

Remote Access to Unix Machines

*Alvin R. Lebeck
Department of Computer Science
Department of Electrical and Computer Engineering
Duke University*

Overview

We are using OIT Linux machines for some homework this semester. Therefore, you need to ensure your account is activated, that you can remotely access the machines, and that you become familiar with using Linux to create files, move around the directory (folder) structure, etc. This tutorial is aimed at getting you up to speed on these aspects early, so you can hit the ground running.

Machine Names

The machines we will use are named login-teer-`<##>.oit.duke.edu` (where `<##>` is 01-45) and are accessible either by directly logging into the machines in the Teer building, or more commonly using remote access using `login.oit.duke.edu` (see below). The machine names are `login-teer-01.oit.duke.edu`, `login-teer-02.oit.duke.edu`, ... `login-teer-45.oit.duke.edu`. Note that the machines 1-9 require the leading 0.

Remote Access to Machines

A convenient way to access a linux machine is remotely over the Internet. The main item necessary for this is a secure shell client, but we will also utilize X-windows so we can run programs with graphical interfaces (GUIs). We will use `login.oit.duke.edu` to remotely connect to one of the teer machines: `login.oit.duke.edu` is a front-end alias that dynamically selects one of the physical teer machines to connect to.

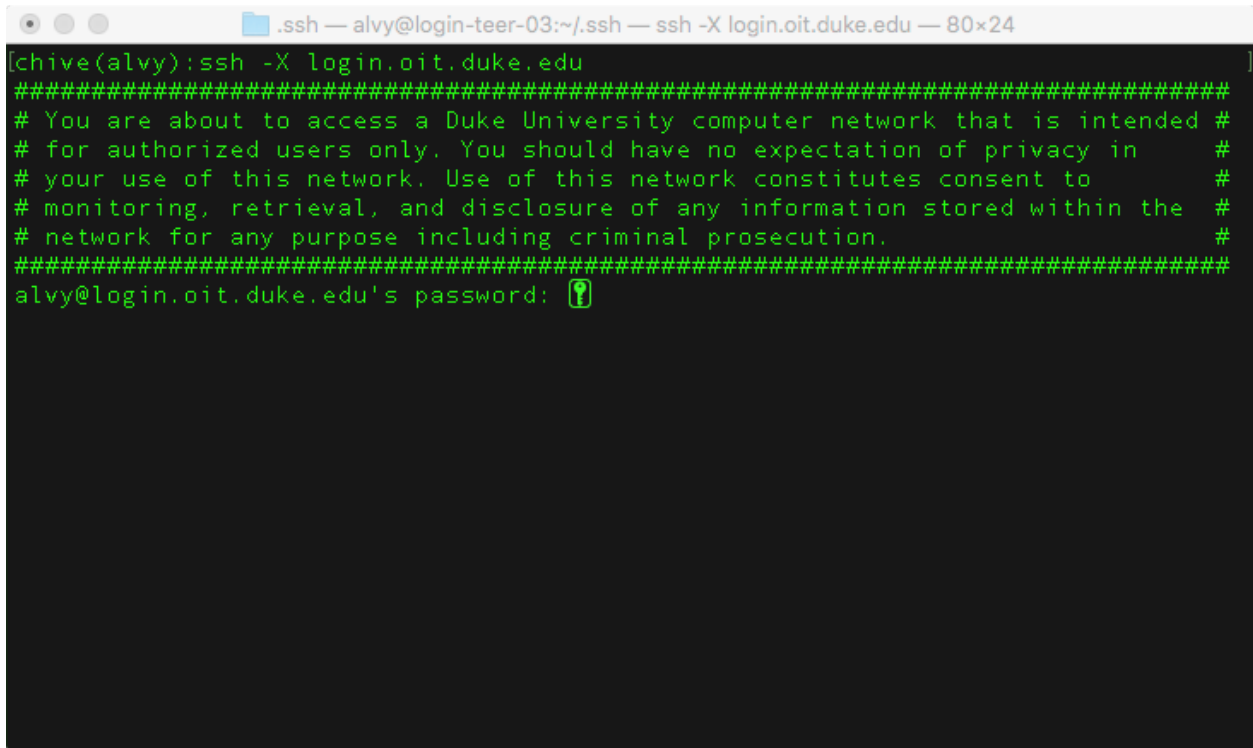
MAC OS

1. Download and install Xquartz from <http://xquartz.macosforge.org>.
2. Logout of your MAC
3. Login to your MAC

X-windows is now ready for use on your MAC. Next up is the secure shell client, on a MAC this is easy since it has a shell built in via the Terminal Application.

