

Generate SSH Keys on PuTTY

1) Download PuttyGen from

<https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>

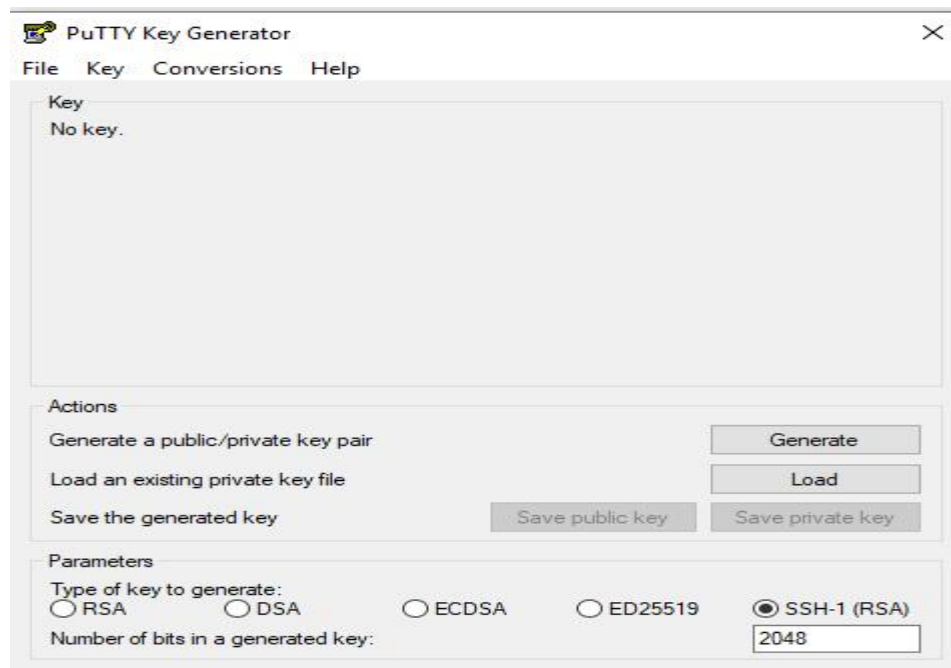
puttygen.exe (a RSA and DSA key generation utility)

32-bit: [puttygen.exe](#) (or by FTP) (signature)

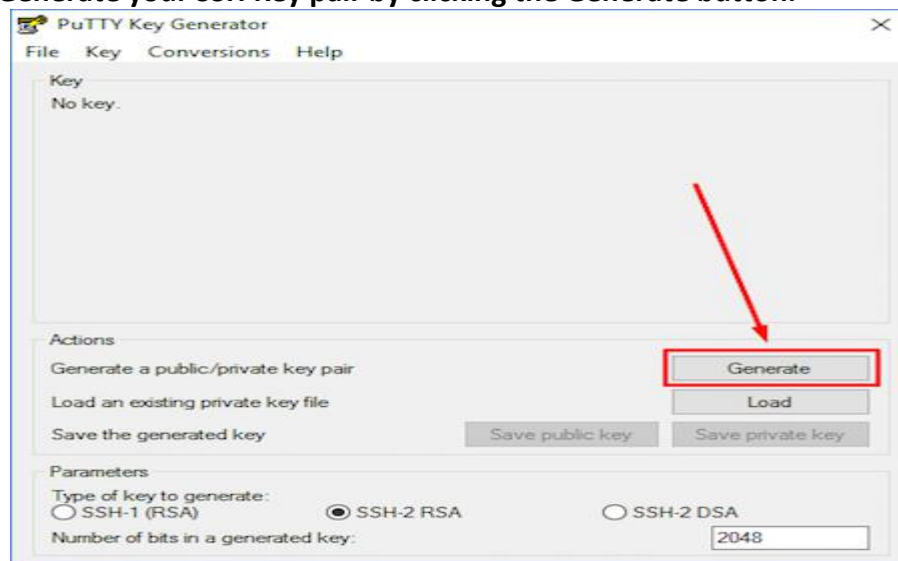
64-bit: [puttygen.exe](#) (or by FTP) (signature)

2) Generating your SSH Key pair :

