Cloud.cs CNM 190 User Guide

The class is being provided with about 800GB of RAID-1 data storage on cloud.cs. This data is backed up offsite approximately every week, but historical records are not kept (meaning if you accidently delete something, you need to mention it before the next backup). Jeremy Huddleston administers this machine, so please email him or bring up any concerns in class rather than contacting inst@eecs. This document is meant to be a quick introduction to getting access to this storage from your computer at home or in the computer labs.

Quick Notes

You can get access to cloud.cs using CIFS, SMB, AFP, or sftp. The login and password for all of these are the same. The user is 'cnm190', and the password was given out in class.

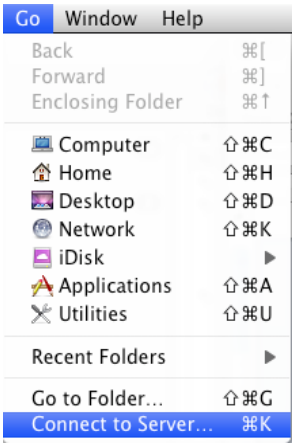
Mac OSX - AFP

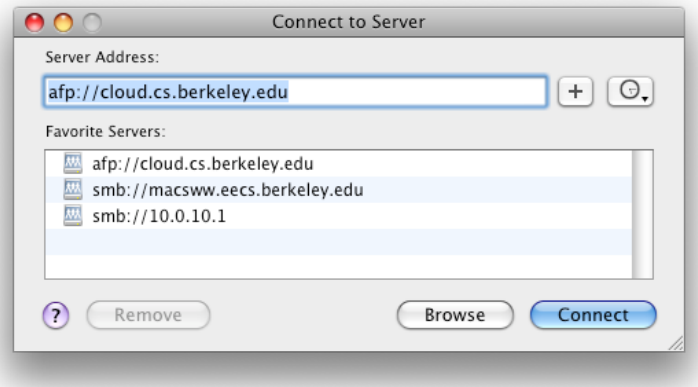
The details and images here correspond to Leopard (10.5), but the same thing should apply to Tiger (10.4) or Snow Leopard (10.6) with little change.

AFP is Apple's "native" network mounting protocol, and it is thus the best supported. To mount the disk, click on the Desktop so "Finder" is the active application. Then choose "Connet to Server" from the "Go" menu.

In the dialog box, enter 'afp://cloud.cs.berkeley.edu'. You can click the '+' icon to add the server to your list of favorites. Click on connect. At the prompt, enter 'cnm190' for the username. The password was given out on the first day of class.

The disk will now be mounted at /Volumes/cnm190 (and appear on your desktop)





If you are on a lab machine (as your cs194 account), you can do this directly from the Terminal (useful if you are ssh'd in for example) by running "mount_cloud.sh" from the command line. Make sure you unmount the server before you leave ("mount_cloud.sh –u" or drag the disk to the trash).

Windows - SFTP

The most secure way to get access to the data from windows is using sftp. My preferred program for doing this is putty. You can find it by googling for 'putty sftp'. Download the putty-0.60-installer.exe (http://the.earth.li/~sgtatham/putty/latest/x86/putty-0.60-installer.exe) and install it on your system.

Putty does not have a graphical interface for transferring files, so if you prefer that, you should try WinSCP (http://winscp.net).

Windows - SMB

As of this writing, SMB support is being firewalled by EECS, so you can only access the file server via SMB within Soda (we're trying to get this firewall opened up).

Open up an explorer window and in the address bar, type '\\cloud.cs.berkeley.edu'. You should then be prompted for a login and password. Enter 'cnm190' for the login and the password you were given. You should then see a list of shares. Open up 'cnm190'.

Additionally, you can right click on 'cnm190' and select 'map network drive' to map this share to a drive letter.

File Syncing

As the project grows, it will become increasingly difficult to sync files between local users and the file server. Plan ahead. In the past, teams have used unison and rsync. There is no one best solution, so find out who in your team has experience with what products and go from there. If you need help, please don't hesitate to ask!

File System Conventions

It is **very important** to come up with conventions for your file naming and directory hierarchy. Your project will fall apart if everything is in one directory. Create subdirectories for models (and subdirectories of that for each model), shots, sound, scripts, etc. It will be up to the Project Manager to lay these guidelines out explicitly in a file in your team's directory (named cat NAMING.txt) as part of an upcoming assignment, and we will discuss and refine these in crit.

Web Space

Each team has a directory designated for your use as a team website. The web server has many modern web technologies available, and if you need something else, please just ask, and we'll try to make it available for you. The team websites are located at http://cloud.cs.berkeley.edu/~cnm190 and the files being served for these sites are on cloud in the "web" subdirectory of your team's directory on cloud (eg: ~cnm190/fa2008/roses/web).

It is up to the team to decide how best to use this site. Please take a look at previous semesters for examples. In your individual status reports that will be submitted as bspace assignments, you will need to provide links that show off your individual work, so having space for each team member may be helpful.

Advanced Techniques (SSH and Mounting on "The Farm")

Mac OSX is a UNIX system and as such is highly scriptable. The machines in the lab have bash, perl, python, and ruby available for scripting. A very straightforward script is available to you to do a command on every system (in serial). You can use this command to mount cloud on every machine on the farm.

Because this command, farm_foreach.sh, uses ssh to connect to each machine, it is best that you setup an ssh key for authentication (on your lab account, not the cloud account) to avoid needing to enter your password every time. Your key can still be protected by a passphrase that you will use to unlock it and add it to ssh-agent. Your first session would look something like this:

1) Login to the remote system. You can do this from Terminal.app on a Mac or putty on Windows:

```
home $ ssh cs194-aa@ akane.cs.berkeley.edu
```

2) Setup an ssh key for authentication and copy over known_hosts for the farm:

```
~ $ ssh-keygen -C "Jeremy's CNM190 Key"
~ $  cp ~/.ssh/id_rsa.pub ~/.ssh/authorized_keys
~ $ cat ~cnm/resources/farm_hosts >> ~/.ssh/known_hosts
```

The last line simply copies over public keys for the farm machines to your known_hosts file to save you from typing "yes" many times the first time you run farm_foreach.sh. When you want to run commands on all the farm machines in serial (so by extension, this is not for rendering), you can do the following (assuming you're logged in already):

1) Cache your ssh credentials with ssh-agent which will allow you to connect again without providing the passphrase each time:

```
~ $ ssh-add
```

2) After that, you can connect to all the farm machines without entering your passphrase. You can then do things easily on all the farm machines:

```
~ $ farm_foreach.sh hostname \; w
~ $ farm_foreach.sh mount_cloud.sh
~ $ farm_foreach.sh mount_cloud.sh -u
```

In order to avoid entering the cnm190 password for every machine, you should take advantage of the OSX Keychain. mount_cloud.sh first checks the keychain for credentials and only asks for a password if it cannot find one (because it's not there or access was denied). To add the credentials to the keychain, you should first mount the server within Finder (as on the first page). To let mount_cloud.sh have access to the credentials, run mount_cloud.sh from Terminal.app (with the server not yet mounted) and choose "Always Allow" from the window. This "Always Allow" has some security implications in that any script could theoretically get access to cloud.cs if they could unlock your keychain (meaning you need to make sure your account password is strong if you do this!)