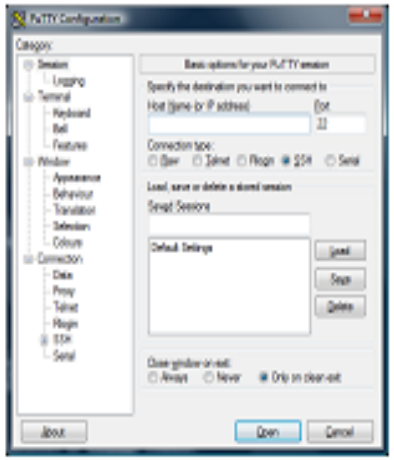


- PuTTY is a free and open-source terminal emulator, serial console and network file transfer application.
- It supports several network protocols, including SCP, SSH, Telnet, rlogin, and raw socket connection.
- It can also connect to a serial port.
- The name "PuTTY" has no official meaning.

The first step is to download Filezilla: https://filezilla-project.org/download.php?show_all=1



The next step is to download and install the PuTTY SSH key generator: **https://www.putty.org/**

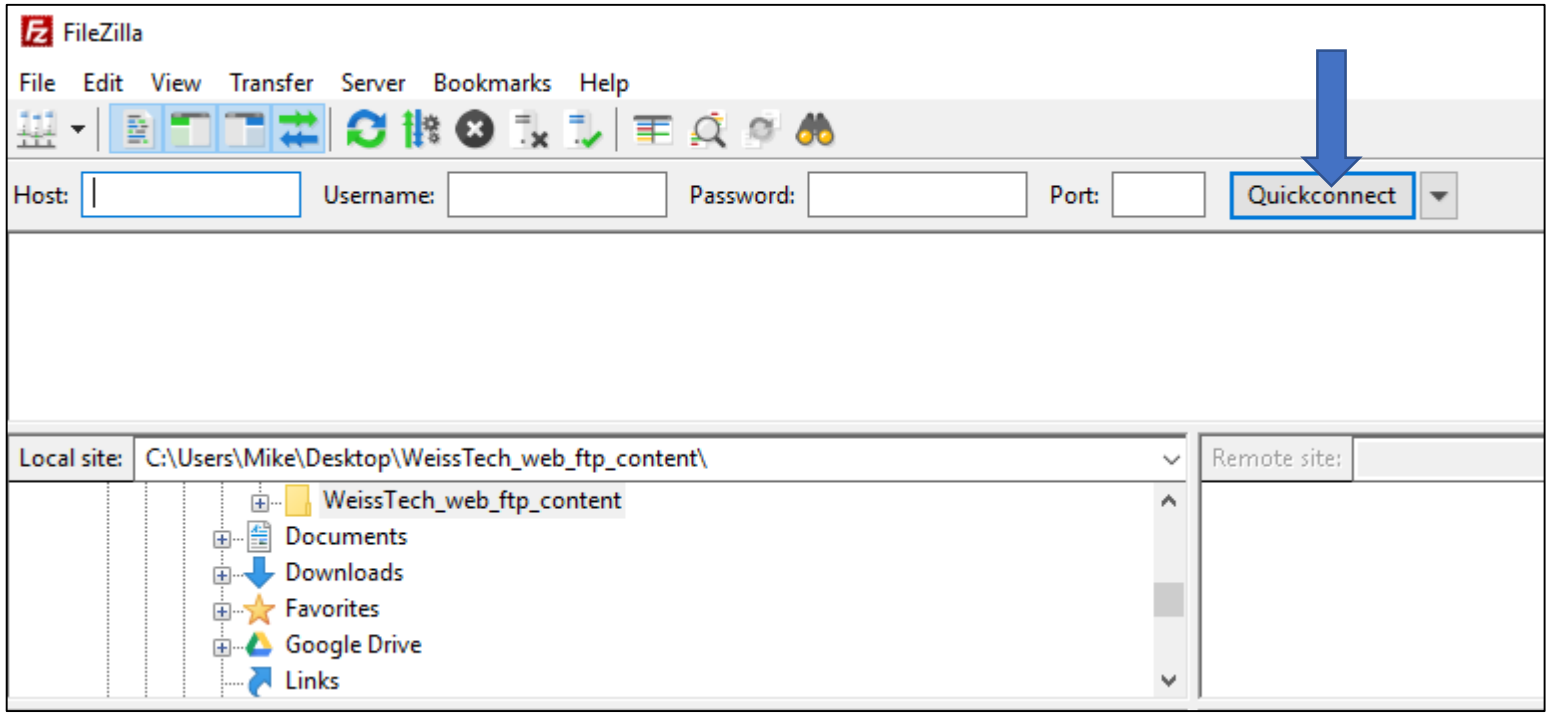


Download PuTTY

PuTTY is an SSH and telnet client, software that is available with source

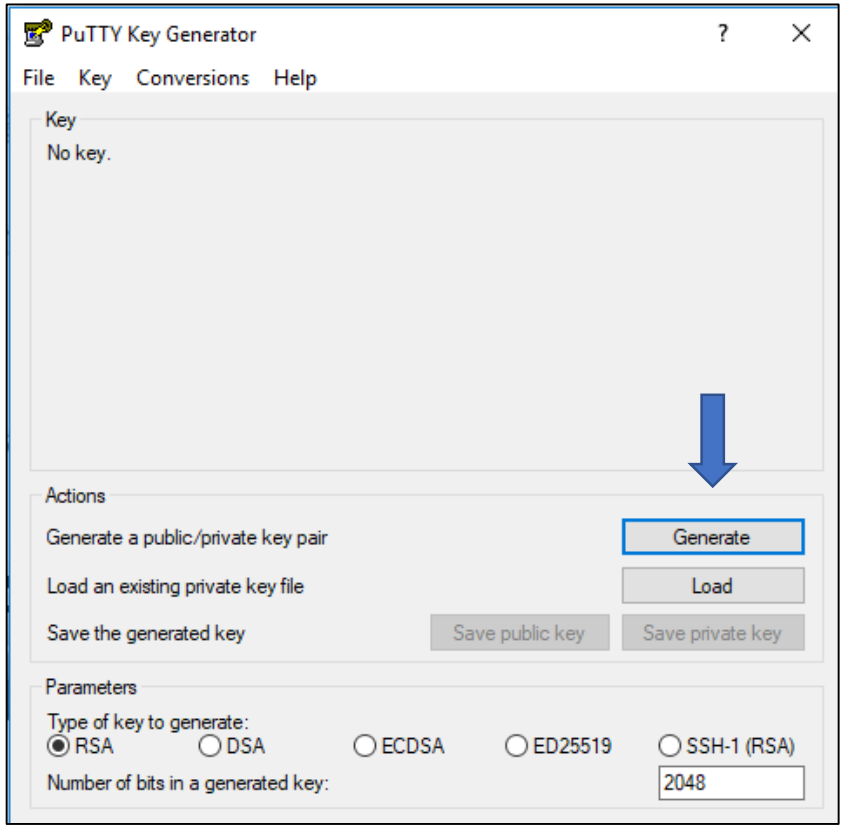
You can download PuTTY here.

Start up the Fillazilla FTP application.
Your version needs to have the <Quickconnect> button as shown here. If you don't see <Quickconnect> you may not have the **client** version of Filezilla.

