

Working from home on a PC

Introduction

Fun fact: It is possible to remotely log in to the CS department's Linux machines from anywhere, using a system called ssh. This means that, using only your ingenuity and the technology at your disposal, you can log into one of the Sun Lab machines and run commands in the shell from your computer in your dorm room.

Why would I want to do that? Well, it's pretty cool. And you'll be able to do your homework assignments and hand them in without leaving your room to come to the CIT. All you have to do is follow these slightly complicated instructions.

This guide will show you how to:

- Set up remote log in using PuTTY, so you can ssh into a Sun Lab machine.
- Set up X forwarding using Cygwin, so you can use programs that have GUIs.
- Set up file transfers using WinSCP, so you can move files between your computer and the CS filesystem.

Let's get started.

1 Remote log in

1.1 1.1 Download the software

You'll need two programs to log in remotely. They are:

- PuTTY this is an ssh client. It will allow you to log in to the department machines. You can download it here:
<http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>
- PuTTYgen this allows you to create a key that you will need to authenticate to the system and use PuTTY. Conveniently, it can also be downloaded at the above location.
- WinSCP this will allow you to transfer files from your home computer to the CS filesystem. It can be found here: <http://winscp.net/eng/download.php> (you want WinSCP 4.2.8 Installation Package)

PuTTY and PuTTYgen are standalone executables that do not need to be installed. Once you've downloaded them, they're ready to use. You'll need to install WinSCP, but we'll get to that later.

1.2 Create a key

1. Create a folder somewhere on your computer called "puttykeys".
2. Open PuTTYgen.
3. In the "Parameters" subsection (at the bottom of the window) choose the "SSH-2 DSA" option.
4. Time to make some keys! Press the "Generate" button.
5. You'll be asked to generate some randomness by moving the mouse around the blank area in the "Key" section. Do it. As the progress bar fills in, your mouse movements will be translated into a random key.
6. Eventually, the program will have enough input to generate a key. Enter a passphrase in the "Key passphrase" field, and repeat it in the "Confirm passphrase" field. This can be anything, but remember to make it memorable and secure! You can also give the key a name in the "Key comment" field.
7. Press the "Save public key" button. Save it as "csfs.pub" in the puttykeys folder you created.
8. Now press the "Save private key" button. Save it in the same folder as "csfs.ppk" (PuTTYgen should add the .ppk file extension automatically).
9. You'll need to bring the public key file (csfs.pub) to the Sun Lab. Recommendations: email it to yourself or bring it on a flash drive.

1.3 1.3 Meanwhile, at the Sun Lab...

1. Open a terminal shell.
2. Put csfs.pub in your home directory.
3. Type `ssh-key-setup csfs.pub`
4. Your work is done here. You can leave the Sun Lab.

Good work so far. Isn't this fun?

1.4 Set up PuTTY

1. Back on your home computer, open PuTTY.
2. In the “Host Name (or IP address)” box, enter `ssh.cs.brown.edu`
3. In the “Connection type” subsection, make sure “SSH” is selected.
4. Click on the minus sign next to “SSH” in the Category menu to expand this category. Select “Auth”.
5. Make sure the “Attempt keyboard-interactive auth” box is checked.
6. Press the “Browse” button. Find where you saved your private key (it should be named “`csfs.ppk`”) and select it.
7. Select “X11” in the “SSH” section of the Category menu. Check the “Enable X11 forwarding” box.
8. Now select “Session” in the “Category” menu to return to the main screen. Type “Brown CS” in the “Saved Sessions” box, and press the “Save” button to store these settings.

1.5 Using PuTTY

1. From now on, when you open PuTTY, you should see the “Brown CS” label in the “Saved Sessions” section. Double click on this label.
2. PuTTY will ask you to log in. Use your CS login and the password you chose in PuTTYgen.
3. You should see something like `cs1ab3b /u/login %`. This means that you are logged into the `cs1ab3b` machine in the Sun Lab. Crazy, right?
4. You can now do anything you could do in the shell if you were physically in the Sun Lab, with the notable exception of running programs that have GUIs (like DrRacket, Eclipse, gedit).

