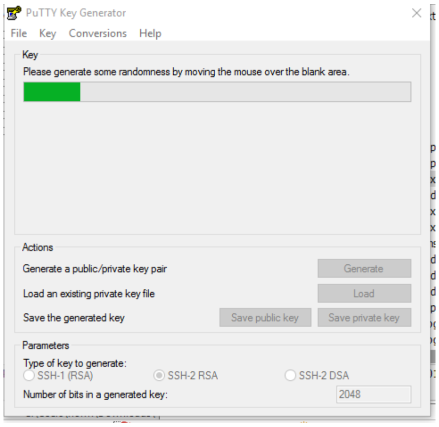
This explains how to create a key pair for connecting to the raspberrypi via ssh. This example is for a windows pc using **PuTTY** a remote terminal program. This example also uses **PuTTYgen** to generate the public and private keys. The download page for both of these tools can be found at this link: http://www.putty.org/

The first step with **PuTTYgen** is to generate a key pair. The program uses random movements of your mouse to generate the key.



Please generate some randomness by moving the mouse over the blank area.

Generate a public/private key pair

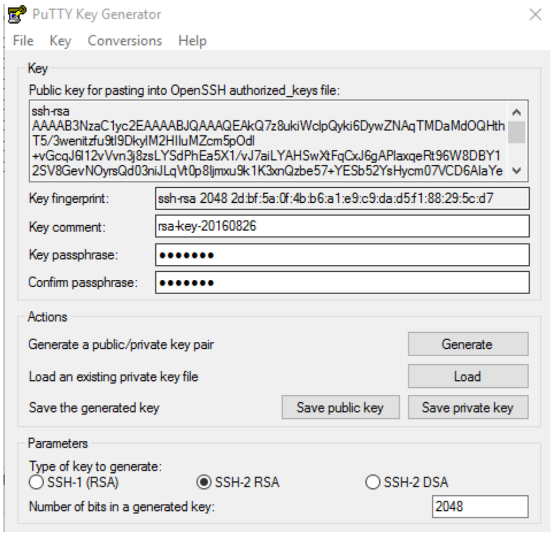
Load an existing private key file

Save the generated key

Type of key to generate:
○ SSH-1 (RSA) ● SSH-2 RSA ○ SSH-2 DSA

Number of bits in a generated key: 2048

Once complete the key is displayed as well as other information including a comment that can be added to the private key file. At this point you can also enter in a passphrase that will be needed to log in as an additional security feature. If you do not enter anything here, no passphrase will be required. Press the buttons to save the public and private keys. It is important that you keep the private key secure as it enables access to the pi for anyone who has the key. You will need to edit the public key that resides on the pi and does not need to be kept private but measures need to be taken so that the key is not replaced by a malicious user. How the private key is used by **PuTTY** described in the next section.



Public key for pasting into OpenSSH authorized_keys file:

```
ssh-rsa
AAAAB3NzaC1yc2EAAAABJQAAAQEAkQ7z8ukiWclpQyki6DywZNAqTMDaMdOQHth
T5/3wenitzfu9tl9DkylM2HIluMZcm5pOdl
+vGcqJ6l12vVvn3j8zsLYSdPhEa5X1/vJ7aiLYAHSwXtFqCxJ6gAPlaxqeRt96W8DBY1
2SV8GevNOyrsQd03niJLqVt0p8ljmxu9k1K3xnQzbe57+YESb52YsHycm07VCD6AlaYe
```

Key fingerprint: ssh-rsa 2048 2d:bf:5a:0f:4b:b6:a1:e9:c9:da:d5:f1:88:29:5c:d7

Key comment: rsa-key-20160826

Key passphrase: ••••••••

Confirm passphrase: ••••••••

Generate a public/private key pair

Load an existing private key file

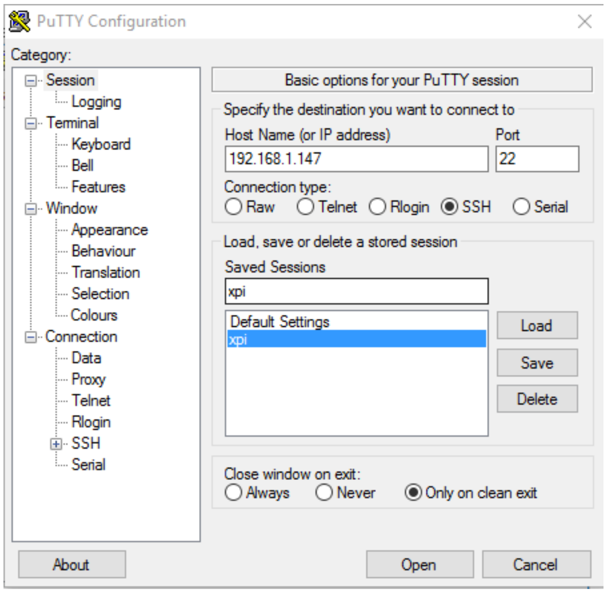
Save the generated key

Type of key to generate:
○ SSH-1 (RSA) ● SSH-2 RSA ○ SSH-2 DSA

Number of bits in a generated key: 2048

These are the steps to use the private key in the **PuTTY** tool.

Step 1: Enter the "Session" dialog, enter in the IP address of the pi, port 22, SSH, only on clean exit. You can save the information in a session for reuse (this one is called xpi).



Basic options for your PuTTY session

Specify the destination you want to connect to

Host Name (or IP address): 192.168.1.147

Port: 22

Connection type: ○ Raw ○ Telnet ○ Rlogin ● SSH ○ Serial

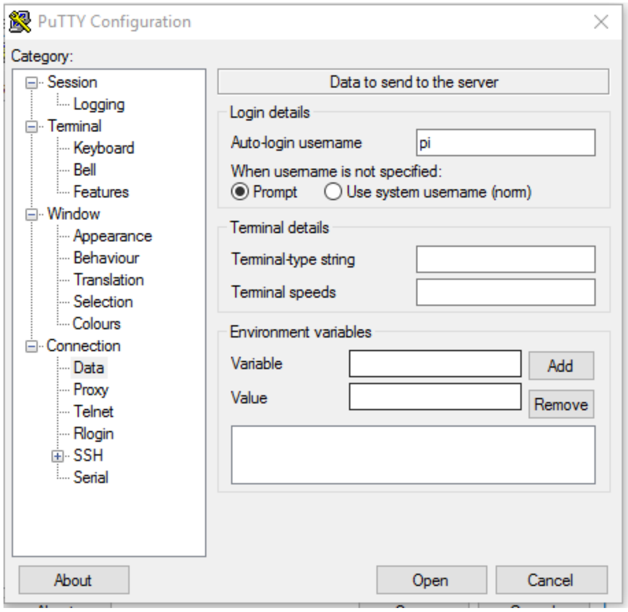
Load, save or delete a stored session

Saved Sessions: xpi

Default Settings
xpi

Close window on exit: ○ Always ○ Never ● Only on clean exit

Step 2: under Connection->Data enter the user name (pi) for auto login.



Data to send to the server

Login details

Auto-login username: pi

When username is not specified: ● Prompt ○ Use system username (norm)

Terminal details

Terminal-type string:

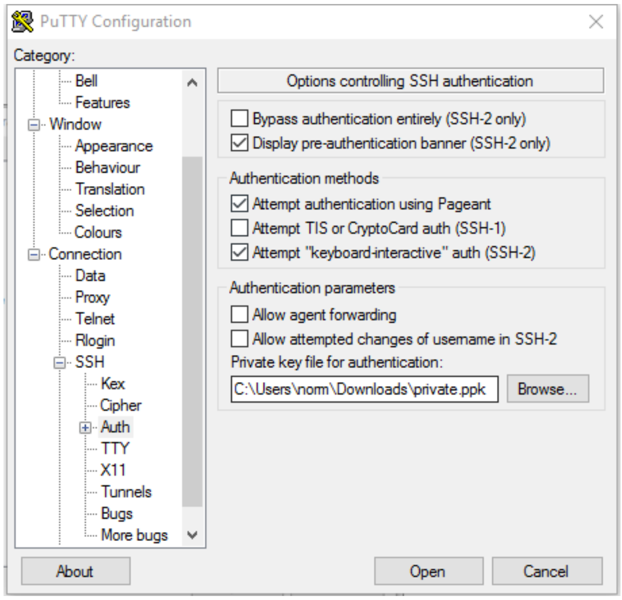
Terminal speeds:

Environment variables

Variable:

Value:

Step 3: select Connection->SSH->Auth to bring up the authentication dialog. Match the checked and unchecked boxes. Then Browse for the private key file you created with **PuTTYgen**. Select Open and you should see a window. IF you entered in a passphrase, this will be required at this time, otherwise if the keys match, you will be logged in without any password or passphrase. IF you enter in the correct passphrase, you will be logged in without a password.



Options controlling SSH authentication

□ Bypass authentication entirely (SSH-2 only)
☑ Display pre-authentication banner (SSH-2 only)

Authentication methods

☑ Attempt authentication using Pageant
□ Attempt TIS or CryptoCard auth (SSH-1)
☑ Attempt "keyboard-interactive" auth (SSH-2)

Authentication parameters

□ Allow agent forwarding
□ Allow attempted changes of username in SSH-2

Private key file for authentication:
C:\Users\norm\Downloads\private.ppk Browse...

This section describes how to edit the public key file generated by **PuTTYgen** so that it is compatible with the pi/debian software.   This is *an example* of the public key file that is generated by **PuTTYgen**:

```
---- BEGIN SSH2 PUBLIC KEY ----
Comment: "rsa-key-20160826"
AAAAB3NzaC1yc2EAAAABJQAAAQEAsU8qpCoL4x10ywlRwa+7yBcgMyUvL/RTVVPG
Cb4WyeAB7o5q8s+/uf7yG5gaEQzeEvO+LrjsZp4HPsf5FlvfDaS9sXqriJ2WALBf
DYw1/YROKSvIk+dmcp6QLGc4OUqvzgBsR7AbibXbSrphy8BpRqSY1F/ipA8m7toX
Tv3L936GBr25KSZOb9wmE+L0Q+CNizY4s/a7oiwltvxgGwB+j85FjR0mRg08C3u+
tD47HKoDmbT+W06hDAkFAW+qscwDHQ3ito/MihlvLJh9NIr/BV66LWz/cKe9W0Xf
t7dQkCRAaDDlbjpw4dbtgDNFspivWJZRzRIrasf9Cc3VXqmW9Q==
---- END SSH2 PUBLIC KEY ----
```

To make this work on the pi you need to edit the file to contain:

```
ssh-rsa
AAAAB3NzaC1yc2EAAAABJQAAAQEAsU8qpCoL4x10ywlRwa+7yBcgMyUvL/RTVVPGCb4Wye
AB7o5q8s+/uf7yG5gaEQzeEvO+LrjsZp4HPsf5FlvfDaS9sXqriJ2WALBfDYw1/YROKSvI
k+dmcp6QLGc4OUqvzgBsR7AbibXbSrphy8BpRqSY1F/ipA8m7toXTv3L936GBr25KSZOb9
wmE+L0Q+CNizY4s/a7oiwltvxgGwB+j85FjR0mRg08C3u+tD47HKoDmbT+W06hDAkFAW+q
scwDHQ3ito/MihlvLJh9NIr/BV66LWz/cKe9W0Xft7dQkCRAaDDlbjpw4dbtgDNFspivWJ
ZRzRIrasf9Cc3VXqmW9Q==
```

This is taking the only the key from the file **PuTTYgen** makes and adding ssh-rsa in front of the key. Even though there are line breaks shown here, the file contains just one line starting with ssh-rsa (space) then the key ending in   …. Q==

`ssh-rsa (space) KEY  --all as one line.`

This file should reside in ~pi/.ssh named authorized_keys. Protections on the .ssh directory should be set to 700, the authorized_keys file to 600.