Computer Science 330
Operating Systems
Siena College
Fall 2022

Lab 1: Unix Setup

Due: 11:59 PM, Friday, September 16, 2022

In this lab, you will gain experience using your Unix accounts for this class. This is an individual lab, but you should feel free to ask for help and be willing to offer help to classmates.

Learning goals:

1. To ensure that everyone's Unix accounts are set up properly, and that you are prepared to be able to run C programs and access GitHub repositories from those accounts.

Getting Set Up

The link to create your GitHub repository for this lab is in Canvas.

Please put your name in the `README.md` of the repository before continuing.

C and GitHub on FreeBSD

Log into `noreaster.teresco.org` using `ssh` and complete these tasks. You might need to ask questions. You might need to create a GitHub Personal Access Token. This assumes you remember how to run C programs in Unix.

- Create a directory on noreaster for your work in this course.
- Clone your repository for this lab into that directory.
- If you haven't previously done so on noreaster, issue these commands (replacing with your own GitHub username and your email, of course) at the terminal prompt to associate your name and email with git commits:

  ```
  git config --global user.name "jcool"
  git config --global user.email "jcool@teresco.org"
  ```

- Run the `uname -a` command and redirect its output to a file `freebsd.txt`. Use `git` commands to add and commit this to your repository.
- Compile and run the `hello330.c` program in your repository. Run it and redirect its output to a file `hello330.txt` and add and commit this file to your repository.

- Using an editor of your choice (possibilities are emacs, vi, nano, a remote editor like the one in WinSCP, probably others), add a second printout to the program that mentions noreaster and/or FreeBSD. Compile and run again, redirecting the output to a file hellofreebsd.txt. Add and commit this file to your repository.
- Commit the modified version of hello330.c, and push this and all of your other commits to GitHub.

Linux VM Configuration

Next, you will do some configuration of the Linux virtual machine (VM) that has been created for you on the College's OpenNebula server.

Your VM has a public name (your last name followed by .teresco.org, so Joe Cool's would be cool.teresco.org) but this name maps to an IP address that is private to Siena's network. You will only be able to connect to your VM from on-campus computers (or other VMs, like VMWare Horizon via aruba.siena.edu).

Before you start, you will need your VM's name as described above, the username and password for the initial account that was created on your VM, and the root password for your VM.

Each of your VMs should be configured with the Debian 9 Linux distirbution, and has access to 1 GB of main memory and 2 GB of disk storage.

Please complete the steps below to configure your VM.

- Log into the initial user account on your VM with secure shell.
- Become the superuser with by issuing the su command and typing the root password.
- Install some important software packages using the apt-get package manager commands below:

  ```
  apt-get update
  apt-get install git
  apt-get install gcc
  apt-get install make
  ```

  and if you'd like to have the Emacs editor, also

  ```
  apt-get install emacs
  ```

  We will likely install additional software later in the semester.
- Set the host name of your VM with the hostname command. I recommend the same as its public name, *e.g.*, cool.teresco.org. Normally the hostname command just tells you your host name, but if you run it as root and give it a command-line parameter, it changes your host name to that value.

- Create your own user account with the login name of your choice. If you want your username to be "joe", you should issue the command

  ```
  adduser joe
  ```

  You will likely get several harmless warnings and/or error messages. If all goes well, you should be prompted for the password to set for the new account (you'll have to type it twice), then for some additional information about the account. Set an appropriate "Full Name" (your actual name would be good) but don't worry about the other fields. You can change your password later with the `passwd` command, and your other account information with the `chfn` command.

- In a separate window (*e.g.*, PuTTY, Git Bash, Mac Terminal, etc.) on your computer, log into your VM as the new account you just created.
- Run the `w` command to see that both you and `user` are logged in.
- Become superuser from your new account's login with the `su` command and giving the root password.
- Log out from the `user` account, then delete that account from the system with the `userdel` command.
- Change your root password to something you will remember with the `passwd` command. If you forget this password, we will not be able to recover it! You will need it later in the semester.
- Edit the `/etc/motd` "message of the day" file with an appropriate login message of your choice.

C and GitHub on your Linux VM

Log into your VM with the account you created for yourself using `ssh` and complete these tasks.

- Clone your repository for this lab into your VM account.
- Issue these commands (replacing with your own GitHub username and your email, of course) at the terminal prompt to associate your name and email with git commits:

  ```
  git config --global user.name "jcool"
  git config --global user.email "jcool@teresco.org"
  ```

- Run the `uname -a` command and redirect its output to a file `linuxvm.txt`. Use `git` commands to add and commit this to your repository.

- Compile and run the `hello330.c` program in your repository. Run it and redirect its output to a file `hello330before.txt` and add and commit this file to your repository.
- Edit the program to add a third printout to the program that mentions VMs and/or Linux. Compile and run again, redirecting the output to a file `hellovm.txt`. Add and commit this file to your repository.
- Commit the modified version of `hello330.c`, and push this and your other commits to GitHub.
- Copy the message of the day file you edited in the previous section into your repository, and add and commit the file (so you'll have a file `motd` in your repository.)
- Log back into your noreaster account and pull the changes you made here to your clone of the repository on that system.

Submission

If you did all of the commiting and pushing, you're already done. Make an Issue in your repository on GitHub tagging me (`@jteresco`) in the text of the Issue to let me know you're done.

Grading

This assignment will be graded out of 40 points.

Feature	Value	Score
C and GitHub on FreeBSD	15	
Linux VM Configuration	10	
C and GitHub on Linux VM	15	
Total	40	