

System Administration Decal
Intermediate Lab #5
March 20, 2008

Introduction

This lab is oriented around getting acquainted with your new virtual server. There aren't very many questions, but do submit responses anyway. In any case, I will be checking that you've done everything by logging into your server. This lab should be done with your entire group, so I expect only three submissions total. Remember to note down the members of your group, the user IDs of each member, and the address of your virtual server.

There are many troublesome parts of this lab that are meant to challenge you as a group, so keep that in mind while working on it. III is notably hard, so you might want to take a look at that first. The first two parts could conceivably be done by one person. If you really need help, you can email me or come to my office hour. Again, my email is joshk.decal@triplehelix.org.

I. Setting Up Users

Since all the members on your group will probably be working at different times, it is going to be important for everyone to be able to log in and work on their own time. They'll also need administrator access. You'll also be learning a few tricks for how to set up user accounts without actually needing to give them a preassigned password or having them type it in on the keyboard.

For this exercise, just paste relevant terminal transcripts of what you did to accomplish what the question asks.

1. Use the adduser program to create users for each member in your group.
2. Install sudo. You can do this from source or using Debian's package system, your choice. If you choose to build it from source, use Debian's package system to install 'build-essential', which provides GCC and the requisite include files to build basic C programs.
3. Add all your users to the sudoers file. You can either add them by creating a line for each user in the sudoers file, or by adding them all to a common group (the traditional "admin" group is 'wheel'), and adding a sudoers line for the group.
4. Now you're going to let me log in to your server, so add a 'joshk' user. To allow me to log in without having to ask me to type in a password, we'll be using SSH public key authentication. See <http://sial.org/howto/openssh/publickey-auth/> for a brief intro. My public key is available at http://triplehelix.org/~joshk/id_rsa.pub. **I will check to see that I can log in to your server once your group emails me indicating that you have completed the lab**, so allow yourself time to recover from mistakes. If in doubt, just generate a private key for yourself and try out SSH public key authentication on your own before doing it with my private key. If even more in doubt, come to my OH Tuesday after spring break.

II. Shootin' The Breeze, Unix Style

You might not think it, but early UNIX geeks who had nothing better to do came up with lots of ways for users on a multiuser system to talk to and interact with each other. If you're a CSUA member, soda is a great (albeit quite restricted) example of this.

Use the 'apropos' command to search through man page synopses to find the commands you need. You can just use keywords like 'user' and 'logged'.

1. Find at least three ways to figure out whether a particular user is currently logged on to the machine. (**Bonus:** Each solution you will find looks in a particular file for the information it displays to you. What is that file?)
2. What does the 'finger' command do?
3. Have all your group members log on to your virtual server. Use the 'write' command to talk to other group members.
4. Find a way to address all the users on a computer.
5. Finally, find a way to prevent other people from using 'write' to spam your terminal.

III. NetHack – Compilation Challenge

NetHack is a game that has existed in some form or another for at least 20 years. It started out as a game called rogue, then evolved to a game called hack, and then NetHack is the latest version that was developed by many people around the world. It's a game – with terminal-only graphics – where you are in a dungeon looking for the famed Amulet of Yendor. During your quest, you move around, pick up treasures, equip weapons, open doors, buy supplies, and learn spells, and of course you defeat a ton of enemies.

In this exercise, you'll compile NetHack from source on your virtual server, and then play it. Be sure to mention all the steps you took to compile the software. Warning: **This is probably the most challenging part of this lab.**

1. You will need to install an additional library, ncurses, to get NetHack to build, so compile that first. Build it from source! Remember to have the build-essential package installed first.
2. Next, you will need to build and install GNU bison and GNU flex. This provides the 'yacc' and 'lex' programs. If it has a dependency, build and install that too.
3. Download NetHack's source from www.nethack.org. Build it on your server. It *doesn't* quite fit the three-step procedure I mentioned in class, so you're on your own! Remember to check the README. You'll have to navigate through a few files to figure it out. This will take you a while, but that's why this is a *group* lab, and everyone can work on it.

You will need to make a change to a Makefile to prevent NetHack from linking against a library you don't have. Finally, you might have to delete a repeat declaration of a function from a source file, which causes a build error.

You will need to make a change to a *different* Makefile to make NetHack use flex instead of trying to use 'lex'. Alternatively, you could create a symlink from lex to flex.

Finally, remember that once you have built the binaries, you have to install them before anything works.

4. Play some NetHack and let me know what your high score is. I suggest starting with the Valkyrie or Samurai character class. Use the Guidebook at <http://www.nethack.org/v343/Guidebook.html> to help you get started with playing. Check out <http://alt.org/nethack/plr.php?player=joshk> for my high scores. Happy hacking!