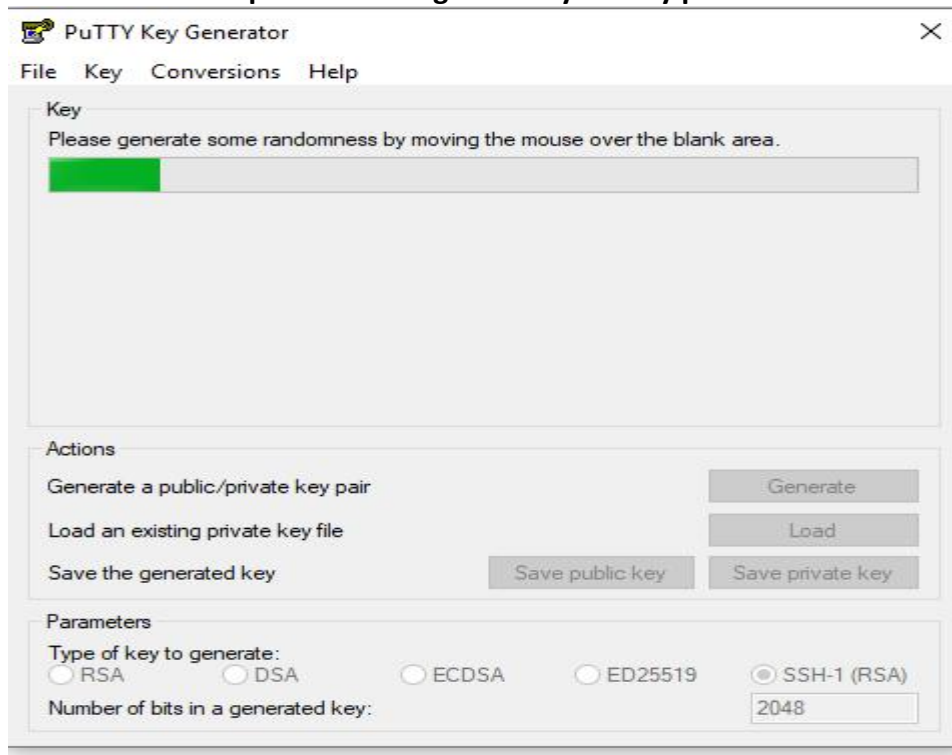
You should be able to see a window like this:



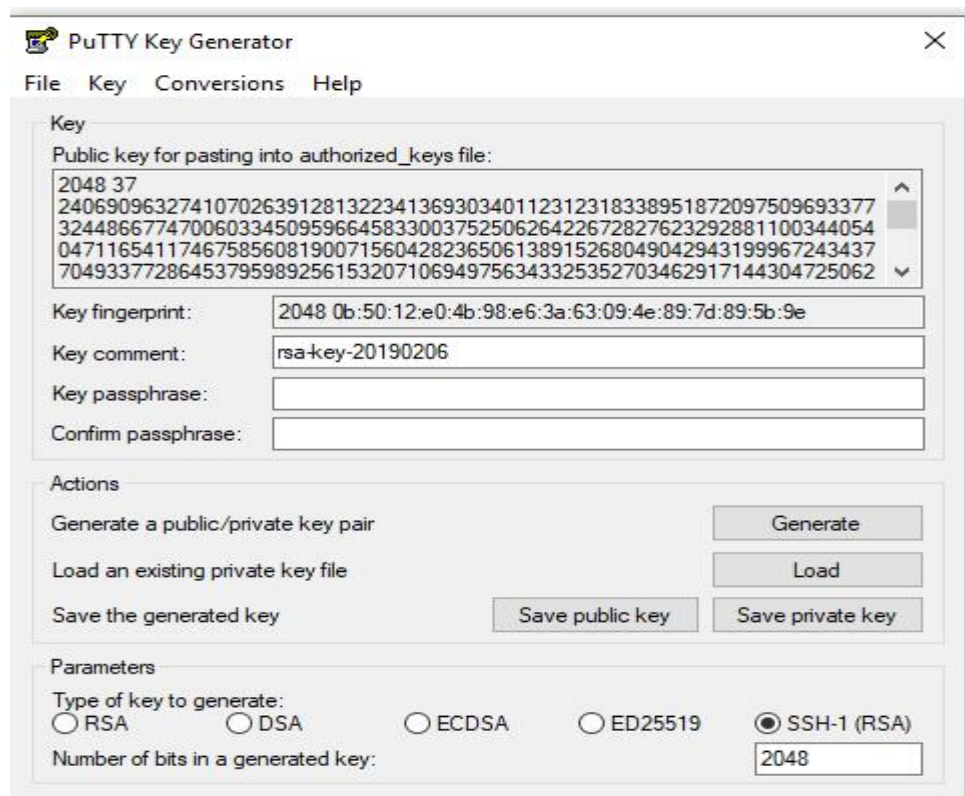
3) Generate your SSH Key pair by clicking the Generate button.