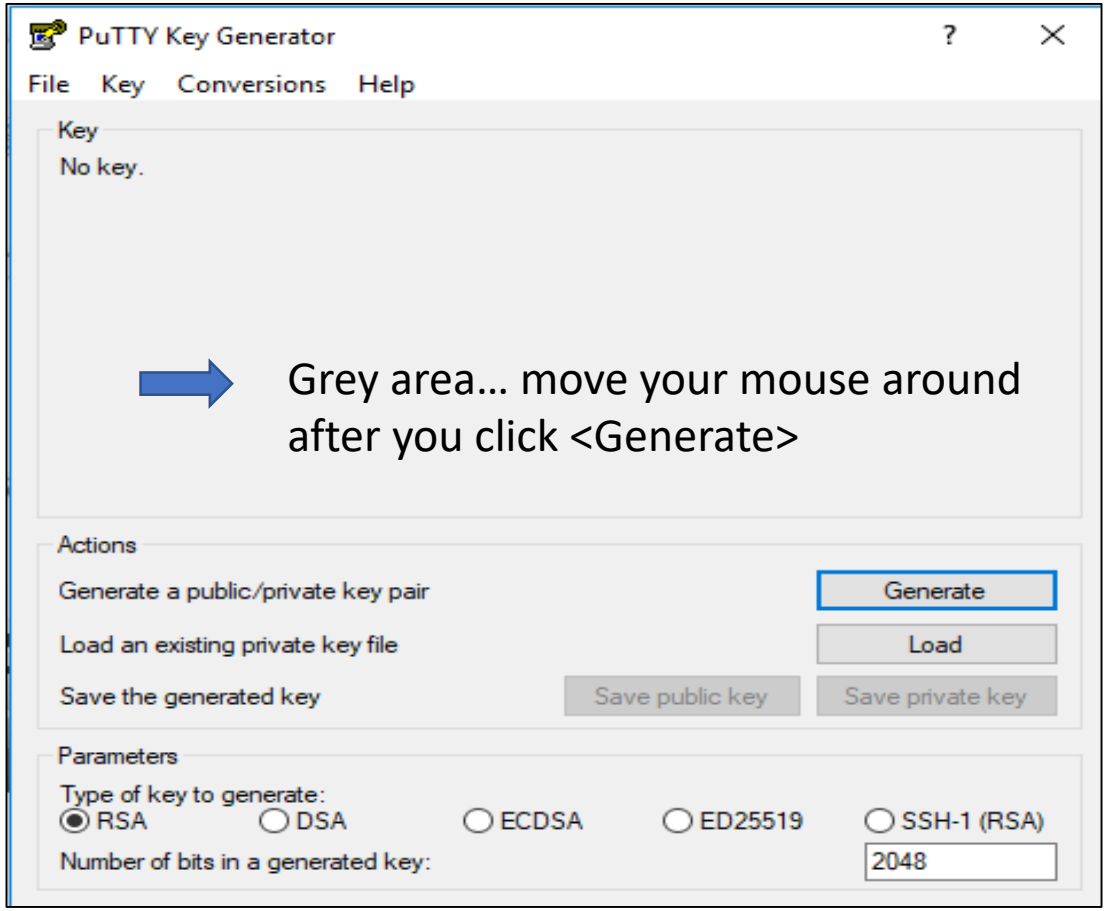


Now that Filezilla and PuTTY are installed we will use PuTTy to generate our SSH key. Start PuTTy and click the <Generate> button.

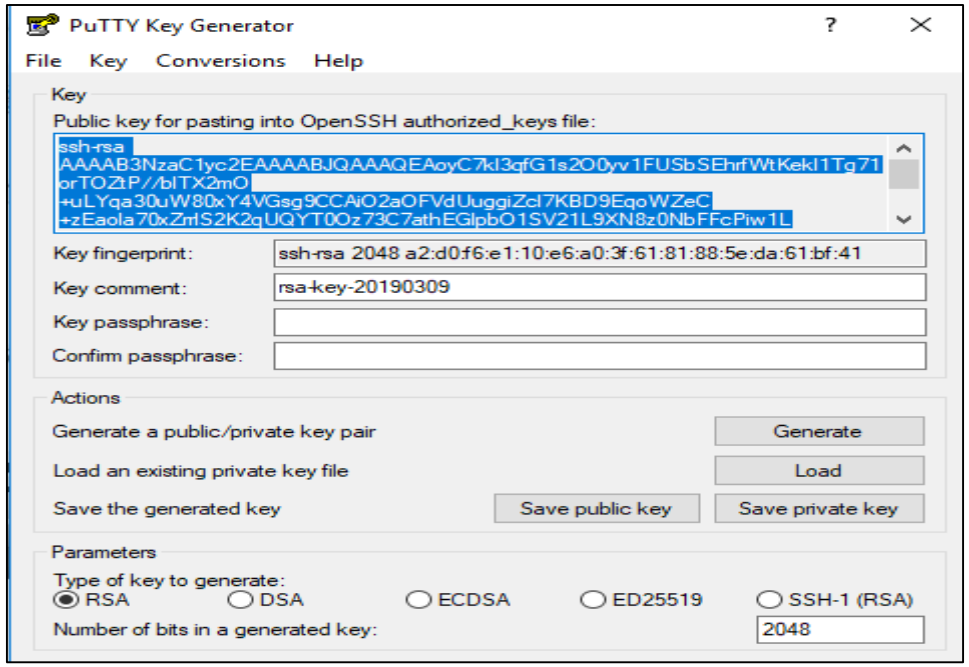
