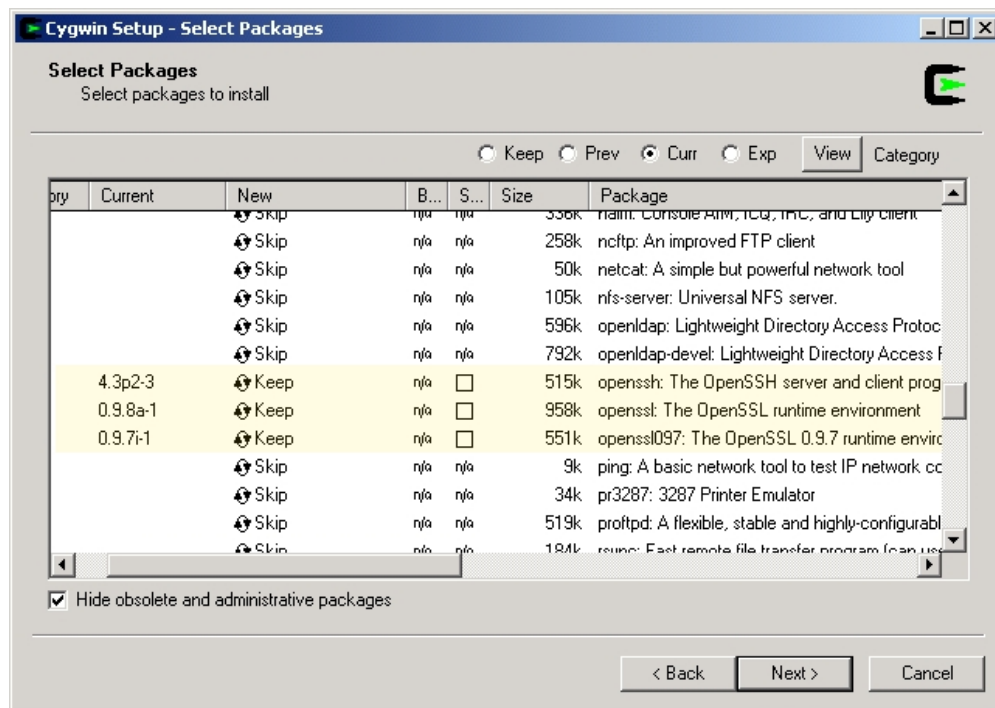


# Using Tramp With NTEmacs and Plink to Connect to OpenSSH

## Installing the tools

I assume you already have emacs running on windows. You will need to have cygwin installed to generate the keys on the local machine.

The first thing to do is grab a copy of cygwin <http://www.cygwin.com/>. When installing cygwin make sure you install the openssh option under net and the autoconfig options under devel:



If you don't have tramp installed you can get a copy from the gnu ftp server:  
<http://ftp.gnu.org/gnu/tramp/>.

From a cygwin shell install tramp. Tramp uses \$prefix/share/emacs/site-lisp to install the elisp files. You need to make sure that \$prefix/share/emacs/site-lisp is a directory that emacs searches.

```
tar -xzvf tramp.tar.gz
cd tramp
./configure --with-contrib --prefix=/cygdrive/c/emacs
make && make install
```

You will also need to download some program from the putty website <http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>. You will need plink.exe, pageant.exe and puttygen.exe. Download these programs to a location in your PATH, C:\windows\system32 or C:\winnt\system32 are good locations.

## ***Generating Keys***

Next we will generate your keys on the local machine. Start a cygwin shell and type:

```
ssh-keygen.exe
```

This will ask you for a passphrase make sure you give it a good one.

In the directory `~/.ssh` you will have some files. The two we are interested in are **id\_rsa** and **id\_rsa.pub**. `id_rsa` is your private key and `id_rsa.pub` is your public key.