- 4) When the progress bar starts loading, move your mouse randomly across the area to load up the bar and generate your key pair.



- 5) Your public SSH key will be displayed on the screen:



- 6) For additional security it is highly recommended to think of a pass-phrase for your SSH key (However, you can also leave it empty):

Key

Public key for pasting into OpenSSH authorized_keys file:

```
ssh-rsa AAAAB3NzaC1yc2EAAAABJQAAAQEAhvP00Fsr
+Thra2AJAcGylTh8kb3IWSEWywymHwUzPgAGvErZJmrLB4oQDiCvdOZgBMfamcvV
78QqYPGxzGW2NWRraERMDdChVibYCY9IG8rV3wdp3FoxWAwYuyG6WfaPyTlm83
Br/C6P2j/2MZzluCNSwAGbx9Rd3LNfa3zyTWj/5+mrUdgg9jUBgLhuM
+Lm4H23U/RjMHtzbPo9kQTaSDPk+U
```

Key fingerprint: ssh-rsa 2048 fd:f7:6f:1f:70:40:a7:c8:aa:cd:59:f0:a3:02:bf:fe

Key comment: rsa-key-20161010

Key passphrase:

Confirm passphrase:

- 7) Save your private key to any desired location on your computer and name it anything you like.

Actions

Generate a public/private key pair

Load an existing private key file

Save the generated key

Save private key as:

← → ↕ ↶ ↷ This PC > Documents Search Documents

Organize New folder

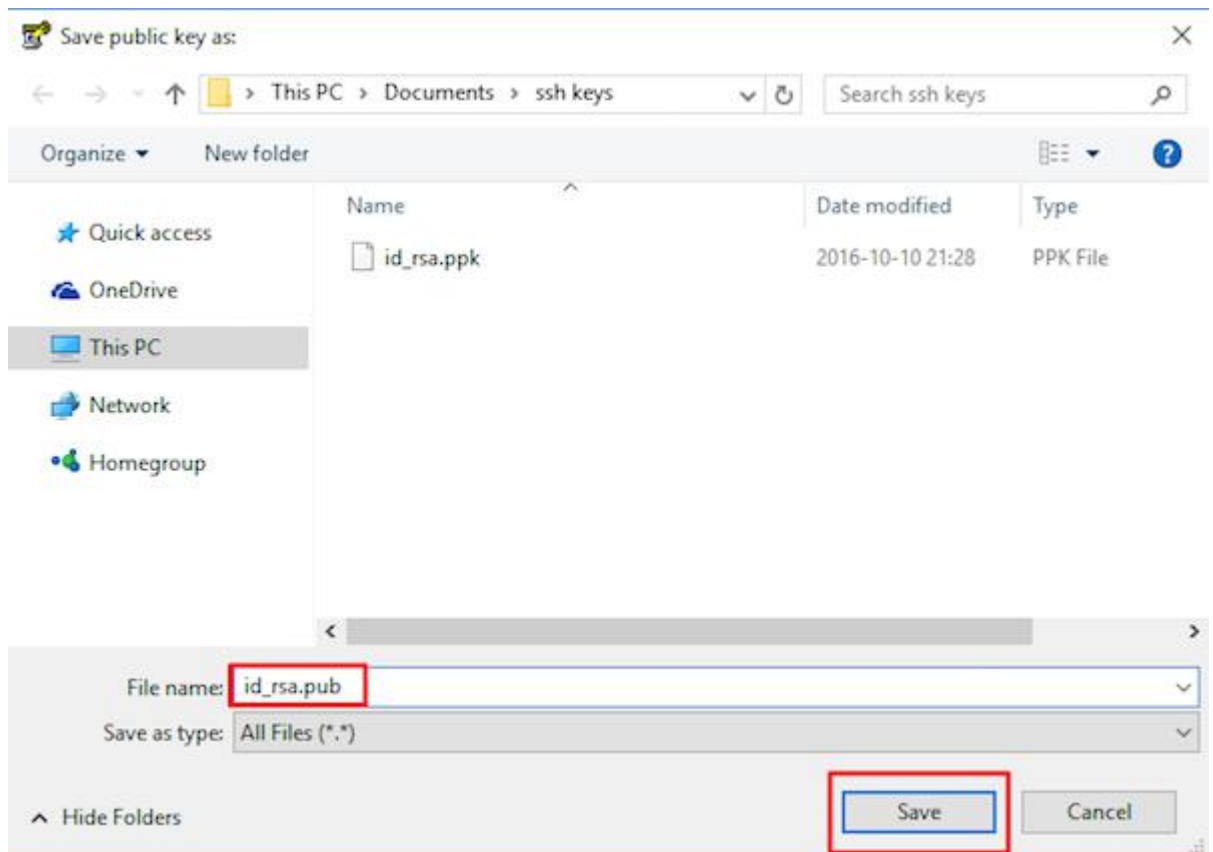
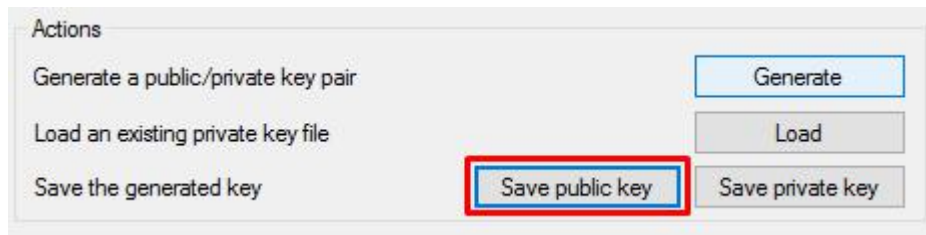
Name	Date modified	Type
Net Control 2	09-01-2019 18:00	File folder
Shared Toad	05-02-2019 13:39	File folder
SoftMaker	24-12-2018 11:21	File folder

File name: is_rsa_putty

Save as type: PuTTY Private Key Files (*.ppk)

Hide Folders

- 8) **Save your public key to the same location on your computer and name it anything you like.**

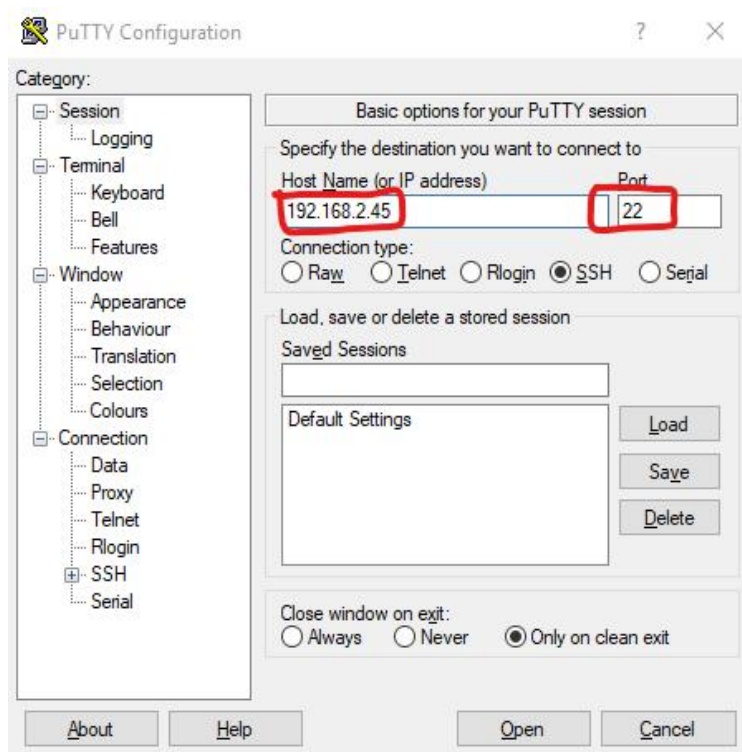


And that is it, you have generated your SSH Key pair. The private key will stay on your computer (do not provide it to anyone) while your public key needs to be uploaded to the server you wish to connect to.