If you do not see the <Generate> button try a different version of PuTTY

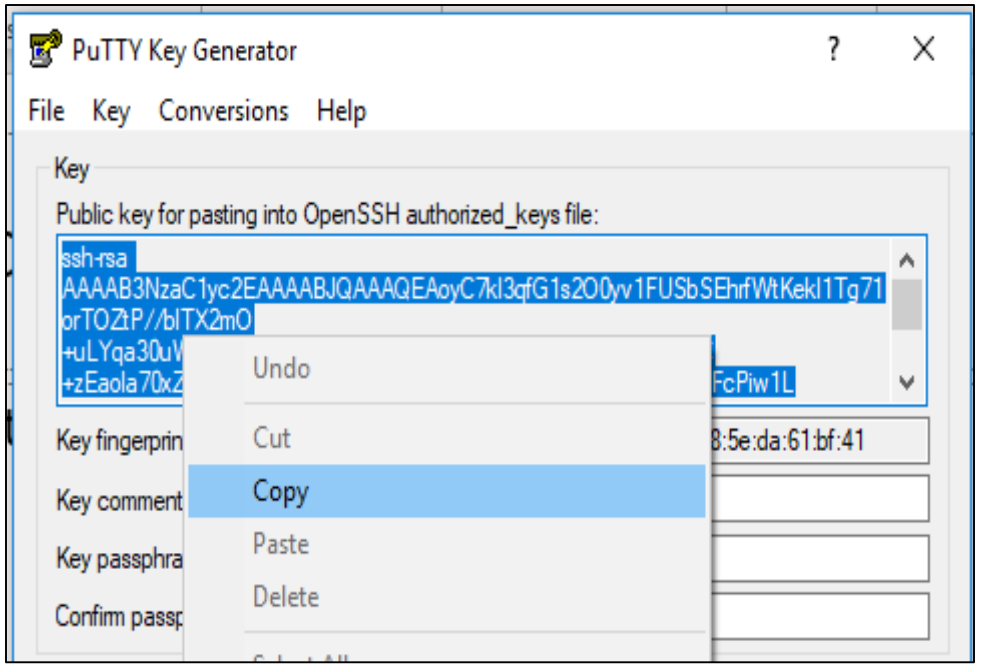


Note: once you hit the <Generate> button you will need to move your mouse around in the grey area of the PuTTY screen.



