Lab Overview:

Lists UPL servers and the services on them that perform tasks essential to running the lab.

Eris – the main server in the UPL

* Home directories (NFS)
  + Home directories provided to all other UPL machines via NFS
  + /home
* Home directory backups as well
  + In /home\_backup
* Postgres Database of Users / their status as member, coord, etc
  + Python scripts use this to generate needed files for linux functionality
* AFS Connection
  + Mounts a network drive of the CSL lab so we an read out valid CS users.
* Webcam
  + Webcam control server
    - Runs out of /webcam folder on eris
    - Started by a line in /etc/rc.local on eris
    - These changes to rc.local should be moved so that cfengine enforces them, but I’m lazy
  + Webcam stream
    - Runs out of /webcam folder on eris
    - Started by a line in /etc/rc.local on eris
    - These changes to rc.local should be moved so that cfengine enforces them, but I’m lazy

Siren – Naming / IP responsibilities

* Bind9
  + DNS Server, basically allows us to use machine names instead of ip addresses
* Isc\_dhcp\_server
  + DHCP server, hands certain machines fixed ip addresses
    - We own a nice block of IPV4 IP addresses
  + Is really dumb and doesn’t always start right automatically
    - “hacky fix” in place for this, I ensure it is started with a line in /etc/rc.local on siren

Spearow

* Apache Webserver
  + Serves the UPL Website / Knowledgebase
  + Should also serve websites people have set up in their ~username/Public folders, but this has been broken for a little while. Might be a good first thing for someone to tackle and fix.

Chiptunes

* Mpd: music player daemon
  + The backend that actually plays music over the stereo
* Ncmpcpp: the nice frontend
  + A frontend to mpd, its how we tell mpd what to play

General Configuration

* Lives in ~upl/newconfig/files/cfmaster
  + Self explanatory, takes out actions on groups of machines (defined in cf.group)
  + Actions are defined in the other cf.\* files
    - Cf.main is the most important one
  + REALLY THOROUGHLY LOOK THROUGH AND UNDERSTAND THESE FILES
    - A lot of what they do is copy files to places
* Kerberos
  + System that allows people in the CSL to log in with their CS user / pass
  + We base our login off of that as well, look through the cf configs for references to a Kerberos config file, then read that.
* Cron Jobs
  + Cron is what ensures tasks run at certain times. All cron jobs are defined in cf.main
    - File backups (happen at 1am)
    - Package updating, upgrading (happens at 1am)
    - Cfagent (a helper program that checks for cfengine configuration changes)
      * Runs on every machine every 5 minutes
      * This is why adding a user can take 5 minutes to propagate and work

Helpful installed packages

Most (if not all machines) have these packages installed to help run the lab

* Nfs\_common – necessary for a machine to load homefolders
* Fail2ban – stops China and Russia from endlessly bruteforcing logins on our machines
  + Ip bans them for a few hours if they fuck around too much
* Libpam\_krb5 – package enabling Kerberos authentication so people can log in with their CS credentials
  + Contacts CS’s Kerberos server (Kerberos.cs.wisc.edu) to auth users
* Cfengine2 – cfengine, the package that really enforces machine “state” and configuration

Common Problems:

* KnowledgeBase
  + I can’t sign in to the Spearow knowledgebase, or page editing gives an error.
    - Disk is out of space on Spearow, probably need to clear /var/log
* Adding User
  + When adding a user, I get an error from adduser.py saying a file reports “Resource unavailable”
    - I LITERALLY have no idea why this happens, but you must take additional measures to ensure the user is added and can log in.
      * At the end of running, adduser.py runs three other python files, you must now run them manually, ( sudo ./script\_here.py) on eris
        + Export\_group.py - takes care of user groups
        + Export\_passwd.py - exports passwd file (for linux login)
        + Install\_passwd.py - puts passwd file in correct place for login
      * If any of the three scripts failed, I think it means that file is locked by something. If you’re more linuxy than I am, there are things that can be done to see what is locking a file.
      * Waiting a few minutes, then trying again, or trying to open the file in nano is usually enough to “unstick it”, then you can run it as a python script again
    - Now you have to manually create the user’s home directory, this is EZ
      * Sudo mkhomedir\_helper usernamehere