Homework 2

UNIX Commands

Outcomes

- Increase your vocabulary of UNIX commands.

Rubric

- Half a point each for the 20 commands mentioned.

Description

Unlike a GUI environment where features are discoverable (there's only so many things you can click on!), the list of UNIX commands is huge and diverse, and just like with a programming language like C or C++, you need to read up and understand what each new command or keyword does. If you have the analogy of a programming language in mind, individual commands are now function calls, and the arguments to the commands are the arguments to the functions. So, the goal is to build up a functional 'vocabulary' that you can use to perform a wide range of operations. Composing that 'vocabulary' into sentences is done with pipes, and the results can be output to the terminal (the default location for both standard output and standard error), stored in variables, or sent to files via redirection.

Towards that goal, we are providing a set of commands that are very widely used in system administration. Commands that have lots of useful options like 'grep' and 'find' often have good tutorials online, and man pages and internet research help. If a man page is particularly opaque, an online search for 'linux <commandname> command tutorial' (unquoted) or something similar will often be useful because other people will be having the same problem. You'll also want to run and test commands to verify that how you think they work is how they actually work.

Specifications

- Given a list of UNIX commands, put a short description of what they do in general as a comment in your script. In this specific case, use the versions of the commands available on CentOS 7.
- Provide one (or more, if you want) examples of using that command to do something you feel is useful in a system administration context in your script. Provide a comment above each one that describes what the command is doing. When you are doing this, think about how the command could be used in system administration: to either change system state in a useful way, provide a service, or find out something useful about a system that you could use when debugging a problem. The options given to the

command change the details of the functionality it provides, that's the area to focus on for how figure out how to do useful things with it. Test the commands in your VM.
- Put all these commands in one script, with a comment above the command about which number the command is a solution to.
- If it makes sense to do so, it is fine to put the command in a pipe. But keep your example limited to one line (no semicolons) at most.

Commands

Here is the list of commands, some of which have been mentioned in class and some which are new:

```
dd
find
file
fuser
grep
host
ldd
lsof
mount
ps
pkill
netstat
renice
rsync
time
ssh
stat
strace
uname
wget
```

You may need to install some of them using yum. If you need to do this, you can find out which package to install by running yum, the following example figures out what provides any file in any `*bin` directory named strace:

```
yum whatprovides *bin/strace
```

Submission

A single bash script with the commands and your descriptions in them. You should have created a script above but you will need to rename it from .sh to .txt so it can be uploaded to iLearn.
- hw2.txt