



SSH and AFS

Doing your work on other people's computers

Check in for credit:

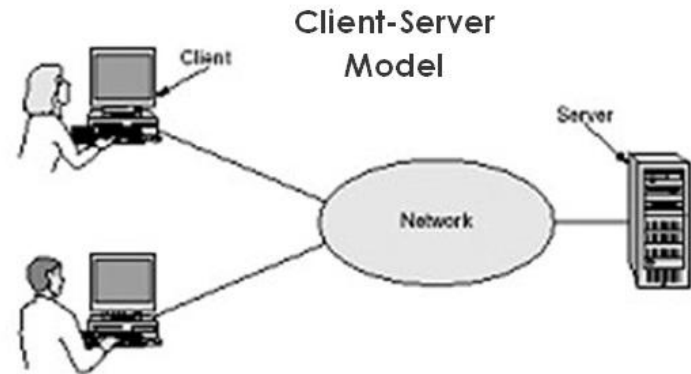
<http://tinyurl.com/extratation1>



Servers

What is a server?

A server is a computer running software that allows it to be accessed over a network connection, typically the internet or a local intranet.





Why are servers useful?

- Might be more powerful than a laptop
- Might have specialized hardware (GPUs) or software (CO, SML)
- Provide a consistent environment that can be accessed from anywhere



CMU Andrew servers

5 (unix4 - unix8) different servers running Ubuntu Linux with the cs.cmu.edu and andrew.cmu.edu AFS cells mounted as storage.

Provide a consistent computing environment for all students.

Accessible over the internet at `unix{4-8}.andrew.cmu.edu`

SSH



What is SSH?

SSH is a protocol for providing a “secure shell” on a server. It lets you access bash on a server over the internet and input commands.

OpenSSH is an implementation of SSH. `sshd` is the server process and `ssh` is the client process. `scp` allowing file copying over SSH.



Wait, so why can't I SCP from Andrew without exiting first?

- Your computer probably doesn't have a static IP or domain name, so SSH can't find it
- Also, you're probably not running sshd.
 - If you are, you probably don't want to be



Authentication

- SSH supports authentication using
 - UNIX username / password (the default)
 - SSH keys (disabled on Andrew, but useful for Github)
 - Kerberos (used by Andrew)



Kerberos authentication with AFS (Linux)

```
sudo apt install krb5-user # Set domain to ANDREW.CMU.EDU  
kinit YOUR_ANDREW_ID
```



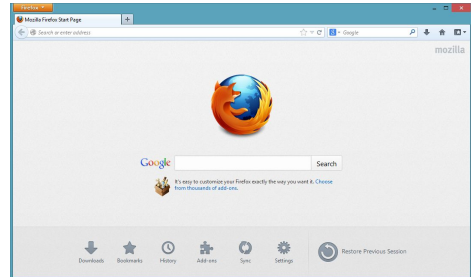

Kerberos authentication with AFS (Mac)

<https://www.auristor.com/filesystem/client-installer/>

You can run graphical applications over SSH

The X Window System is a client/server architecture, so you can actually run, for instance, Firefox on Andrew and interact with it pretty normally.

Just use: `ssh -X andrew`



AFS



What is AFS?

- Andrew (Carnegie, Mellon) File System
- AFS is a networked filesystem, similar to NFS but with a different consistency model.
- Kind of like an old-school Dropbox, but with limited sync capabilities.



Special directories

- private
 - The only directory that is not public by default.
- www
 - Content in here can be served over the internet (check out last week's extratation!)
- OldFiles
 - A copy of the filesystem from yesterday. Great if you accidentally delete something with like `rm -rf`



The AFS Protection Server

Unix implements fairly simple file permissions based on read/write/execute for a user/group/everyone.

AFS has a very complicated file permission scheme implemented using the Protection Server. You usually won't need to know about it, but if you lack permissions to a file, it's good to know that it could be either Unix or AFS blocking you.



Mounting a filesystem

- This is a Unix concept where one filesystem is “attached” as a directory to another filesystem
- Mounted filesystems are stored at /mnt
 - This is why on Bash on Ubuntu on Windows has your C: drive at /mnt/c
- AFS mounts its filesystems at /afs

Shell Tricks



Shell Tricks

- What is their purpose?
 - Speed up your workflow on terminal.
 - Good to know how to use.



Commands

- Useful commands:
 - “cd ..” to go to the parent directory.
 - “cd -” to go to whichever directory we were in before coming to this one.
 - “cd” to go to your home directory.
- Misc.
 - “history” will present you with a list of commands that you’ve already run.
 - “!!” will expand to the last thing you ran.
 - “!x” will expand to the last thing you ran that began with x
- Caret
 - “^C” to exit the program that you’re currently running
 - “^D” sends the EOF(end of file) signaling that there is no more input. This usually causes the shell to exit.
 - “q” to exit programs like less or man
 - “:q” to exit vim specifically.



Vim mode for Bash

Add the following to your `~/.inputrc` (you'll have to create the file) and restart Bash:

```
set editing-mode vi
```

```
set keymap vi-command
```