Multi-User Environments

Hands-On UNIX System Administration DeCal

Lab 3 — 6 February 2012
Due 13 February at 6:10 PM

1 Playing with users and groups

This short exercise should familiarize you with some basic tools used on multiuser systems. Please do the following exercises on an EECS Instructional server (like `nova.cs`), not an OCF login server, since the OCF's servers have security restrictions in effect and not all of these commands will work.

1. Try out the following commands. What do they do? (Include sample output in your answer.) Is there any overlap between these programs?

   (a) `finger Sysadmin_DeCal`

   (b) `w`

   (c) `who`

   (d) `getent passwd` *yourusername*

   (e) `last`

   (f) `id`

   Some of these commands accept usernames as input, and some of them behave differently without any arguments. Experiment!

2. `finger` lookups take a *long* time. Find an option (use the ~~Force~~ manual, Luke!) that lets you make `finger` lookups faster if you know exactly what username you're looking up.

3. Use one of the above programs to look up the UIDs of some system users (try root, daemon, bin, sys, adm, lp, sshd — note that not all of these exist on any given system) and some normal users (say, cs61b, cs198-8, bh, ddgarcia, whoever's sitting next to you). What pattern do you notice?

2 Nitty-gritty stuff

1. The UNIX manual has multiple *sections* — you can look up program documentation, general system documentation, and even access a C library

reference (try `man malloc`). Running `man passwd` will open the passwd(1) reference page, which contains documentation on the `passwd` binary, but you can also access documentation on the syntax of the `/etc/passwd` file.

Figure out how the manual system is organized on Solaris (used on INST's servers and the lab workstations) and how to call up the `/etc/passwd` manpage. What are the different sections used for, and what command do you have to run?

2. By default, `man` on Solaris uses `more -s` to display manpages, which works in a pinch but isn't very convenient. `less` is `more`, but with some additional features like the ability to scroll backwards. Figure out how to display manpages in `less` instead of `more`.

3. Write a simple pipeline using `getent passwd` *username*, your result from (1), and the `cut` utility, to turn *username* into the given user's real name (this is sometimes called the GECOS field).

4. *(Extra for Experts. We'll be covering shell scripting in a few weeks, so don't worry if you don't know how to do this problem.)*

   Write a simple shell script that, given *username* as an argument,

   - looks up every group that *username* is a member of and then
   - prints out every other user that has a group in common with *username*.

   You'll want to write a `for` loop using `groups`, `getent group`, and `cut`.