Day 2-2-1 - ssh

Using Public Key Cryptography

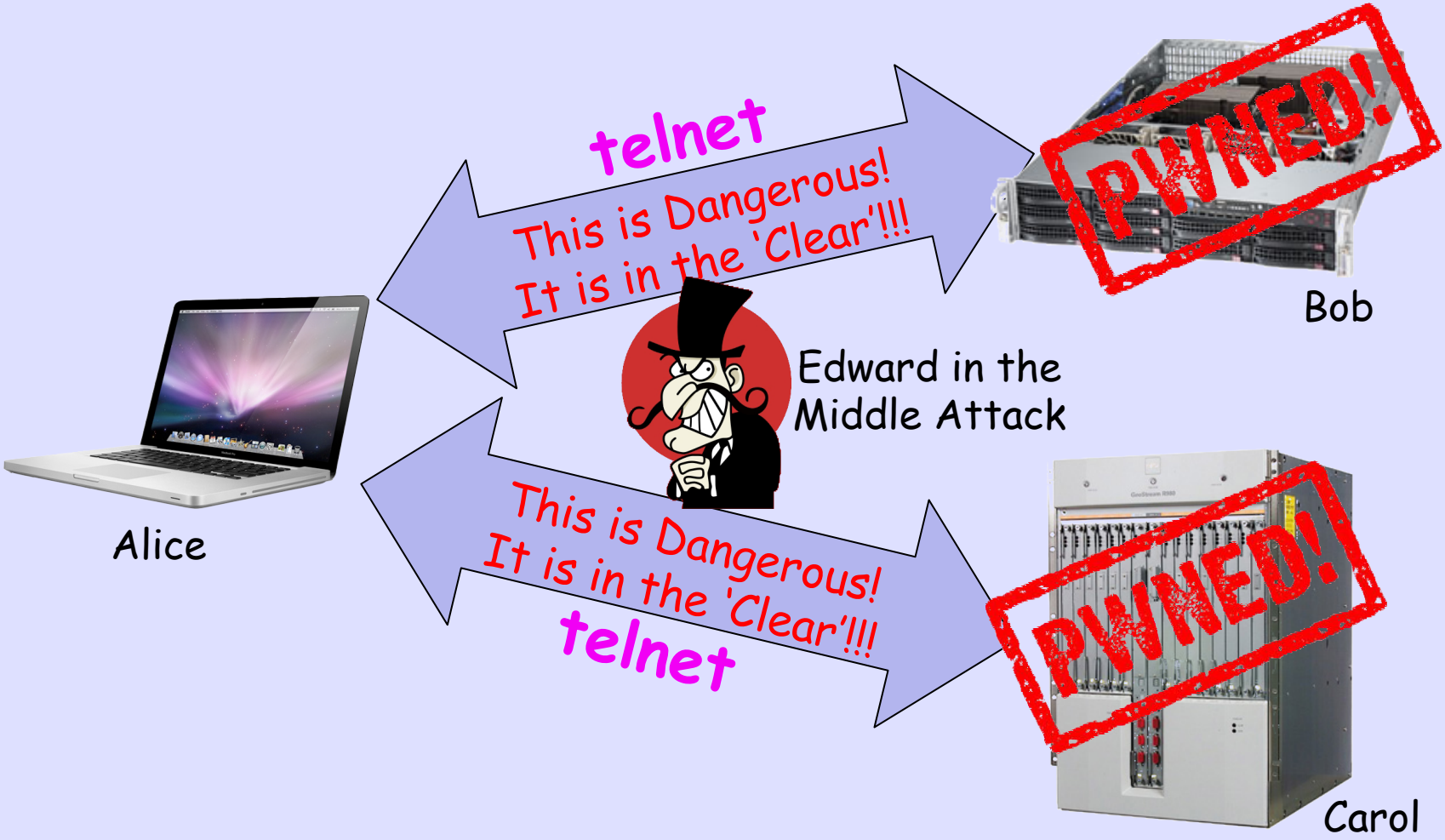
Keying, Key Exchange, and Session Setup

Communicate Safely with Remote Systems

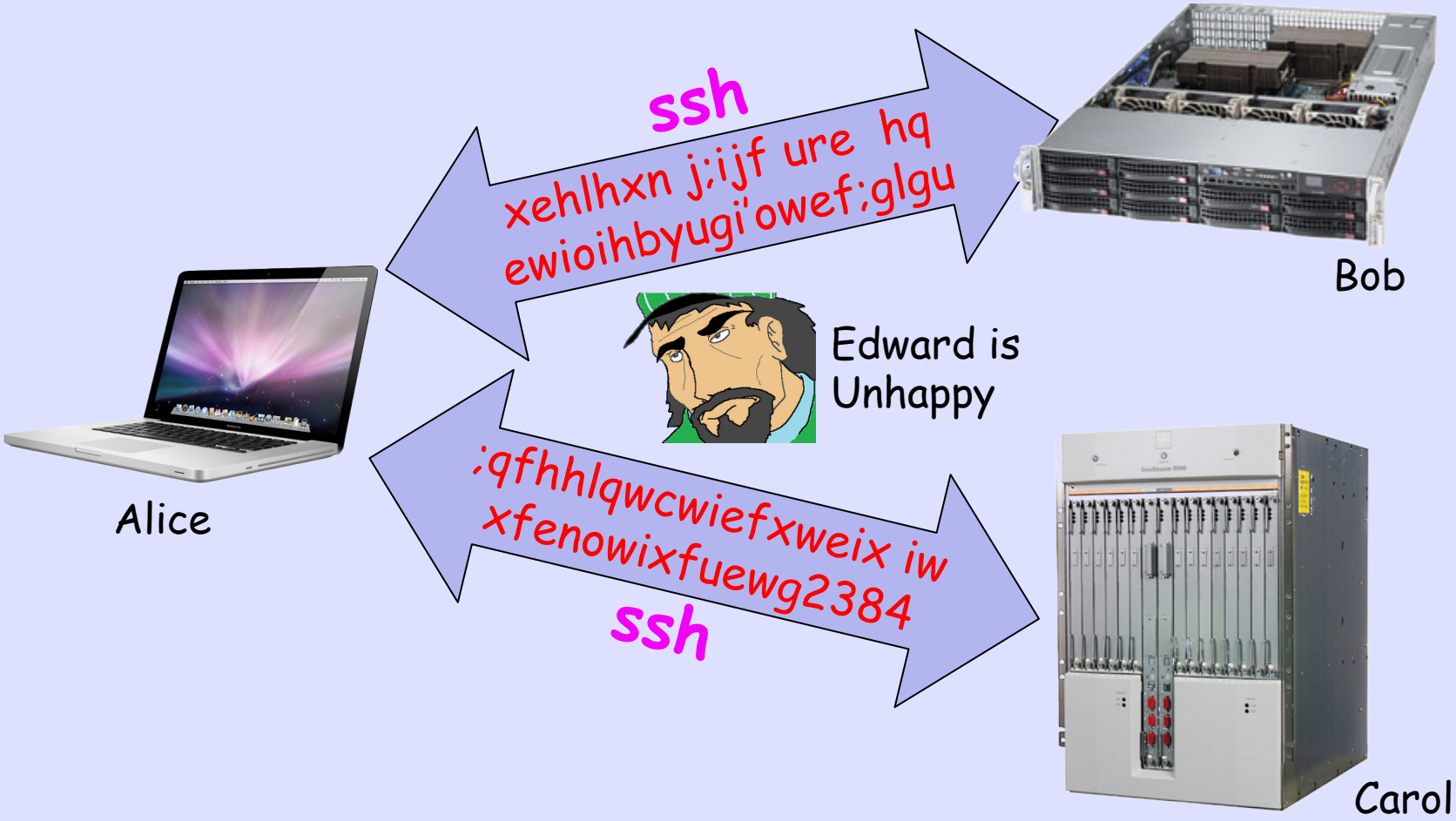
What is "Safely"

- Authentication – I am Assured of Which Host I am Talking With
- Authentication - The Host Knows Who I Am
- The Traffic is Encrypted

Traditional



Encrypted



Secure SHell

- Provides authenticated and encrypted shell access to a remote host
- But it is much more
- It is used by other protocols, sftp, scp, rsync, …
