Using Putty on Windows to login Linux securely via OpenSSH

- Submitted by: Man-wai CHANG
- Update by: Man-wai CHANG
- Date Submitted: 31 May 2006
- Document Version: 1.0
- Last Updated: 02/02/2008 17:46:43

This is a guide about using Putty on Windows with OpenSSH on Linux. You would learn about how to:

- configure OpenSSH on linux side to accept version 2 public-key authentication.
- create public and private keys with OpenSSH on the linux side,
- convert OpenSSH keys to Putty format using puttygen.exe at the Window side,
- use putty.exe to talk to OpenSSH using the converted private key.

> I would assume that you have OpenSSH installed. As per 31-May-2006, the latest version of OpenSSH was 4.3p1. Your Linux distribution may likely use an older version, however.

Configuring OpenSSH to accept public-key authentication

To enable your OpenSSH to accept version 2 public key, you would need to modify /etc/ssh/sshd_config. You could use vi editor (or whatever editor you are familiar with) to uncomment/add/modify the following lines to /etc/ssh/sshd_config:

```
# the default SSH port is 22, you could alter it if necessary
Port 22

# accept version 2 keys only
Protocol 2

# NEVER allow root to login directly over the net
PermitRootLogin no
StrictModes yes
MaxAuthTries 3

# enable public-key authentication
RSAAuthentication no
PubkeyAuthentication yes

# securing your OpenSSH
# do not use host-based authentication for security reason
RhostsRSAAuthentication no
HostbasedAuthentication no
IgnoreUserKnownHosts yes
PermitEmptyPassword no

# do not allow telnet-type login for security reason
```

```
ChallengeResponseAuthentication no
PasswordAuthentication no

X11Forwarding yes
X11DisplayOffset 10
```

After you have made changes to /etc/ssh/sshd_config, you would need to restart the OpenSSH daemon by executing `/etc/init.d/ssh restart` (on Ubuntu).

Generating OpenSSH private and public key pair

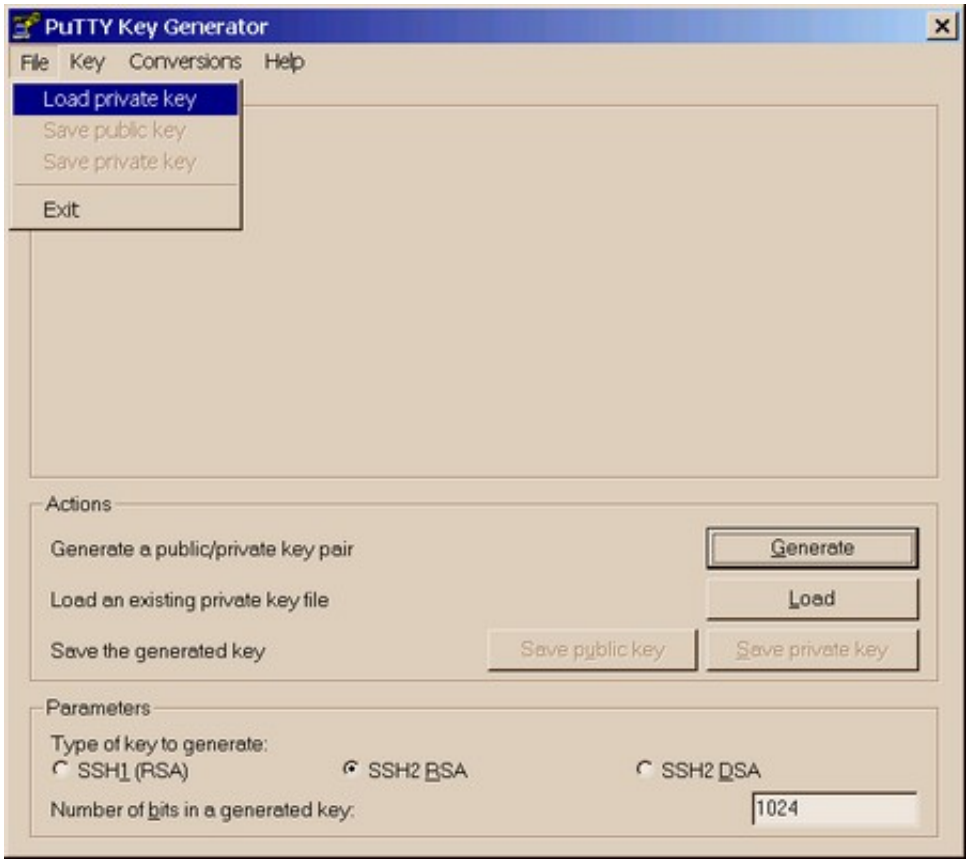
To use public key authentication, the first step is to generate a pair of private and public keys on the Linux side. I would assume that you login as a user called "toylet".

```
1. Login Linux as user "toylet". You could do it at the Linux console
   or via telnet.
2. Execute `ssh-keygen -t rsa` to generate a version 2 public
   and private key pair into directory /home/user/.ssh.
   The passphrase is optional (but preferred).

toylet@server:~$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/toylet/.ssh/id_rsa):
/home/toylet/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/toylet/.ssh/id_rsa.
Your public key has been saved in /home/toylet/.ssh/id_rsa.pub.
The key fingerprint is:
ec:f4:3f:b5:fe:2f:de:22:6c:42:8c:38:ad:6c:5e:96 toylet@server

3. Execute `cd /home/toylet/.ssh`
4. You should see 2 files: id_rsa and id_rsa.pub.
   Now execute the following command:
   cp id_rsa.pub authorized_keys
5. Copy /home/toylet/.ssh/id_rsa from Linux to Windows.
```