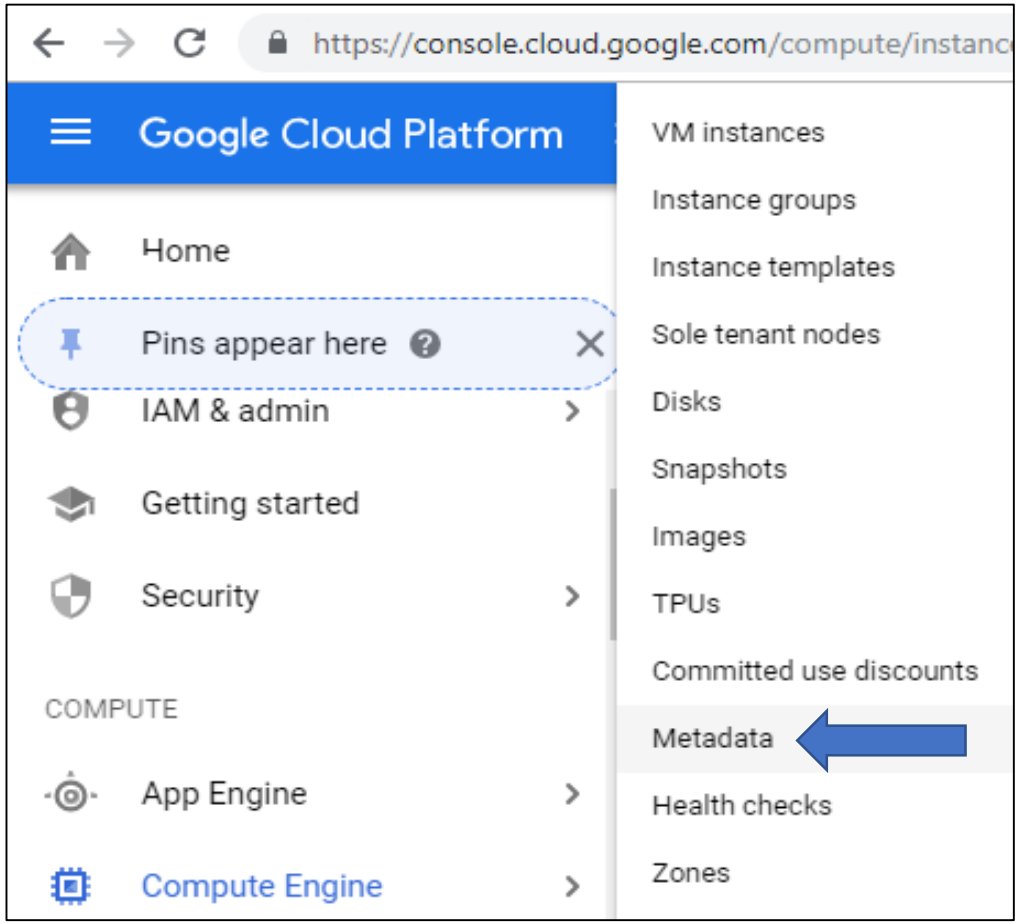
You have just generated a ‘key pair’. There is a public key that is **shown in blue** and a private key that is… private! Select all the text for the public key. Now copy all of the text, then click on the <Save private key> button.
[Note: There will be a message telling you that you have not created a passphrase… you do not need to create a passphrase!]
[You need to copy this private key to a folder location on your computer and call it ‘LinuxVMKey’ or something similar to that so you can identify it later!]