4. Open the Terminal App. You can find Terminal in the Applications/Utilities folder or by searching in Spotlight for Terminal.

5. At the command prompt type `ssh -X netID@login.oit.duke.edu` where *netID* is your Duke *netID*. This command initiates a secure shell connection to one of the teer machines. If this is the first connection that you've made to a particular machine you will receive a warning about the authenticity of the host and RSA key fingerprint. This is expected behavior for first-time connections. If you typed the machine name correctly, you can just type *yes* and the appropriate information will be added to your local machine, and you shouldn't see the error again on subsequent connections to this same machine.



```
.ssh — alvy@login-teer-03:~/ssh — ssh -X login.oit.duke.edu — 80x24
[chive(alvy):ssh -X login.oit.duke.edu
#####
# You are about to access a Duke University computer network that is intended #
# for authorized users only. You should have no expectation of privacy in      #
# your use of this network. Use of this network constitutes consent to        #
# monitoring, retrieval, and disclosure of any information stored within the    #
# network for any purpose including criminal prosecution.                       #
#####
alvy@login.oit.duke.edu's password: [?]
```

6. Enter your password. Note the characters you type are not displayed on your screen, neither are any `***` corresponding to the characters. So, if login fails you may have to try again.

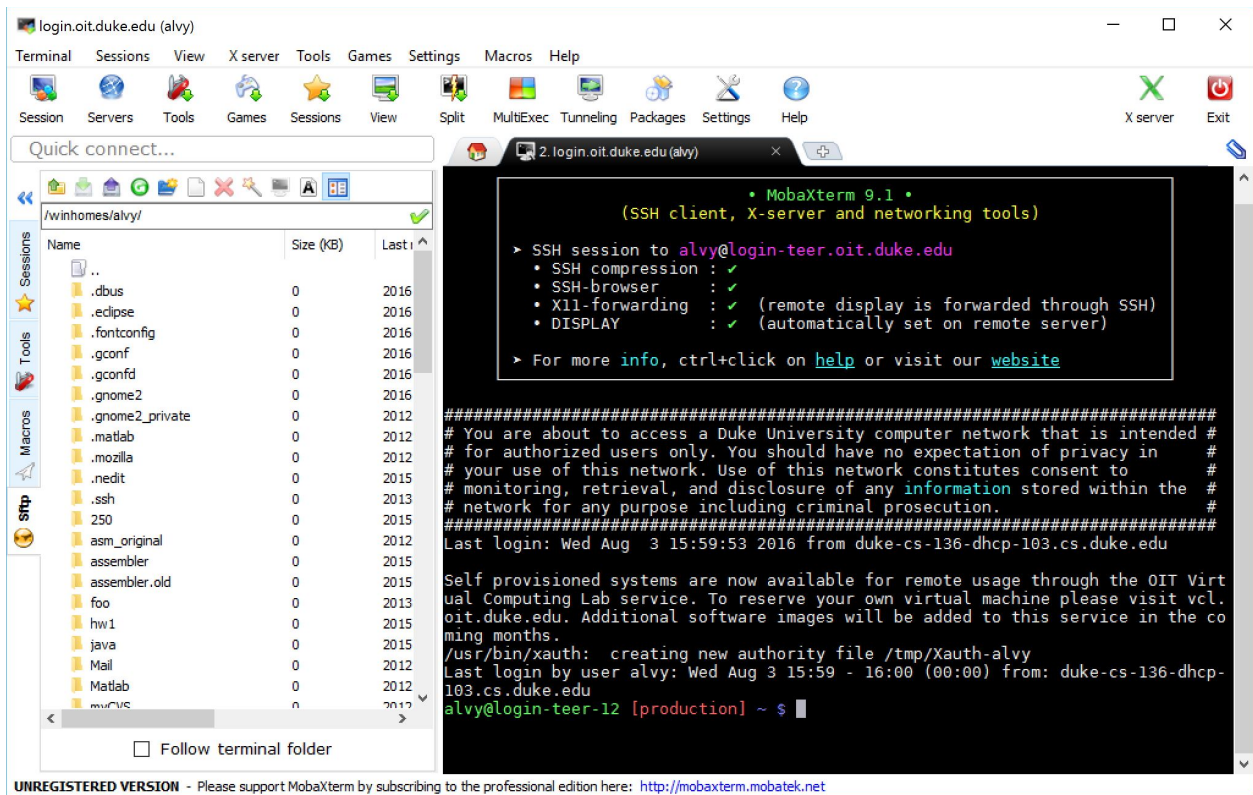
```
.ssh — alvy@login-teer-15:~ — ssh -X login.oit.duke.edu — 80x24
[chive(alvy):ssh -X login.oit.duke.edu
#####
# You are about to access a Duke University computer network that is intended #
# for authorized users only. You should have no expectation of privacy in     #
# your use of this network. Use of this network constitutes consent to       #
# monitoring, retrieval, and disclosure of any information stored within the   #
# network for any purpose including criminal prosecution.                     #
#####
[alvy@login.oit.duke.edu's password:
Last login: Mon Aug 22 09:33:31 2016 from duke-cs-136-dhcp-103.cs.duke.edu

Self provisioned systems are now available for remote usage through the OIT Virtual Computing Lab service. To reserve your own virtual machine please visit vcl.oit.duke.edu. Additional software images will be added to this service in the coming months.
/usr/bin/xauth: file /tmp/Xauth-alvy does not exist
alvy@login-teer-15 [production] ~ $
```