- You can use it to build custom tunnels

Think of SSH as a Bit Like PGP where the Other End is a Computer, Not a Human

But PGP is

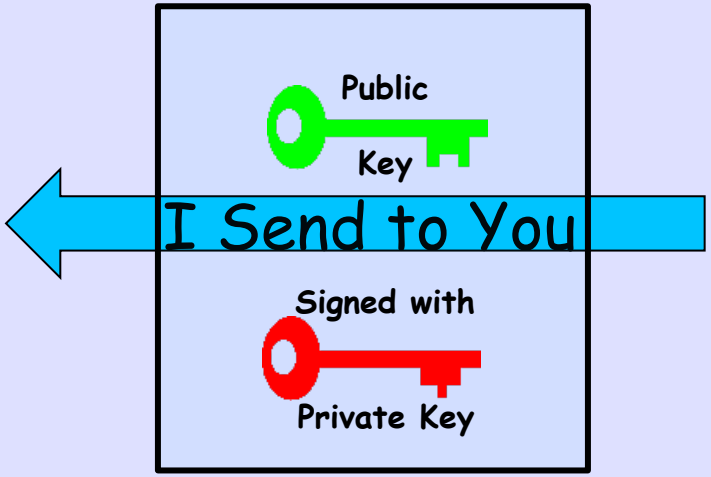
Object Security

SSH is

Channel/Transport Security

If I Have a Key Pair

How Do I Convince You That I Have Both Private and Public Keys Over The Public Net?



You Verify Signature Using My Public Key

If It Verifies, Then You Know That I Must Have The Private Key

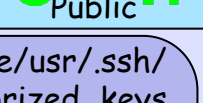
And You Know You Have My Corresponding Public Key