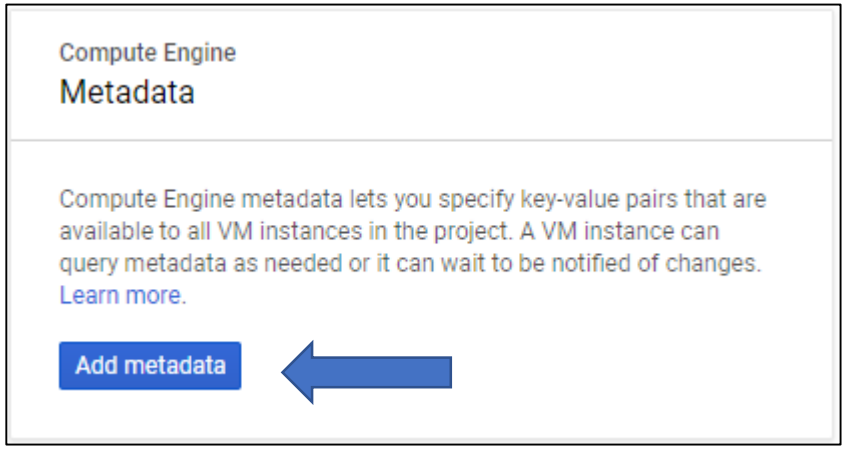




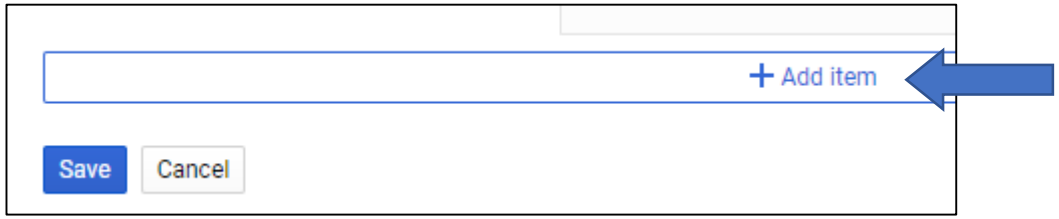
Note: leave PuTTY running, in case you need any of the information, until you are fully connected to your Google VM

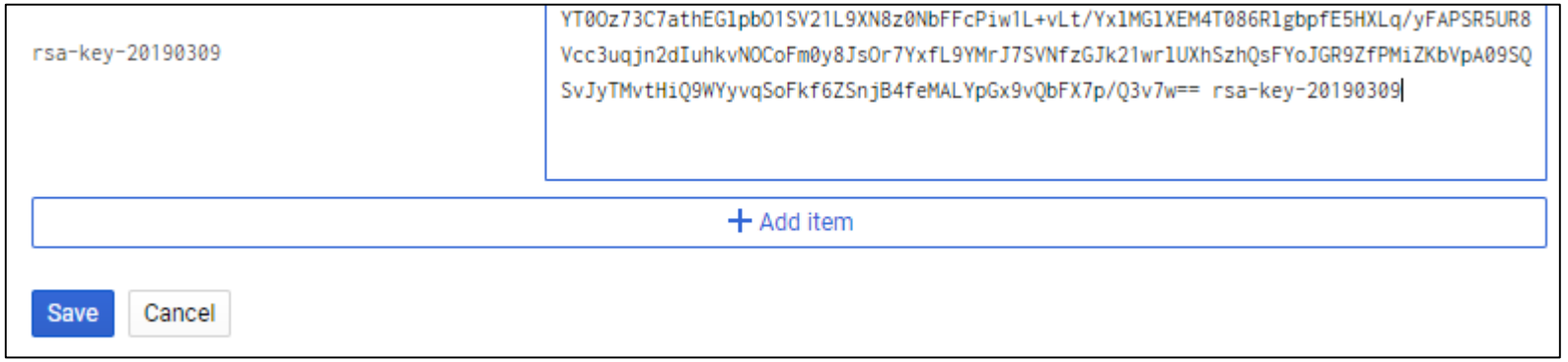
Now we will bring the copied text to the Compute Engine of our Google Cloud Platform:
Click on the Main menu, choose Compute Engine, then choose Metadata, then click <Add metadata> as shown:





Now, up at the top, click on <SSH keys> to view all of our SSH keys and click <Edit> then scroll to the bottom of the page and <Add item>. Finally, paste the private key that you had copied before into the box and <Save>