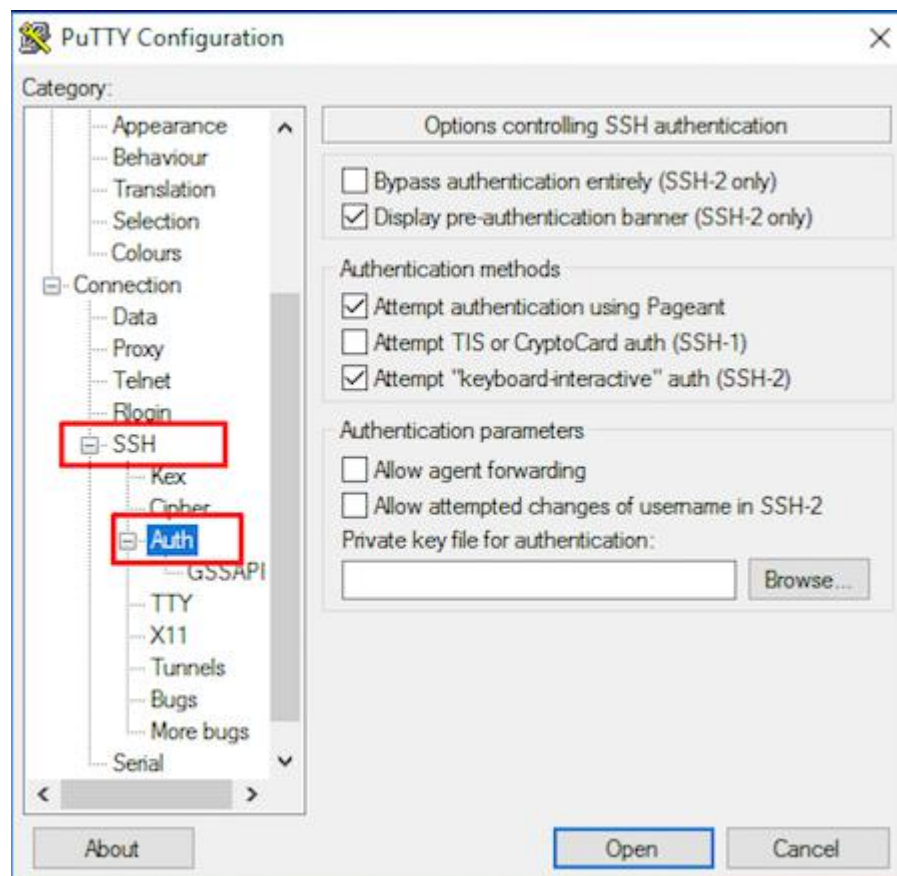
9) **Setting up your private key on PuTTY :**

In order for the server to recognize your computer when connecting from PuTTY, you need to attach the private key to PuTTY.

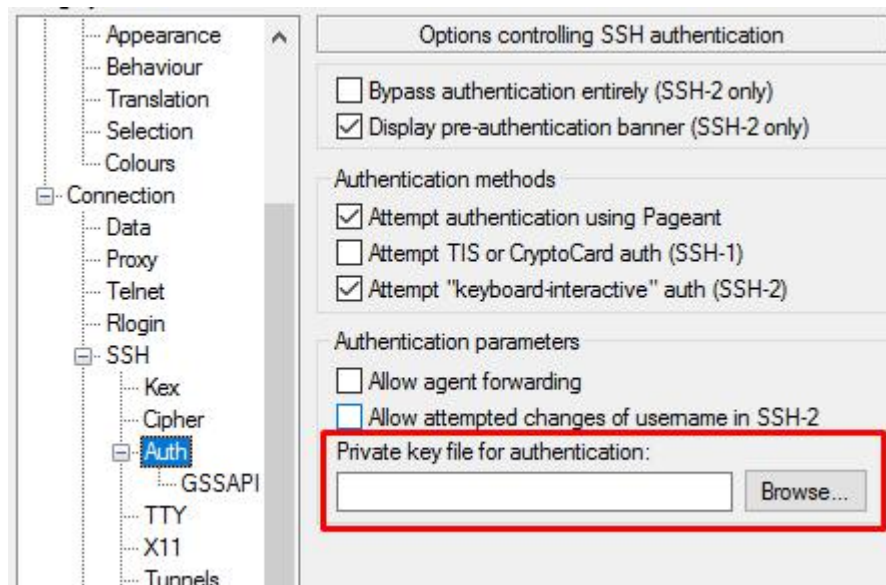
- 1) Open up PuTTY and enter Host-name (or) IP-address and Port Number.