Ta-dah! You are now successfully connected to a remote machine over the Internet. Goto the “At the remote machine command prompt” section after the Windows information below.

Windows

- 1. Download and install MobaXterm at <http://mobaxterm.mobatek.net/>
- 2. Start MobaXterm (i.e., run the program)
- 3. Click *Session* in the upper left of the mob MobaXterm window
 - a. You should get a popup window with choices across the top for type of session. Choose SSH.
 - b. For Remote host enter login.oit.duke.edu
 - c. Click the specify username checkbox, and enter your Duke NetID for the username.
 - d. Click OK. This creates a session for later use and also connects you to one of the teer machines, with a prompt for your password in the shell window.
- 4. Type your password at the prompt. Note the characters you type are not displayed on your screen, neither are any *** corresponding to the characters. So, if login fails you may have to try again.



5. The next time you start MobaXterm, you can just double click the login.oit.duke.edu session in the saved sessions tab to connect to a Duke Linux machine.

Ta-dah! You are now successfully connected to a remote Linux machine over the Internet. I'll leave you to explore the capabilities of MobaXterm on your own.

Linux

If you are using Linux then you probably don't need these instructions, but just in case... Simply start at step 4 of the Mac instructions.

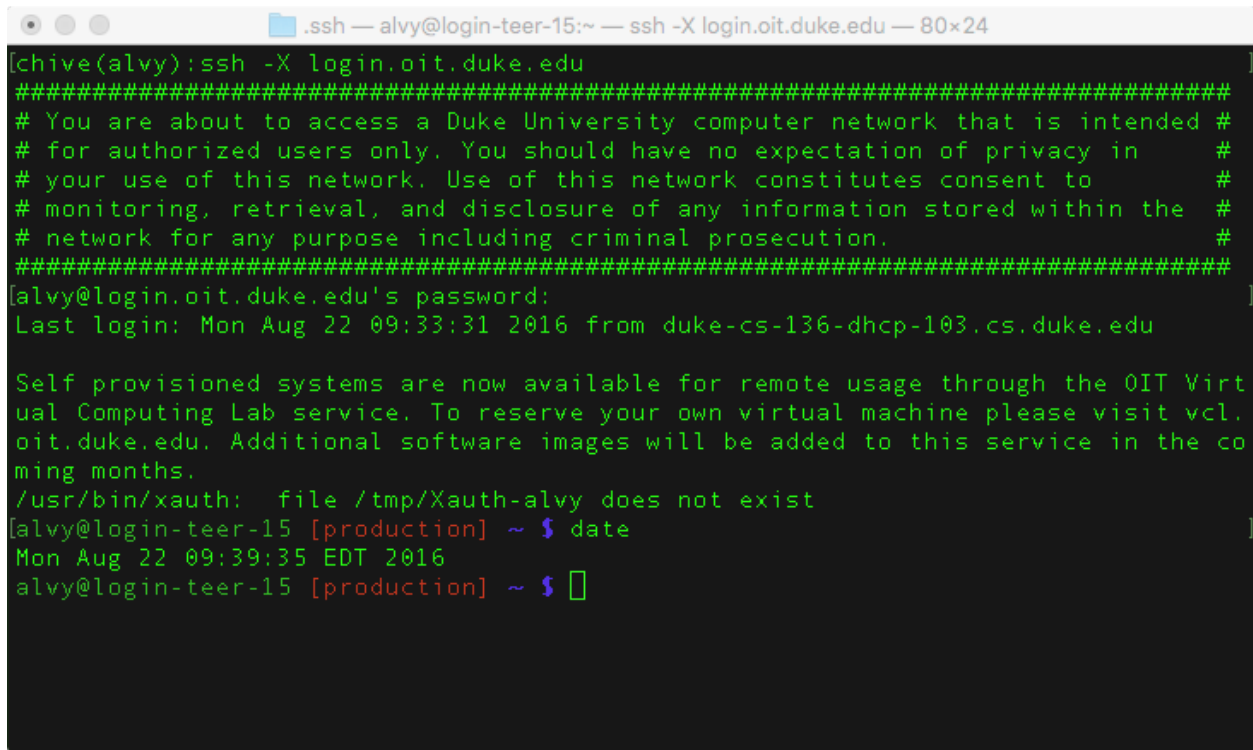
1. Open a Terminal
2. Type `SSH -X netID@login.oit.duke.edu` at the Linux prompt

