<Brian W. Lee>

CSC 2510: DevOps

Lab 07 – Unix Utilities

General Instructions

Using your book and previous lecture material, fill out this assignment sheet. **Use red text to signify your answers.** This assignment corresponds with various sections of your textbook. You should utilize online resources to answer these questions as well.

Submission Instructions

To submit, **change the name in the header** and save this document as a PDF. Attach your PDF document to the iLearn dropbox.

Lab Questions

Write the commands used or the answers to the following questions.

1. The `find` (Sobell, page 828) utility locates files based on simple or complex criteria you specify. Use `find` with the `-name` (Sobell, page 830, and see the examples on Sobell, page 832) criterion to list the names of files in `/bin` that begin with the letter `c`.

command: find /bin -name 'c*'

output:

```
vanquisherx615@demo:~$ find /bin -name 'c*'
/bin/chgrp
/bin/chacl
/bin/chvt
/bin/cpio
/bin/chown
/bin/chmod
/bin/cat
/bin/cp
```

2. Use `find` with the `-name` (Sobell, page 830) and `-type` (Sobell, page 830) criteria to list all files in the `/bin` directory that have the characters `sh` in their names and are symbolic links.

command: find /bin -name '*sh*' -type l

output:

```
vanquisherx615@demo:~$ find /bin -name '*sh*' -type l
/bin/sh
/bin/sh.distrib
/bin/rbash
/bin/static-sh
```

<Brian W. Lee>

3. Use `find` to list all the files in the `/usr` hierarchy. Send the output through a pipeline to `head` to list only the first 10 files in the list. It may take a while before `find` displays output.
command: find /usr | head

output:
```
vanquisherx615@demo:~$ find /usr | head
/usr
/usr/include
/usr/include/gawkapi.h
/usr/include/reglib
/usr/include/reglib/regdb.h
/usr/include/reglib/reglib.h
/usr/include/reglib/nl80211.h
/usr/include/sudo_plugin.h
/usr/include/btrfs
/usr/include/btrfs/extent_io.h
```

4. Use `grep` to display the line(s) that contain(s) the string `model name` in the `/proc/cpuinfo` file. How can you ensure that `grep` interprets a string that contains a SPACE as a single argument? the two single quotes around model name ensure everything inside the quotes are interpreted including the space.
command: grep 'model name' /proc/cpuinfo

output:
```
vanquisherx615@demo:~$ grep 'model name' /proc/cpuinfo
model name      : Intel(R) Xeon(R) CPU @ 2.20GHz
model name      : Intel(R) Xeon(R) CPU @ 2.20GHz
```

5. How many lines in `/etc/services` do not contain the string `send`? 590
```
vanquisherx615@demo:~$ grep -v -c 'send' /etc/services
590
```

6. List each line in `/etc/services` that contains the word `send`. Precede each line by its line number in the file.
command: grep -n 'send' /etc/services

output:
```
vanquisherx615@demo:~$ grep -n 'send' /etc/services
23:msp          18/tcp                          # message send protocol
```

The next set of questions assume the following file named `food` is in the working directory. The file lists foods, cost, and calories. **The blanks in the file must be SPACEs and not TABs, and the `LC_ALL` locale variable must be set to `C` for the steps to work.**

```
$ export LC_ALL=C
$ cat food
tuna salad  .25  100
rare hamburger  2.50  500
french fries  0.75  625
tea bag  .20  0
apple pie  2.75  1500
milk shake  3  200
potato chips  0.50  80
```

<Brian W. Lee>

7. Use the `sort` (Sobell, page 975) utility to display the lines of the `food` file in alphabetical order.

command: sort food.txt

output:

```
vanquisherx615@demo:~$ sort food.txt
apple pie  2.75  1500
french fries  0.75  625
milk shake  3  200
potato chips  0.50  80
rare hamburger  2.50  500
tea bag  .20  0
tuna salad  .25  100
```

8. Use `sort` to display the `food` file in reverse alphabetical order.

command: sort -r food.txt

output:

```
vanquisherx615@demo:~$ sort -r food.txt
tuna salad  .25  100
tea bag  .20  0
rare hamburger  2.50  500
potato chips  0.50  80
milk shake  3  200
french fries  0.75  625
apple pie  2.75  1500
```

9. Use `sort` with the `--key=` option to display the lines of the `food` file sorted by the second word in the name of the food.

command: sort --key==2 food.txt

output:

```
vanquisherx615@demo:~$ sort --key=2 food.txt
tea bag  .20  0
potato chips  0.50  80
french fries  0.75  625
rare hamburger  2.50  500
apple pie  2.75  1500
tuna salad  .25  100
milk shake  3  200
```

10. Use `sort` with the `--numeric-sort` (`-n`) option to create a file named `calo` that holds the lines of the `food` file ordered by calories with the food with the most calories at the top of the list.

command: sort --key=4 -n -r food.txt >> calo

output:

```
vanquisherx615@demo:~$ sort --key=4 -n -r food.txt >> calo
vanquisherx615@demo:~$ cat calo
apple pie  2.75  1500
french fries  0.75  625
rare hamburger  2.50  500
milk shake  3  200
tuna salad  .25  100
potato chips  0.50  80
tea bag  .20  0
```