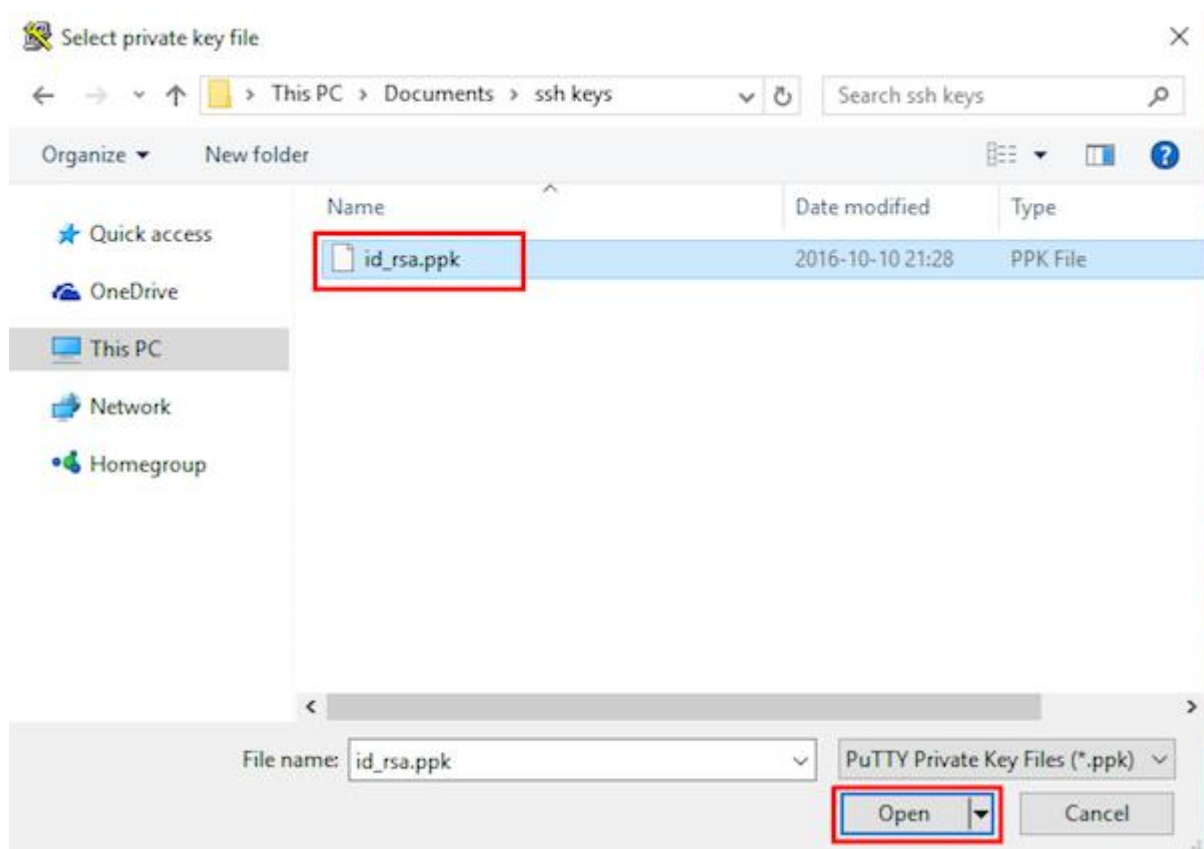
2) Navigate to Connection -> SSH -> Auth in the left sidebar.