At the remote machine command prompt

You now have a terminal session that is connected to one of the Duke Linux machines (e.g., login-teer-15.oit.duke.edu).

You should see a command prompt that is something like "`netID@login-teer-15 [production] ~ $`" with a blinking cursor after the \$. Note that you may have a slightly different prompt, e.g., with a > or % instead of the \$, not a big deal... look for the blinking cursor. Your specific prompt may differ from the screen grabs below due to some customizations I've added.

1. Type `date` at this prompt, and press return, this is a unix command to display today's date, as shown below.



```
.ssh — alvy@login-teer-15:~ — ssh -X login.oit.duke.edu — 80x24
[chive(alvy):ssh -X login.oit.duke.edu]
#####
# You are about to access a Duke University computer network that is intended #
# for authorized users only. You should have no expectation of privacy in    #
# your use of this network. Use of this network constitutes consent to      #
# monitoring, retrieval, and disclosure of any information stored within the  #
# network for any purpose including criminal prosecution.                    #
#####
[alvy@login.oit.duke.edu's password: ]
Last login: Mon Aug 22 09:33:31 2016 from duke-cs-136-dhcp-103.cs.duke.edu

Self provisioned systems are now available for remote usage through the OIT Virtual Computing Lab service. To reserve your own virtual machine please visit vcl.oit.duke.edu. Additional software images will be added to this service in the coming months.

/usr/bin/xauth: file /tmp/Xauth-alvy does not exist
[alvy@login-teer-15 [production] ~ $ date]
Mon Aug 22 09:39:35 EDT 2016
[alvy@login-teer-15 [production] ~ $ ]
```

2. Type `xterm` at the prompt. This should open a window on your machine's screen that gives you another terminal on the remote machine. Note you may need to tab between windows since it may not automatically come to the front of your desktop. If this fails, you've done something wrong in setting up X-windows, please review the steps above according to your operating system.

[production] ~ \$ `xterm`



3. Click in the xterm window to activate it. Close the new xterm window by typing `exit` at the prompt (`[production] ~ $ exit`) within the xterm window or by clicking the close button of the xterm window.
4. Background jobs are often useful since they allow you to use multiple windows simultaneously. For example, let's say you want to have two xterms running and use both along with the original session window. To achieve this you can type:
`[production] ~ $ xterm&`
`[production] ~ $ $ xterm&`
The `&` tells unix to run the specified program in the background (in this case it is `xterm`). Now you can click around and type `date` in each of the three windows
5. Close the two xterm windows (type `exit` in them or click the close button).
6. Type `logout` at the prompt in your initial session window to end the SSH connection and disconnect from the remote machine (`[production] ~ $ logout`).

Unix Tutorial

You are now ready to go through the Unix tutorial available at <http://www.cs.duke.edu/~alvy/courses/unixtut>.