ssh - Keying Setup

ssh-keygen -t rsa

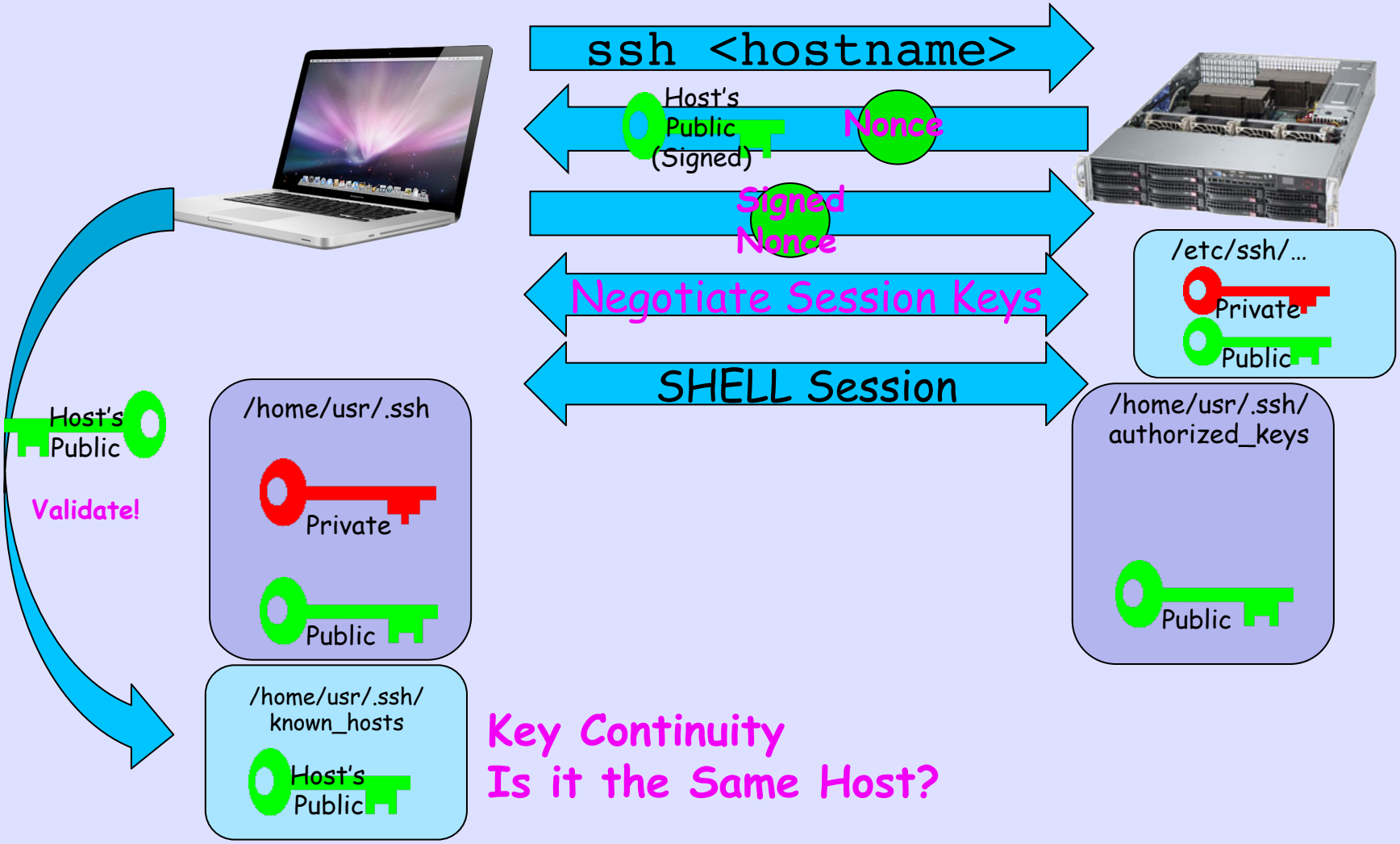
/home/usr/.ssh

/etc/ssh/...



/home/usr/.ssh/
authorized_keys

2-Way Authentication



Key Continuity
Is it the Same Host?

Checking Host's Keys

```
$ ssh -o VisualHostKey=yes psg.com
Host key fingerprint is
d2:2b:f1:17:75:0d:c9:86:74:71:e2:00:62:0f:22:02
+--[ RSA 1024]----+
|E.. . . + .ooo=o.|
|   . . o +  .++= |
|         . ..o .|
|      .   . .   |
|     o S .      |
|      + . .     |
|     . o .      |
|      . .       |
|                |
+-----------------+
```

And you check it against what you got out of band

ssh-keygen RSA Key

```
/usr/home/foo> ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/usr/home/foo/.ssh/id_rsa):
Created directory '/usr/home/foo/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /usr/home/foo/.ssh/id_rsa.
Your public key has been saved in /usr/home/foo/.ssh/id_rsa.pub.
The key fingerprint is:
27:99:35:e4:ab:9b:d8:50:6a:8b:27:08:2f:44:d4:20 foo@psg.com
The key's randomart image is:
+--[ RSA 2048]----+
|E.o       .      |
|.. .     o       |
|.         +      |
| .       + o     |
|.       S o      |
|..     o +       |
|.o .  + .        |
|. o .o.= o       |
| .  .oo +        |
+-----------------+
```

Eliptical Curve Key

```
/usr/home/foo> ssh-keygen -t ecdsa
Generating public/private ecdsa key pair.
Enter file in which to save the key (/usr/home/foo/.ssh/id_ecdsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /usr/home/foo/.ssh/id_ecdsa.
Your public key has been saved in /usr/home/foo/.ssh/id_ecdsa.pub.
The key fingerprint is:
7a:9d:c5:05:5e:39:95:ae:f7:87:0a:43:66:67:2d:45 foo@psg.com
The key's randomart image is:
+--[ECDSA  256]---+
|            . Eoo|
|           . +o. |
|            . +. |
|           . + . |
|        S + * o  |
|       . = = o . |
|      . . =   ...|
|       .   o  . o|
|            ..  .|
+----------------+
```

ssh-keygen - sshv1 key

```
/usr/home/foo> ssh-keygen -t rsa1
Generating public/private rsa1 key pair.
Enter file in which to save the key (/usr/home/foo/.ssh/identity):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /usr/home/foo/.ssh/identity.
Your public key has been saved in /usr/home/foo/.ssh/identity.pub.
The key fingerprint is:
1e:c2:df:cd:54:60:63:24:58:71:1f:ac:36:67:c8:b6 fo@ran.psg.com
The key's randomart image is:
+--[RSA1 2048]----+
|         o+oB..  |
|        .  = +.. |
|           . oo  |
|     .      B.o  |
|      o S  o.=   |
|       + o +E    |
|        o . o    |
|                 |
|                 |
+-----------------+
```

ssh v1 is for 2511s and other antiques

Use Keys Not Passwords

- In `/etc/ssh/sshd_config`
  `PermitRootLogin without-password`
  `UsePAM no`
- Never Store Private Key on a Multi-User Host
- Store Private Key ONLY on Your Laptop and Protect Your Laptop (Encrypt Disk!)
- It is OK to Use SSH_AGENT to Remember your Key ONLY if your Laptop Locks Very Quickly