3) Browse for your private key file in the field Private key for authentication



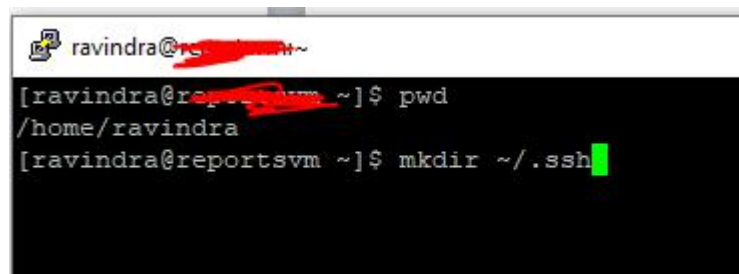
4) Select the private key file with .ppk ending and click Open.



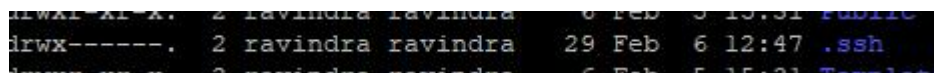
10) Adding public key to Linux server:

- 1) On your local computer, open public key file (id_rsa.pub) you generated in **Step 8** with any text editor and copy its contents (public key).
- 2) Connect to your Linux server using Putty.
- 3) If .ssh folder does not exist, create it together with authorized_keys file with the following command:

```
# mkdir ~/.ssh
```

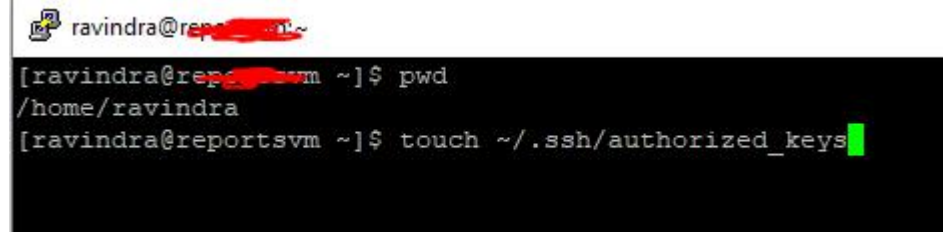


```
ravindra@reportsvm ~  
[ravindra@reportsvm ~]$ pwd  
/home/ravindra  
[ravindra@reportsvm ~]$ mkdir ~/.ssh
```

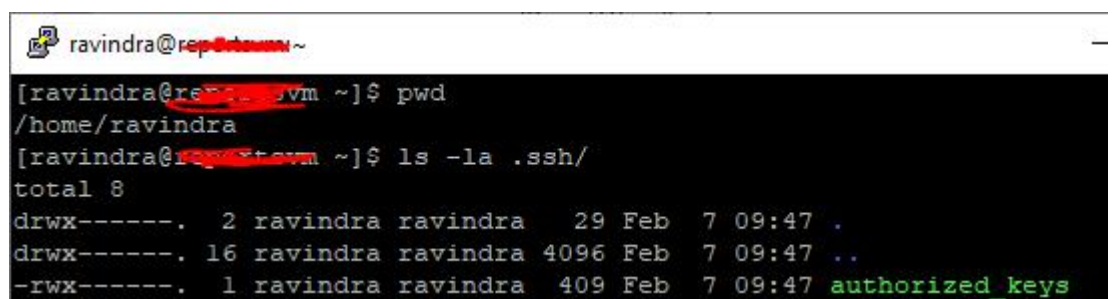


```
drwxr-xr-x. 2 ravindra ravindra 6 Feb 5 13:31 public  
drwx-----. 2 ravindra ravindra 29 Feb 6 12:47 .ssh  
drwxr-xr-x. 2 ravindra ravindra 6 Feb 5 15:31 Template
```

```
# touch ~/.ssh/authorized_keys
```



```
ravindra@reportsvm ~  
[ravindra@reportsvm ~]$ pwd  
/home/ravindra  
[ravindra@reportsvm ~]$ touch ~/.ssh/authorized_keys
```



```
ravindra@reportsvm ~  
[ravindra@reportsvm ~]$ pwd  
/home/ravindra  
[ravindra@reportsvm ~]$ ls -la .ssh/  
total 8  
drwx-----. 2 ravindra ravindra 29 Feb 7 09:47 .  
drwx-----. 16 ravindra ravindra 4096 Feb 7 09:47 ..  
-rwx-----. 1 ravindra ravindra 409 Feb 7 09:47 authorized_keys
```