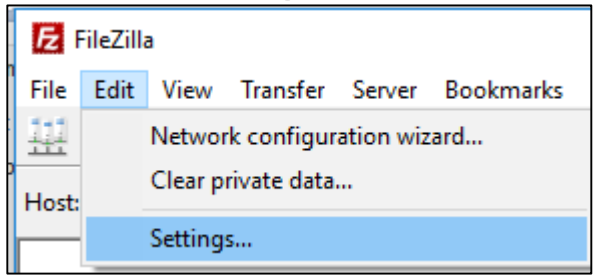


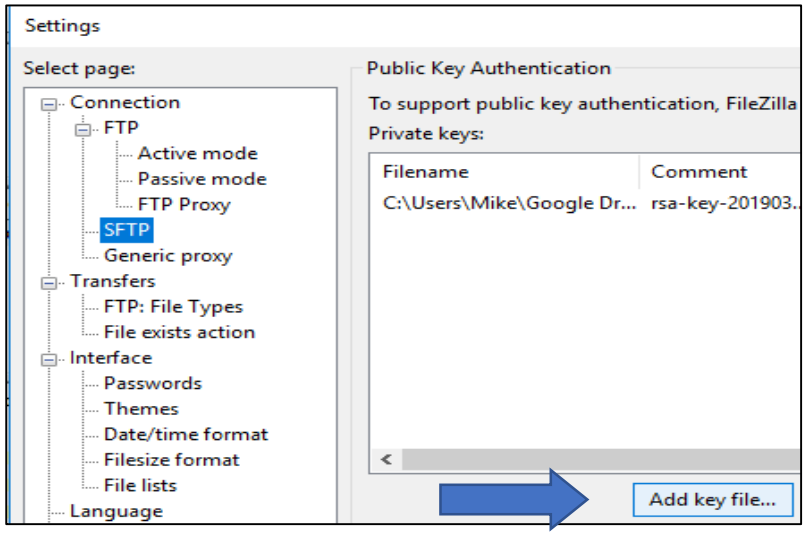
So far we have generated a public key and pasted it into our Google Cloud Platform console.

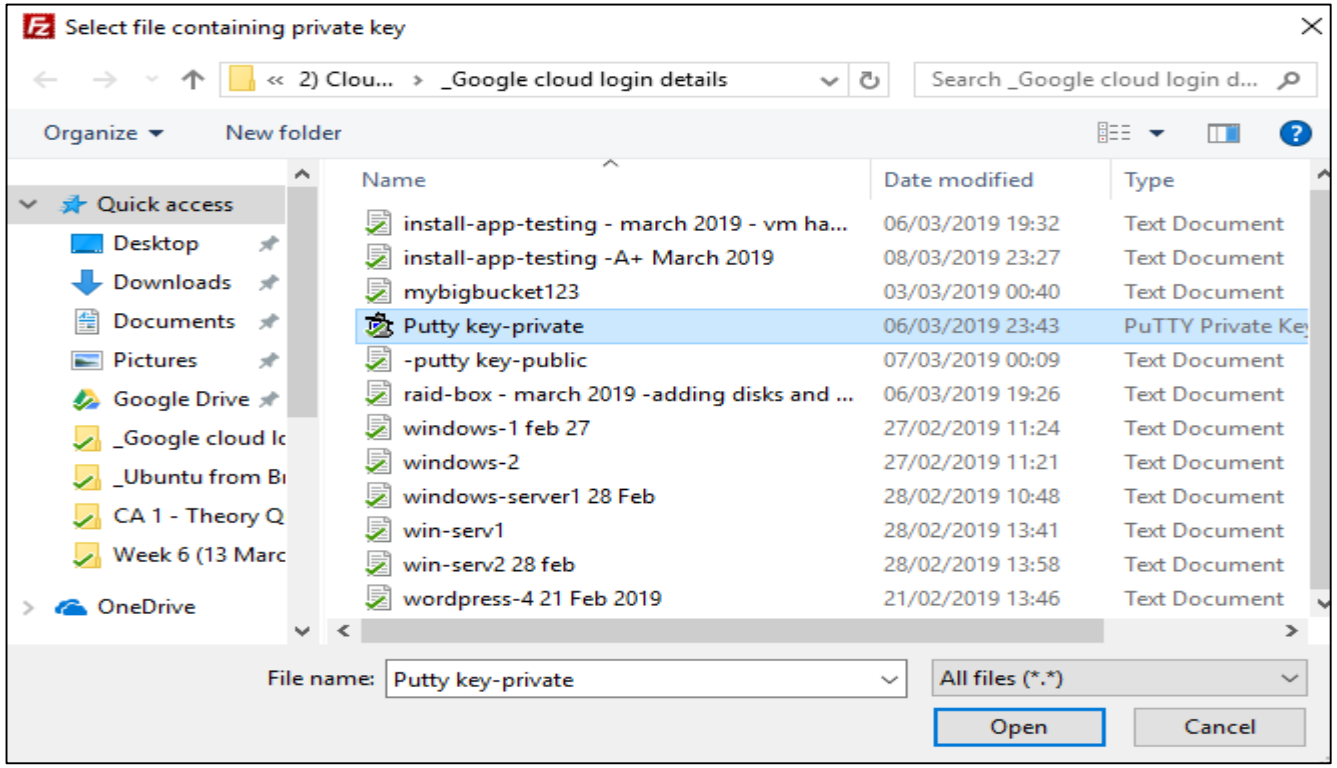
Now we need to add the private key into Filezilla.