## ***Setting up the server***

Now we need to upload your new public key to the server. From your cygwin shell type:

```
sftp example.server.com
```

```
put ~/.ssh/id_rsa.pub
```

```
exit
```

Login to your server to setup the `authorized_keys` file:

```
ssh example.server.com
```

```
mkdir ~/.ssh
```

```
chmod 700 ~/.ssh
```

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

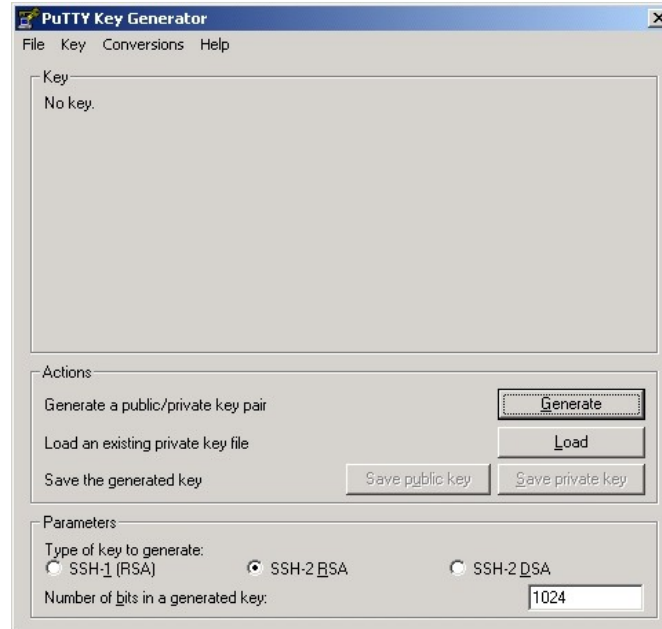
```
chmod 600 ~/.ssh/authorized_keys
```

```
rm ~/id_rsa.pub
```

Your server is now setup to work with your new public key.

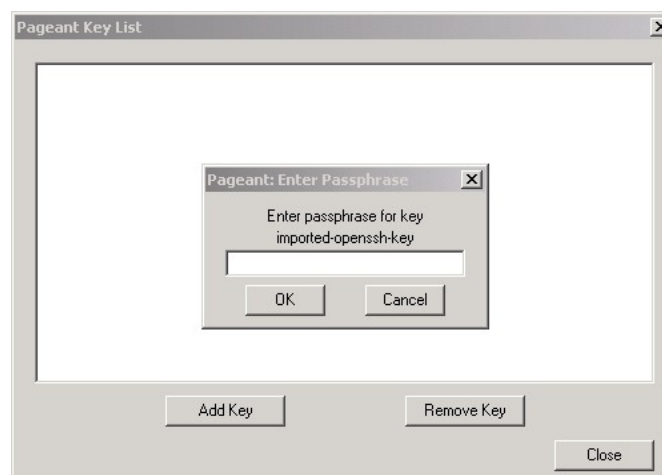
## Setting up the client

You will need to convert your private key from OpenSSH's format to Putty's format. To do this start puttygen.exe:



From the conversions menu click on import key, go to your cygwin home directory and pick the file .ssh/id\_rsa. It will ask you to enter your passphrase. Once the key has been imported click on “Save private key” and put the key somewhere safe.

Close puttygen and start pageant, click on add key and pick the key file that you imported using puttygen. It will ask you to enter your passphrase. Close pageant by clicking the x, this will put it in the system tray.



## **Configuring emacs**

Add the following lines to your `_emacs` file:

```
(require 'tramp)  
(setq tramp-default-method "plink")
```

If tramp was installed to a location that is not in your path you can add it with the following line:

```
(add-to-list 'load-path "C:/emacs/share/emacs/site-lisp/")
```

Now you can use C-x C-f to open a remote file. Ex

```
/jhe@example.server.com:/home/jhe/lisp/music-theory.lisp
```

If emacs tells you it can't find **file-remote-p** when you try to save a file you can add the definition for it to your `_emacs` file:

```
(unless (fboundp 'file-remote-p) ;; emacs 21.4 on debian at least,  
      ;; doesn't provide file-remote-p  
  (defun file-remote-p (file) ;; stolen from files.el  
    "Test whether FILE specifies a location on a remote system.  
    Return an identification of the system if the location is indeed  
    remote. The identification of the system may comprise a method  
    to access the system and its hostname, amongst other things.  
    For example, the filename `"/user@host:/foo` specifies a location  
    on the system `"/user@host:`.`"  
    (let ((handler (find-file-name-handler file 'file-remote-p)))  
      (if handler  
        (funcall handler 'file-remote-p file)  
        nil))))
```

- suggestions welcome [jeremy.english@gmail.com](mailto:jeremy.english@gmail.com)