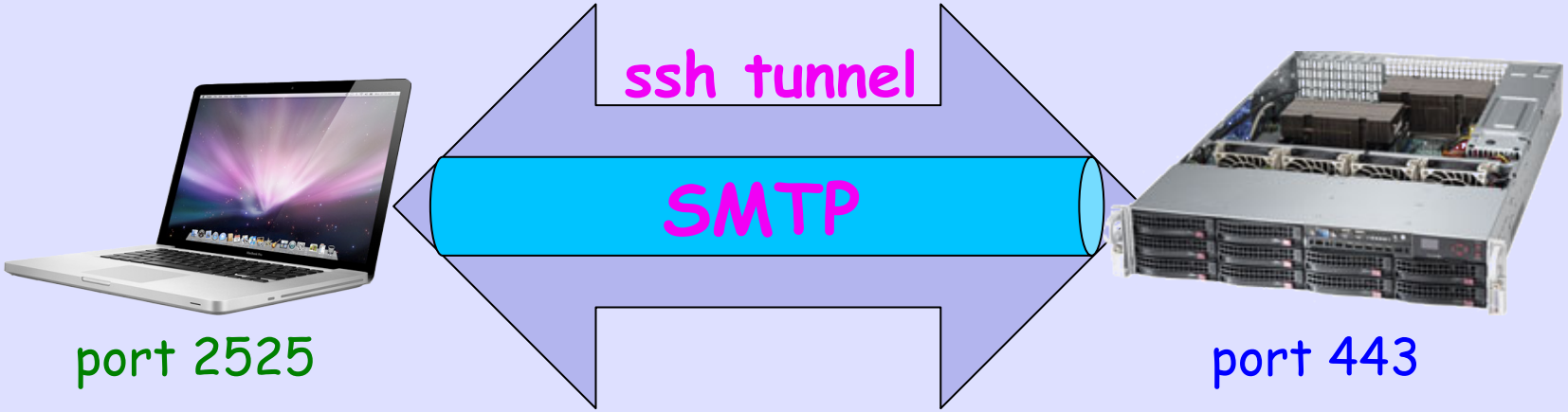
The Only Compromise I Have Had to My Infrastructure was a Researcher who Stored Their Private Key on a Shared University Host

Private Key Protection

- FreeBSD Repository Compromise Two Years Ago

  "The compromise is believed to have occurred due to the leak of an SSH key from a developer who legitimately had access to the machines in question, and was not due to any vulnerability or code exploit within FreeBSD."

General Purpose Tunnel




```
$ ssh -N ssh.psg.com -p 443 -L 2525:127.0.0.1:25
```

Target Host

Tunnel Port

Port on MacBook

Tunnel EndPoint

SSH is Built In

UNIX

Linux

MacOS X

Get Software

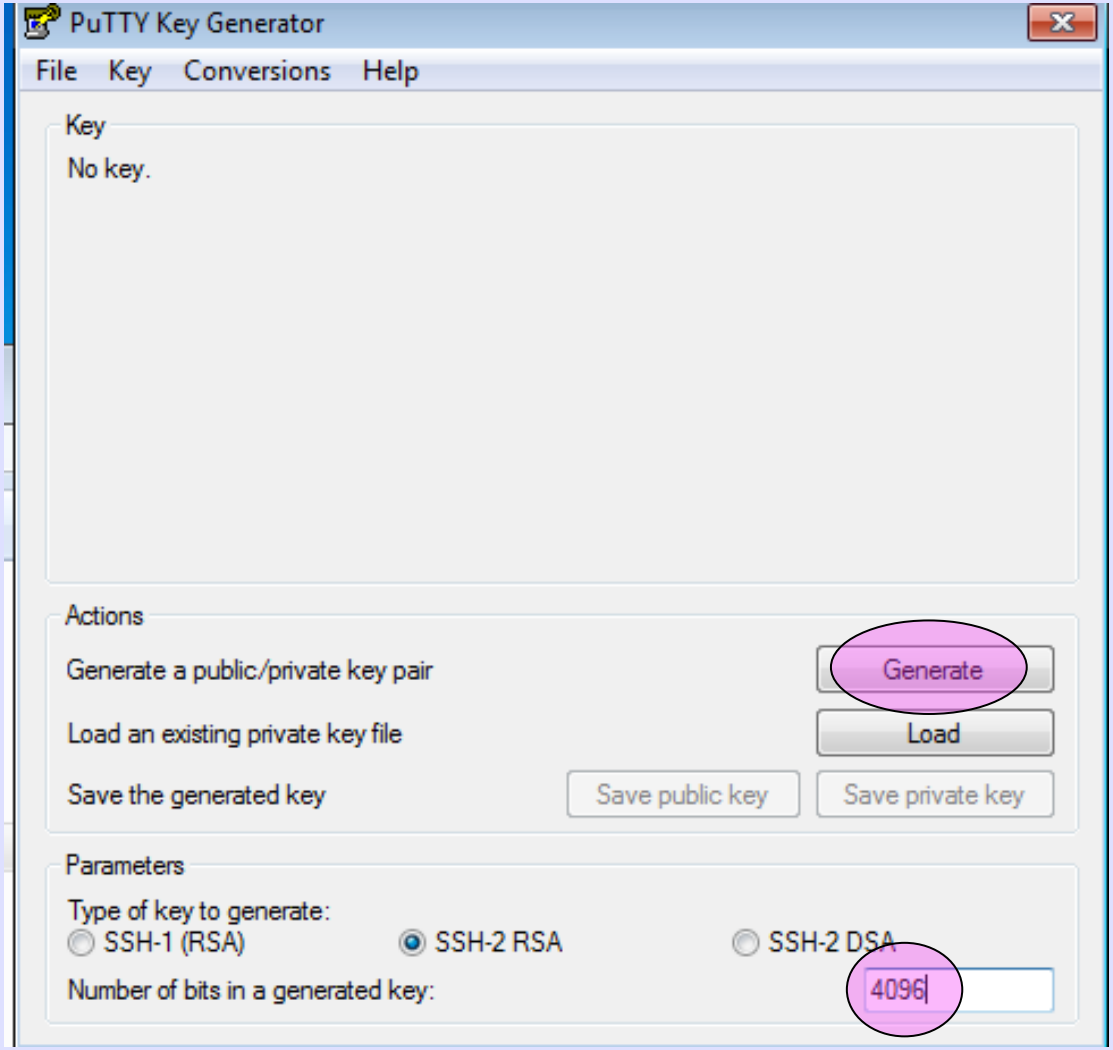
http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html