2 X forwarding

Linux programs that have GUIs are written according to the X display protocol. In order to run these programs, you’ll need an X server, which receives display information over your ssh connection and displays them on your computer. First we will download Cygwin, a Linux-like environment for Windows which includes an X server.

2.1 Download Cygwin

1. Go to <http://www.cygwin.com> and click on "Install Cygwin now" to download and run setup.exe.
2. Click next at the title screen, and choose "Install from Internet".
3. The default settings are fine on the next few screens (Installation Directory, Local Package Directory, Connection Type, Download Site). You can change them if it matters to you.
4. After the setup files download from the mirror chosen on the Download Site screen, you'll see a long list of packages. Scroll to the very bottom to the "X11" package, and click on the arrow symbol next to it until it reads "Install" instead of "Default". You can browse through the other packages to see what else installs by default and make any changes you like. The only important thing is that you get the entire X11 package. Click next.
5. You may see a screen warning you about unmet dependencies. If so, keep "Install these packages to meet dependencies" checked, and click next.
6. The selected packages will now download and install. This may take a while.
7. You're done installing! You don't need to create desktop or start menu shortcuts you won't actually need to use Cygwin, just Cygwin's X server.

2.2 Set up the X server

1. Now navigate to `C:\cygwin\bin\`, and find the file called `startxwin.bat`. This is a batch script which starts the X server.
2. However, it also opens an xterm, which we don't need. To remove this functionality, right click on the file and choose "Edit." The script's code should open in a text editor.
3. Scroll down almost to the bottom and find the line that reads `%RUN% xterm -e /usr/bin/bash -l`. Add "REM" to the beginning of this line so that it reads `REM %RUN% xterm -e /usr/bin/bash -l`. Then save and close the editor.
4. Right click on `startxwin.bat` and choose "Create Shortcut". Put this shortcut somewhere handy.

2.3 Using the X server

1. When you run `startxwin.bat` via the shortcut you created, an X symbol should show up in the notification area of your taskbar. Working from home on a PC This means that the X server is running.

2. Now open PuTTY and log in. Try running a program like gedit or Eclipse. The program should start and function as usual. You may notice a lag, especially if you are working from off campus.

3 File Transfer

Using PuTTY and X forwarding, you can log in remotely and run programs, but what if you want to transfer files between your computer and the CS filesystem? To do this you'll need a program called WinSCP. This means that the X server is running. It can be found here: <http://winscp.net/eng/download.php> (you want WinSCP 4.2.8 Installation Package).

3.1 Set up WinSCP

1. Download and open the WinSCP installer.
2. Press next twice to accept the License Agreement. Choose "Custom installation".
3. Accept the default destination (or change it, if you really want to).
4. At the "Select Components" screen, uncheck everything except "WinSCP application".
5. Do what you like for the next two screens, but make sure you leave "Register to handle sftp:// and scp:// addresses" checked on the "Select Additional Tasks" screen.
6. Choose the "Norton Commander" interface. Press "Install".
7. Open WinSCP.
8. In the "Host name" box, type "ssh.cs.brown.edu". Enter your login in the "User name" box, but leave the "Password" box blank. Press the "..." button next to the "Private key file" box, and find your private key (csfs.ppk). Make sure the File protocol subsection is SFTP, and that the "Allow SCP fallback" box is checked.
9. Press the Save button. Accept the default session name (yourlogin@ssh.cs.brown.edu).

3.2 Using WinSCP

1. From now on, when you open PuTTY, you should see the "yourlogin@ssh.cs.brown.edu" label in the "Saved Sessions" section. Double click on this label.
2. You'll be asked for a password use the one you chose in PuTTYGen for your private key. The first time you log in you may receive a warning about accepting and caching the key.

3. You'll see some status messages, and then the WinSCP interface should open. The left side of the screen represents your computer, and the right side represents the CS Filesystem.
4. You can navigate around both file systems, and drag and drop files between them as necessary. Congratulations, you can now work effectively from home! Wasn't that fun?