4) Secure SSH Key file by changing permissions using below command:

```
# chmod 0700 ~/.ssh
# chmod 0644 ~/.ssh/authorized_keys
```

```
[ravindra@ravindrasvm ~]$ chmod 0700 ~/.ssh
[ravindra@ravindrasvm ~]$ chmod 0644 ~/.ssh/authorized_keys
```

5) Open **authorized_keys** file with vim text editor and past the **Public key** content in **authorized_keys** file

```
# vi ~/.ssh/authorized_keys
```

```
[ravindra@ravindrasvm ~]$ vi ~/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAABJQAAAAQEAru2kE3ybIqGCS1kSFdmpNjBDEYvYdv
Okp+SoRSkp2TvAkYW/LIx3TY/6j7HpfkMVft9LjJUImlAk5bC8OgpED9pD8Zry0KC1I
K/jRKCUnp+zp+Y3z9RzmcjtLbLGX4cAg6aN8wH46pWsAWk6wyqqNwgKoFAlIjKcTK3
4XDR7b4mT7TlpcmxZQrmy3CFNGMYEgdyHmeCta/TCno2Rt/shi2UTk42siIw7GwIztI
```

TO Allow Users To login through .ppk file only

1) Disable Username/Password Logins:

Up to now, you can log in with your private/public key pair and still with username/password logins, so if someone doesn't attach a private key to his security, we must disable the username/password logins (you should do this only when you know that your key-based logins are working, because if they aren't and you disable username/password logins, then you have a problem...).

To disable the username/password logins, we must modify the sshd configuration file. On Linux systems, it's **/etc/ssh/sshd_config**. You should set **Protocol to 2** (1 is insecure and should not be used!), **PasswordAuthentication to no**, and **UsePAM to no** (or comment out the UsePAM line), e.g. like this:

```
root@localhost:~
login as: root
root@192.168.2.69's password:
Last login: Wed Feb  6 16:28:53 2019 from 192.168.2.213
[root@localhost ~]# vi /etc/ssh/sshd_config
[root@localhost ~]#
```



```
# Set this to 'yes' to enable
# and session processing. If t
# be allowed through the Chall
# PasswordAuthentication. Dep
# PAM authentication via Chall
# the setting of "PermitRootLo
# If you just want the PAM acc
# PAM authentication, then ena
# and ChallengeResponseAuthent
# WARNING: 'UsePAM no' is not
# problems.
Protocol 2
PasswordAuthentication no
UsePAM no
```

2) Then restart sshd.

systemctl restart sshd (In centos 7)

service sshd restart (In centos 6)

```
root@localhost:~
[root@localhost ~]# systemctl restart sshd
[root@localhost ~]#
```

3) Check sshd service status:

systemctl status sshd (In centos 7)

service sshd status (In centos 6)

```
root@localhost:~
[root@localhost ~]# systemctl status sshd
● sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2019-02-06 17:39:48 IST; 3min 38s ago
     Docs: man:sshd(8)
           man:sshd_config(5)
  Main PID: 3757 (sshd)
    CGroup: /system.slice/ssh.service
            └─3757 /usr/sbin/sshd -D

Feb 06 17:39:48 localhost.localdomain systemd[1]: Stopped OpenSSH server daemon.
Feb 06 17:39:48 localhost.localdomain systemd[1]: Starting OpenSSH server daemon...
Feb 06 17:39:48 localhost.localdomain sshd[3757]: Server listening on 0.0.0.0 port 22.
Feb 06 17:39:48 localhost.localdomain sshd[3757]: Server listening on :: port 22.
Feb 06 17:39:48 localhost.localdomain systemd[1]: Started OpenSSH server daemon.
[root@localhost ~]#
```

4) Now try to login linux server with out .ppk file from putty.