PuTTY: putty.exe

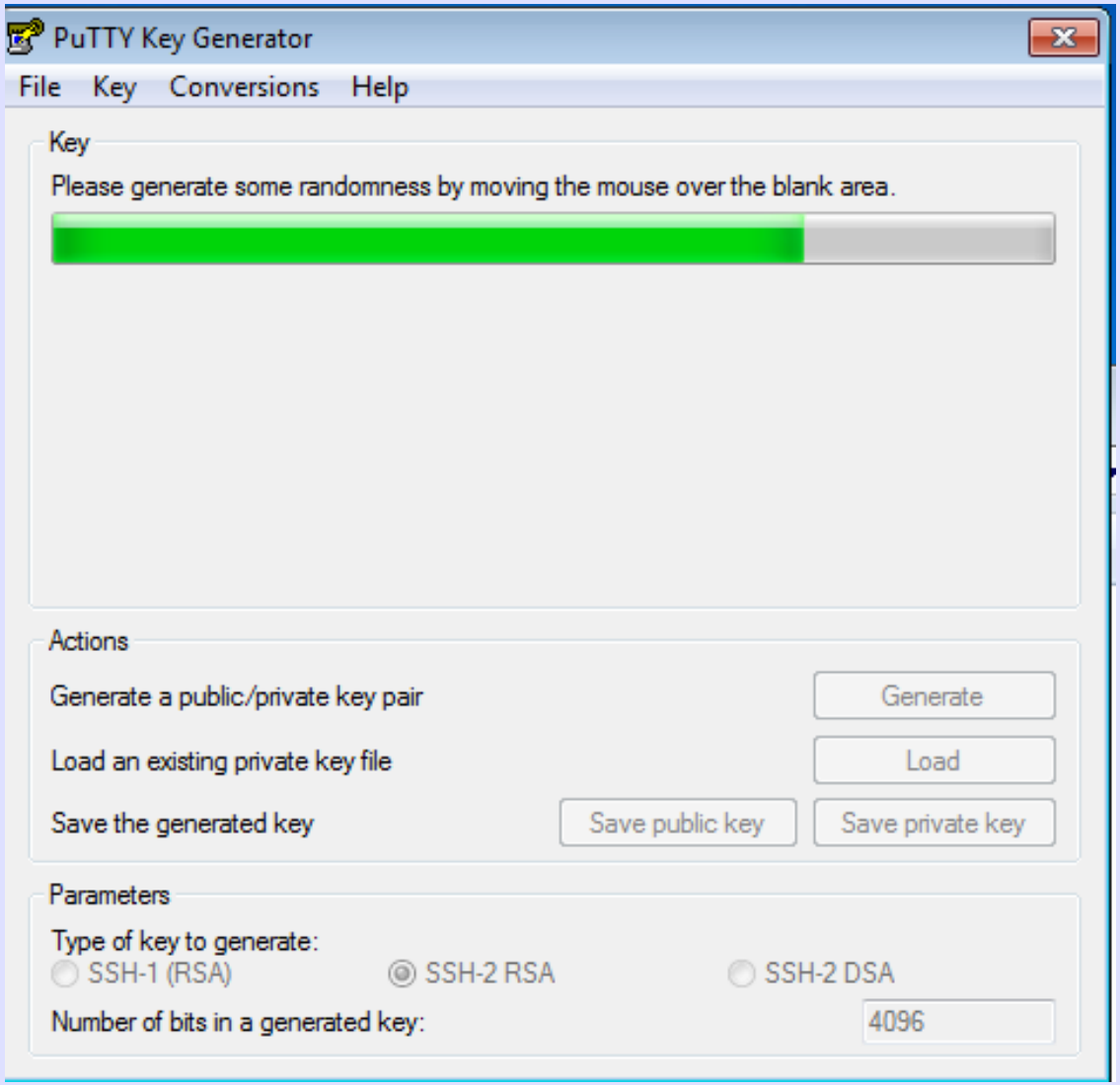
Pageant: pageant.exe

PuTTYgen: puttygen.exe

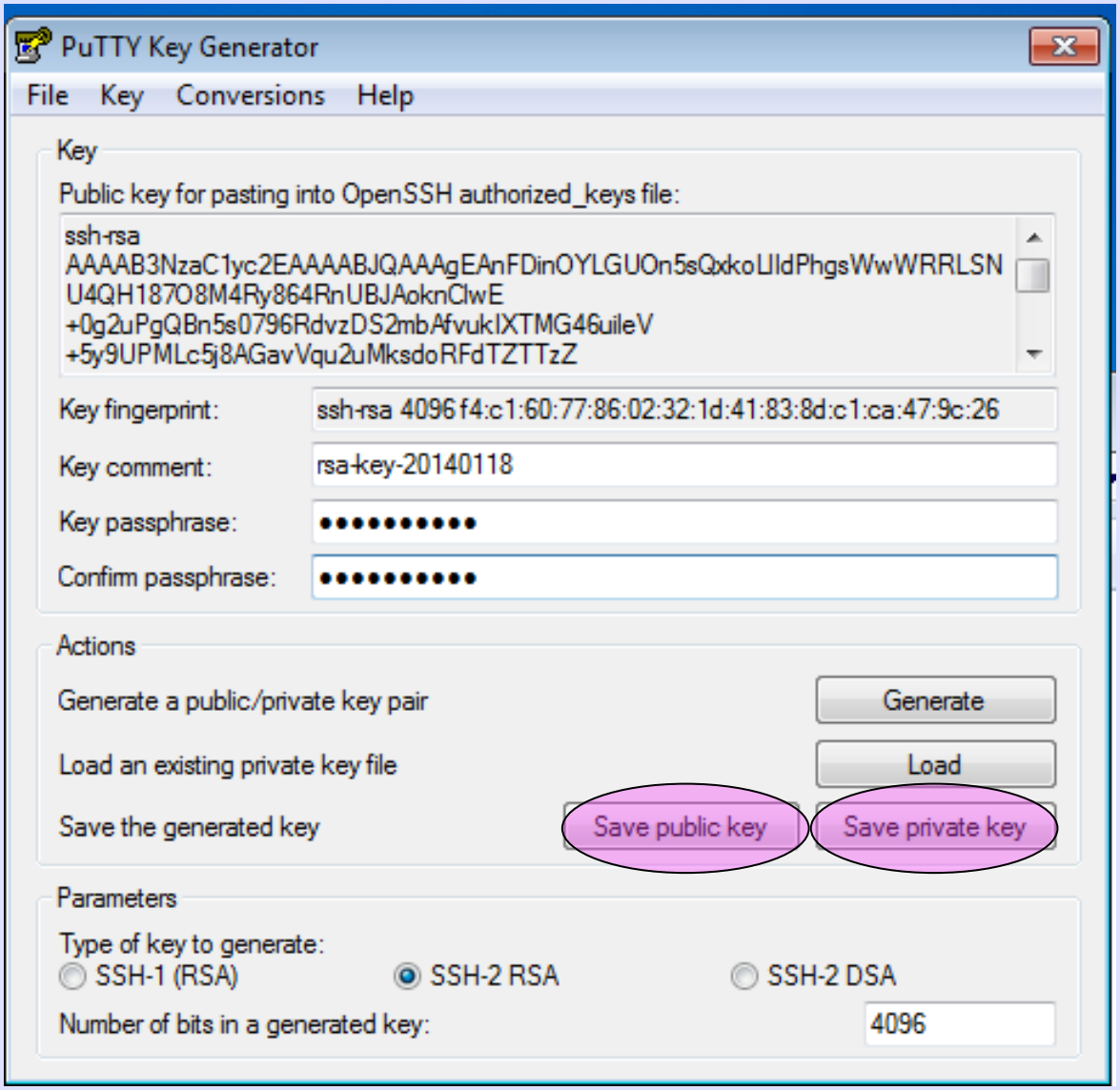
PuttyGen



Generate Key



Enter Passphrase & Save Key



Putting the Key on the Target Host

- Mail the **Public** key to your sysadmin: (randy@psg.com) and he will install it