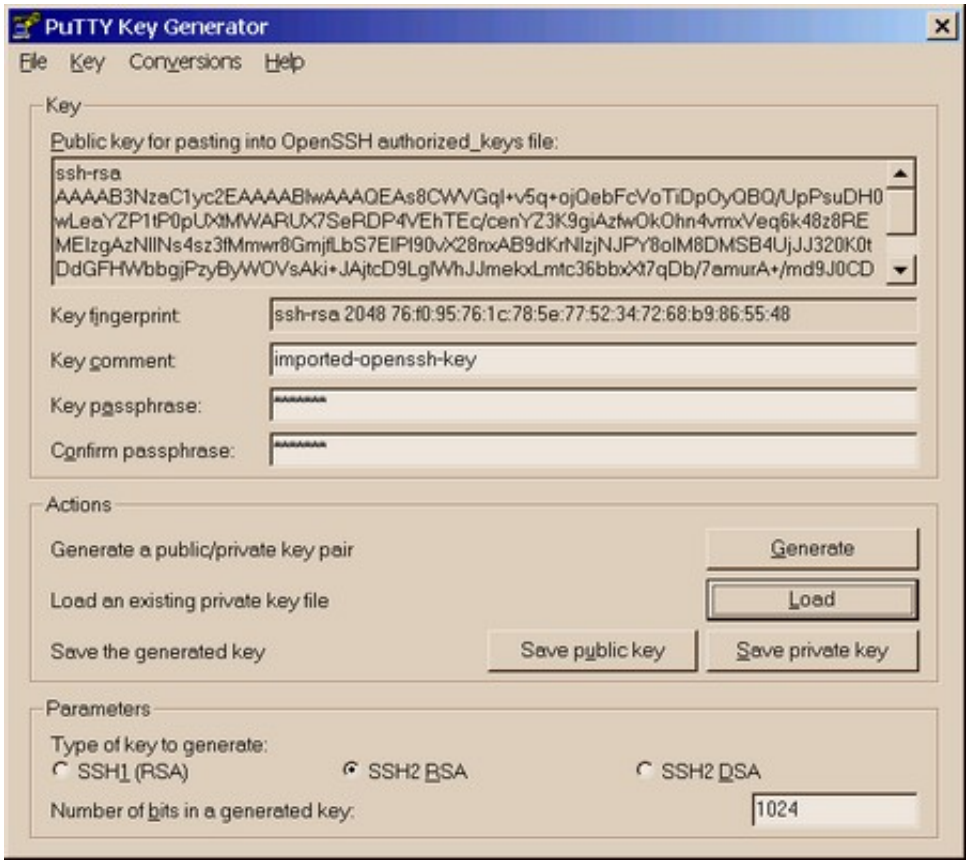
Converting the OpenSSH private key to Putty format