First we go to the <Edit> menu in Filezilla and choose settings:

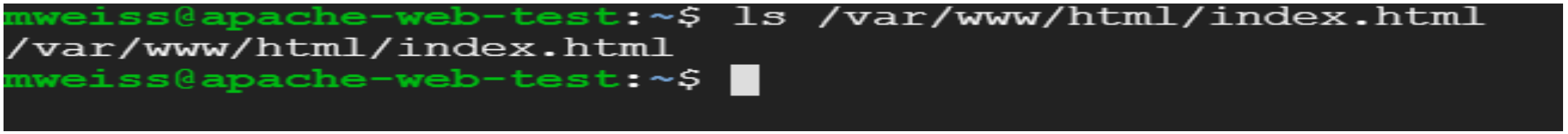
Then we go to SFTP and click on <Add key file…> and select your key file



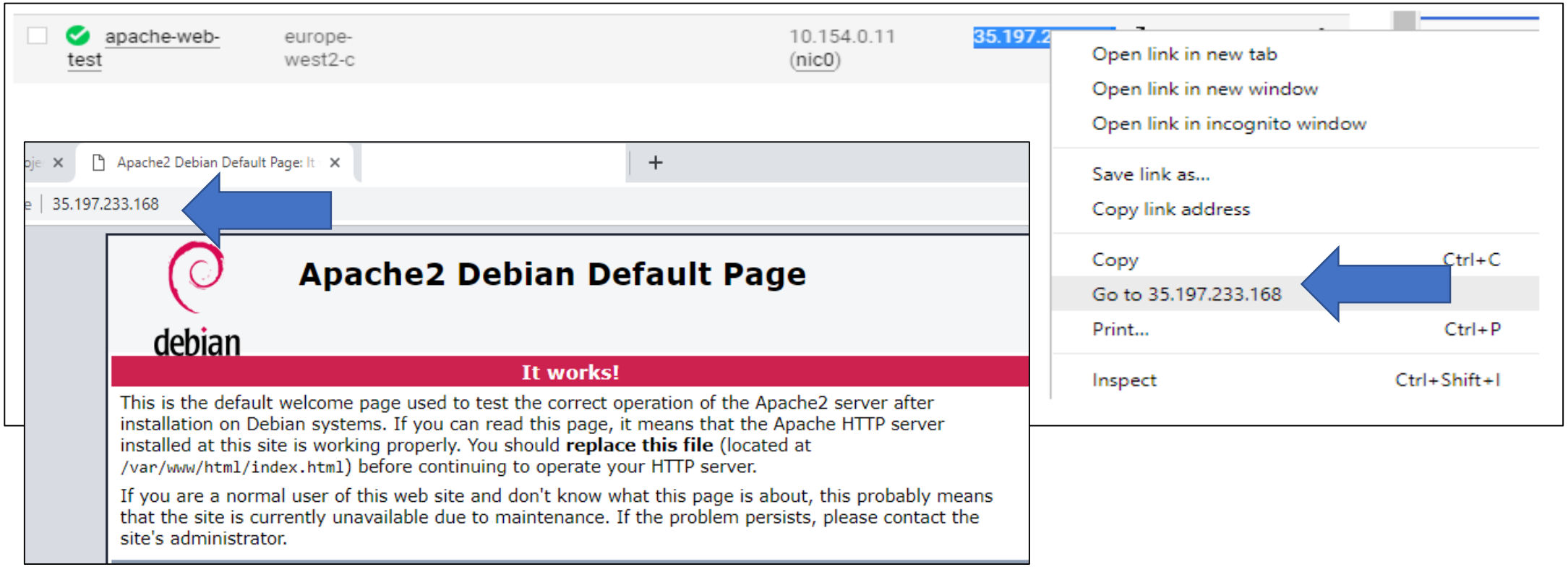




Once the private key is saved inside Filezilla we go back into our Google Cloud Platform and choose which VM instance we want to connect to.

- I have created a Debian Linux VM and installed the Apache web service.
- I will use Filezilla to copy my website files from my computer up to my Apache webserver which is running on Google Cloud.
- Once Apache (or NGINX) has been installed on the server we need to place the web content into the /var/www/html folder.
- Once Apache is installed we see that the default Apache web page is called index.html located in the /var/www/html folder.
- 

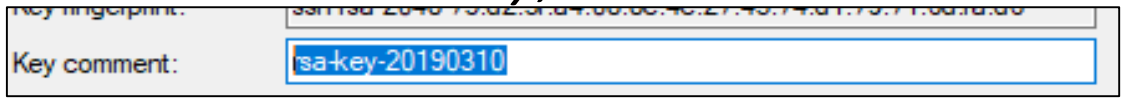
You convert your Linux VM into a web server by running this command: sudo apt install apache2
We can view the default Apache web page by using the external IP address of out Linux VM that is running on the Google cloud (the VM must be running)



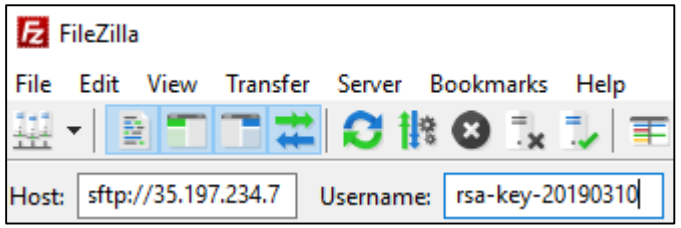
Once you have installed Apache or NGINX onto your Linux VM we are ready to use FileZilla to transfer files

- You need to know the external IP address of your VM.
- Go to you Google cloud console and copy the external IP address of the VM that you wish to connect to as shown below:
- Note that the internal IP address is always there but *the external IP address is only available when the VM is running*.