- He will then create the .ssh directory in your home directory

```
mkdir ~username/.ssh
```

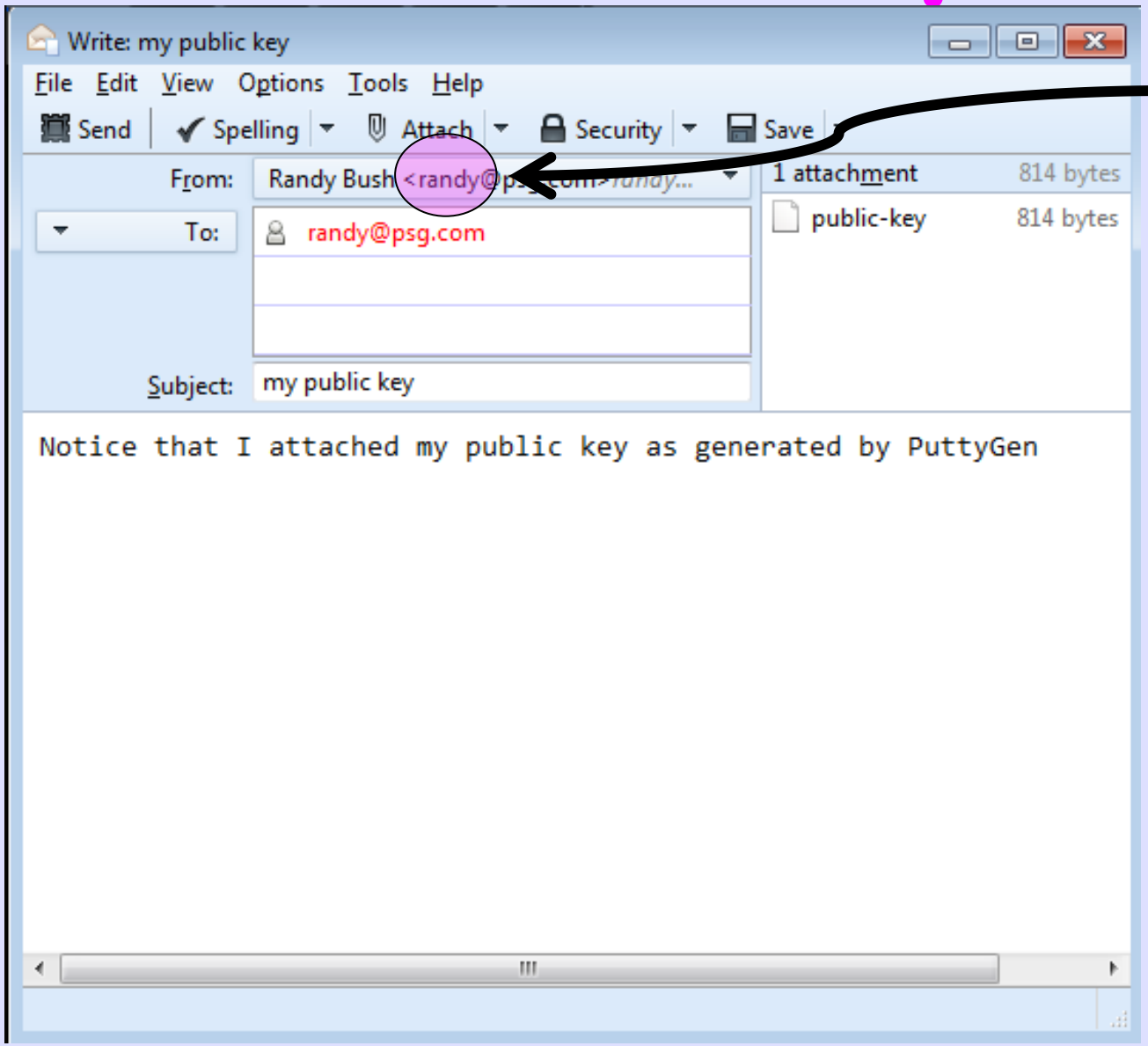
- And put the **public** key in a file called authorized_keys

```
cat id_rsa.pub >> ~username/.ssh/authorized_keys
```

- Permissions have to be non world readable

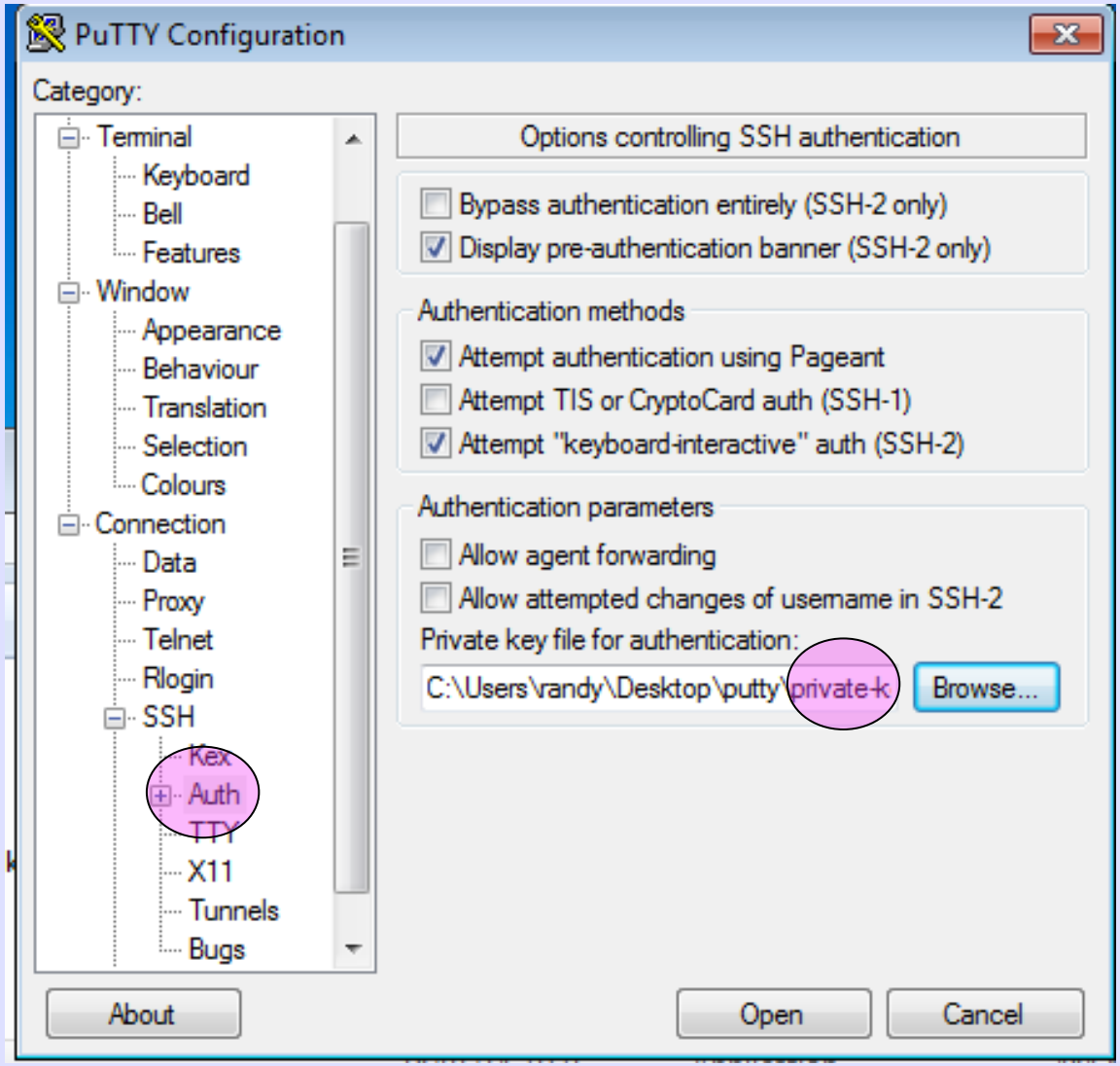
```
chown –R username:guest ~username/.ssh
chmod –R go-rwx ~username/.ssh
```

Mail the Key

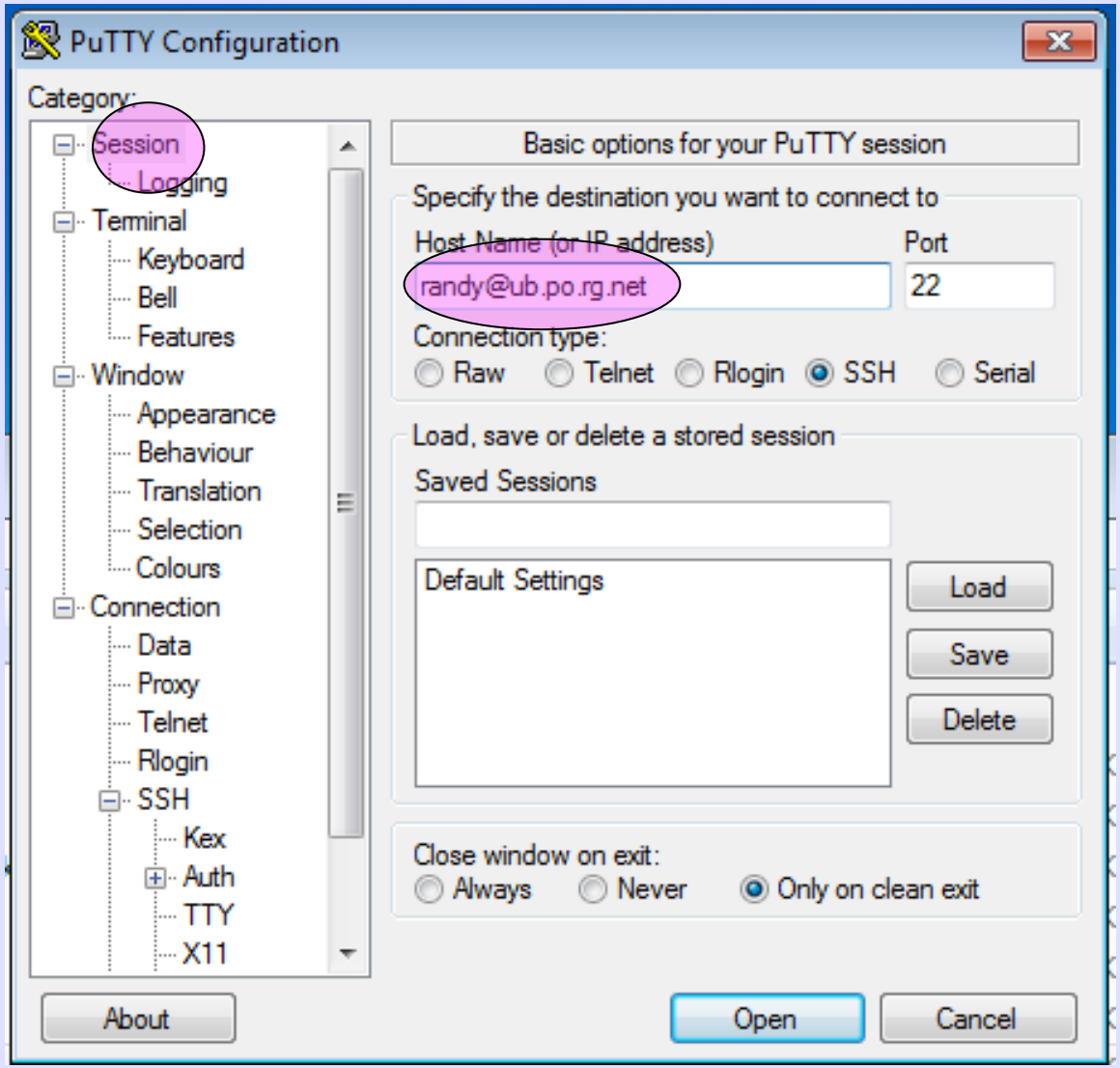


Your User Name

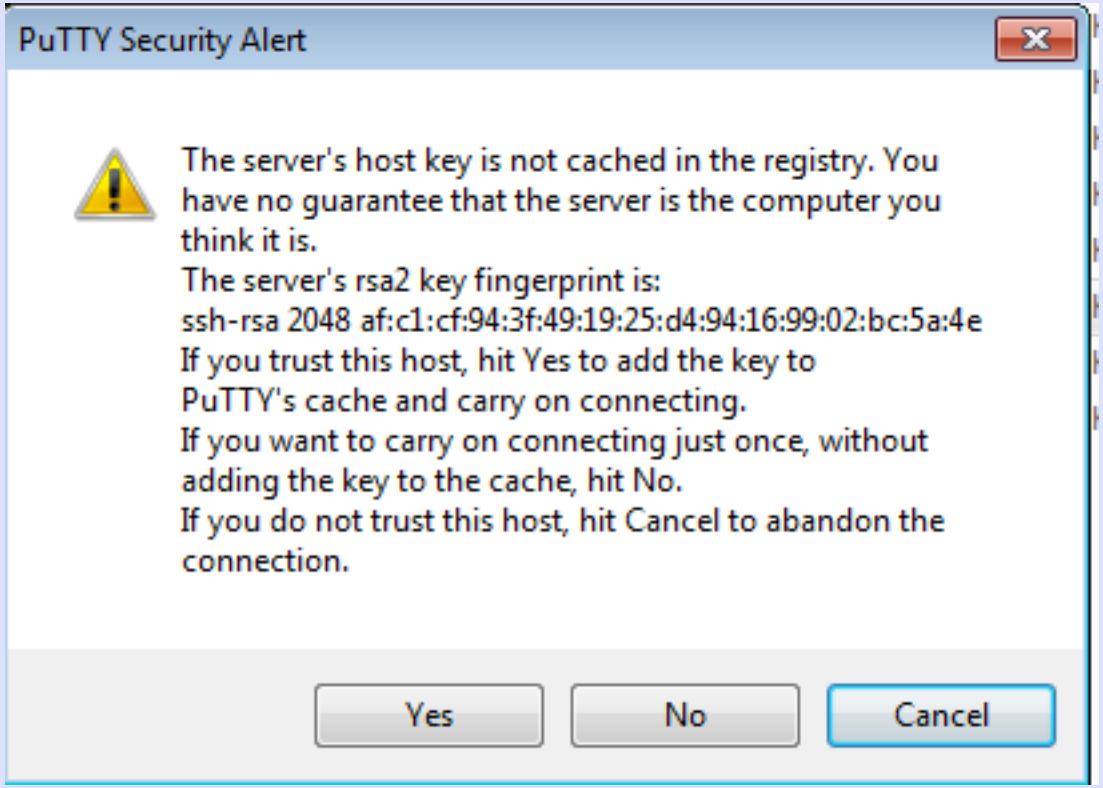
Load Key in Putty



ssh to Host



Accept Host's Key



Passphrase for Key

ub.po.rg.net - PuTTY

```
Using username "randy".
Authenticating with public key "rsa-key-20140118"
Passphrase for key "rsa-key-20140118":
```

You Are In!

randy@ub: ~

```
Using username "randy".
Authenticating with public key "rsa-key-20140118"
Passphrase for key "rsa-key-20140118":
Welcome to Ubuntu 12.04.1 LTS (GNU/Linux 3.2.0-36-virtual i686)

 * Documentation:  https://help.ubuntu.com/

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

randy@ub:~$
```

ssh - Shell Session

```
$ ssh username@ub.po.rg.net
```