Next, we head to the Windows side. In step 4, you created two key files (id_rsa and id_rsa.pub). Putty cannot directly open OpenSSH keys. We need to convert id_rsa to id_rsa.ppk using a program called puttygen.exe.

```
6. At the Windows side, download puttygen.exe from Putty website.
7. Execute puttygen.exe
```



8. Click File->Load Private Key, load the file "id_rsa" from Step 5.
   Enter the passphrase if you used it in step 2.

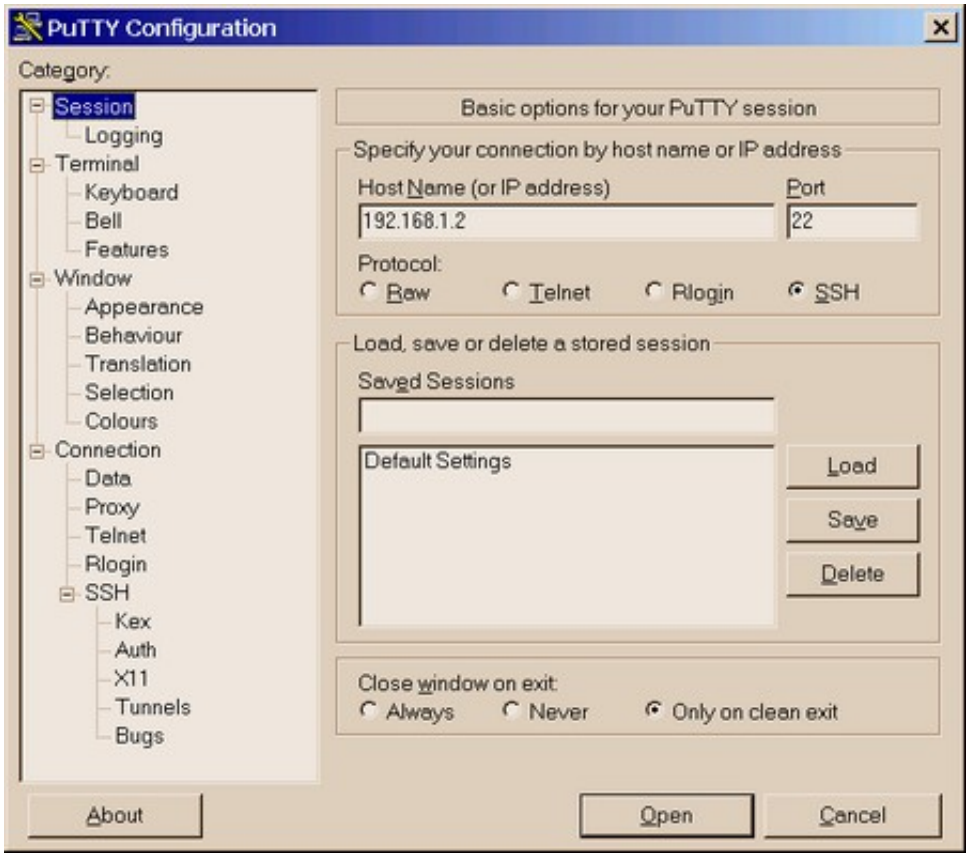


9. Now the key has been loaded as in the figure above.
   Hit the button "Save private key".
   The converted key would be saved as "id_rsa.ppk".

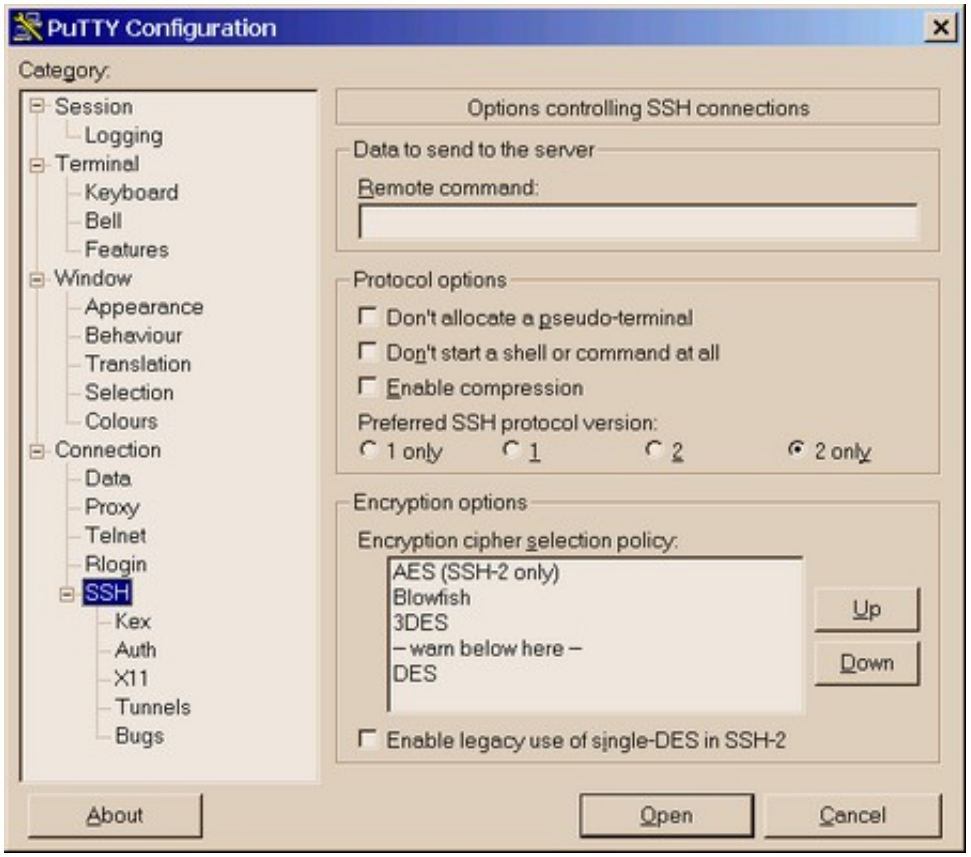
Logging in Openssh using id_rsa.ppk

Download putty.exe from [Putty website](). It's time to really login OpenSSH using putty.exe on Windows side. The steps here would be a little bit more complicated.

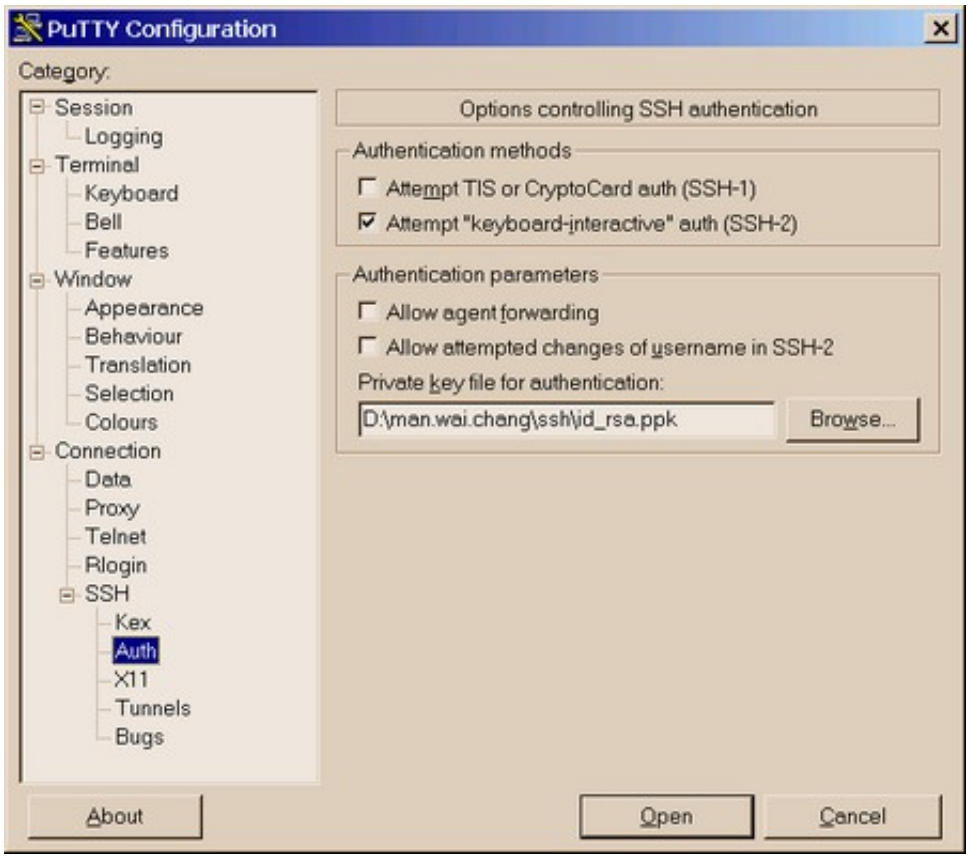
```
10. Invoke putty.exe
10.1. Click "Session" in the sidebar.
```



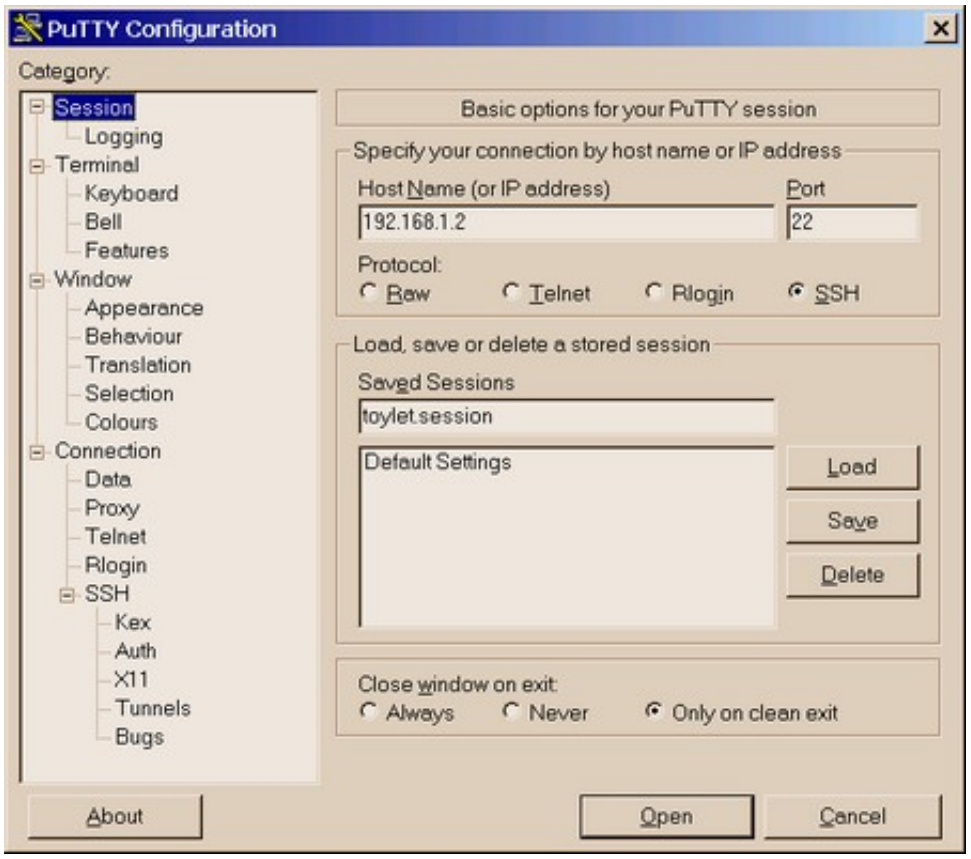

```
10.1.1. Enter ip address of your server (e.g., 192.168.1.2)
10.1.2. Click "SSH" in the Protocol option
10.2. Choose "SSH" under "Connection" in the sidebar
```



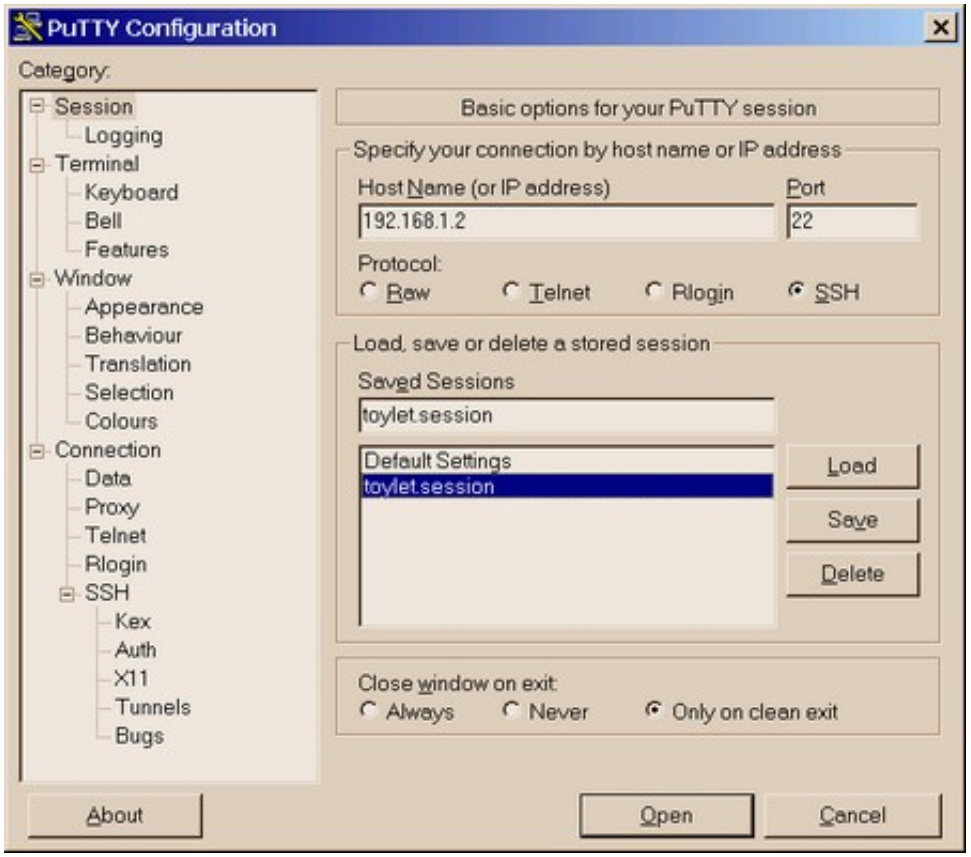
10.2.1. In "Preferred SSH protocol version", select "2 only"
10.2.2. click "Auth" under "SSH"



10.2.2.1. Hit the Browse button, select the file "id_rsa.ppk" from Step 9.
10.3. hit "Session" again in step 10.1



10.3.1. Enter a name (e.g. "toylet.session") in the textbox directly
under "Saved Sessions".
10.3.2. Hit the "Save" button. The name "toylet.session" would appear in
the listbox of "Saved Sessions".




```
10.4. Double-click "toylet.session". Now you would be presented
      with a login screen for OpenSSH.
10.4.1. Enter the linux user name "toylet"
10.4.2. Enter the passphrase if you specified it in step 2.

Login as: toylet
Authenticating with public key "imported-openssh-key"
Passphrase for key "imported-openssh-key":
Last login: Wed May 31 12:35:00 2006 from 192.168.1.10
toylet@server:~$

11. You have successfully logged into your Linux server via OpenSSH.
```

Epilogue

- You should change both your private and public keys periodically by repeating the steps above.
- You may disable the telnet daemon foreever since telnet doesn't encrypt the connection, allowing eavedropping easily.