Next I go back to Filezilla and write sftp://
and then paste the external IP address which
will something look like this- [Host:] sftp://35.197.234.7
You will also need to enter your Username into Filezilla.
The Username is an RSA key, located in the PuTTY key comment:



You need to copy this and paste it into the Filezilla field called Username:



Now you should be able to connect up to the address of this Windows virtual machine when we click on <Quickconnect>

Below we see that I have used Filezilla
(and PuTTY) to connect to my Linux Gcloud VM.
I have navigated to the /var/www/html folder
where I need to put my web content.
The easy way to transfer files and folders with Filezilla is to ‘drag-and-drop’ your files or folders from the source (in the left side) to the target folder (on the right side)

When I attempted to use Filezilla to upload the folder from my computer to the VM I got an error message saying the following: open for write: **permission denied** Error: File transfer failed

The target permissions need to be changed inside the Linux VM

We need to change the permissions on the target folder which is /var/www/html
To change the permissions we must log in locally to our VM. We use the following command to view our current permissions:
stat -c "%a %n" /path/to/file ← we need to enter:
stat -c "%a %n" /var/www/html ← which shows us:
755 /var/www/html

```
mweiss@apache-web-test:~$ stat -c "%a %n" /var/www/html
755 /var/www/html
```

As you can see from the image, the file's permissions are set to 755. We must change them.

Linux uses the chmod command to change file and folder permissions

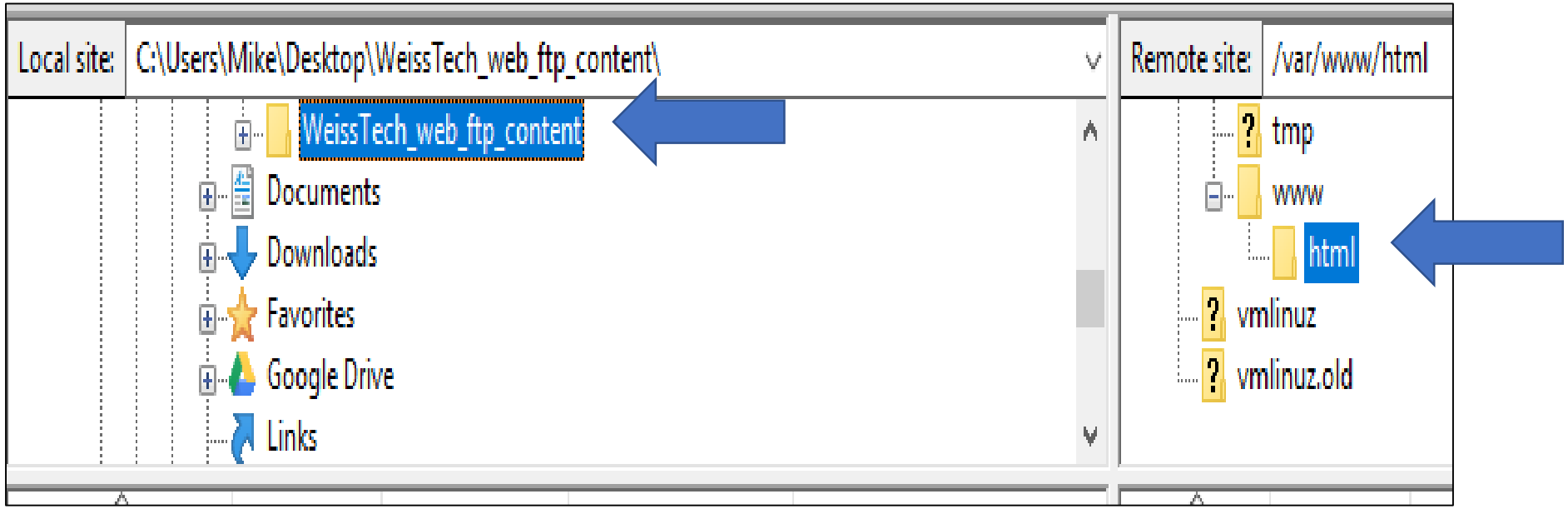
- Because the file's permissions are set to 755, you would not have sufficient permissions to edit the file via FTP.
- To fix this problem, you are going to change the file's permissions from 755 to 777, thereby granting all permissions.
- To do this, run the following command:

sudo chmod 777 /path/to/file

→ in our current situation this will be:

sudo chmod 777 /var/www/html

Now we see that we are able to drag-and drop files and folders from source to target



Finally, once we are sure that we have transferred the index.html file (and any other web content) into the correct location, we see our web site: