**CSE360: Script Programming Final Exam**

**Name: Yousef Jarrar Student ID: 005845836**

**SECTION I**

[1] How to find out your server’s uptime on Linux System?

* To find out your uptime, you would type: #uptime -p
* The “-p” will print based on days, hours, and minutes

[2] What does the following command do?

[ root@lin ~]# **w**

* This command also allows you to view how long the server has been running with more information.

[3] Compare following commands:

[ root@lin ~]# **users 🡺 This outputs who is currently logged in according to FILE**

[ root@lin ~]# **who 🡪 This command allows you to view who is currently logged in.**

[ root@lin ~]# **w 🡪 This command allows you to view complete information about user, tty, and server information.**

* All of these commands above share the same information, they all perform the same function. It just depends on exactly what you are looking for.

[4] What does the following command do?

[ root@lin ~]# **ls -latr**

🡪 This is a list files command on unix, and the -latr does the following:

* Ls -l 🡪 Long Format, that displayes unix file types, permissions, and number of hard links, owner, group size, and last modified date.
* Ls -a 🡪 list list all files in the given directory, including those whose names start with “.” (hidden files)
* Ls -t 🡪 This will allows for sorting by date and time
* Ls -r 🡪 This option allows for listing in the reverse order.

[5] What does the following command do?

[ root@lin ~]# **crontab -e**

* **This allows you to either edit or create a new crontab if it does not exist. A crontab is setup to run a command at a certain time, on a scheduled time.**

[6] What does the following crontab will do?

[ root@lin ~]# **crontab -l**

**15 0 \* \* \* root find /tmp -type f -empty -delete**

* The crontab -l command will display your crontab file that already created. An exmaple is shown above this answer on how it will be formated.

[7] What are the differences between **more** and **less** command?

* The “more” command is a filter for paging through text one screen at a time, It was used in the early stages of linux. The “less” command is a program that is like “more” but it allows for backward movement in the file as well as forward movement; The “less” command also does not read entire input files, that is why it starts up faster than a vi editor.

[8] What are the differences between following commands?

[ root@lin ~]# **cp -p file1.txt file2.txt**

[ root@lin ~]# **cp -i file1.txt file2.txt**

* **These 2 commands are copy commands but there is a a difference in the way they work. The “cp -p” command allows the person to “preserve” the following characteristics of each souce path in the corresponding target. This includes modify date and ownership etc..**
* **The “cp -i” (interactive) command allows the person to be prompted with the name of a file to be overwritten. This can be taken from the sorcefile.**

[9] What does the following command do?

[ root@lin ~]# **cat fileA.txt fileB.txt > fileC.txt**

* **The cat command above is called (concatenate) which is used to create and or read whatever is in the files. The above command will allow you to take the information from “fileA.txt” and “fileB.txt” and copy it into a new file “fileC.txt”. Simply put all output will be redirected in a newly created file.**

[10] What does the following command do?

[ root@lin ~]# **mv fileA.txt fileB.txt**

* **This command allows you to move all the content from fileA.txt into fileB.txt. The system will confirm with you, asking if you would like to overwrite the file. It takes all the information from A and overwrites into B.**

[11] What is the final directory after the last command?

[ root@lin ~]# **cd**

[ root@lin ~]# **cd /etc/sysconfig/network-scripts/**

[ root@lin ~]# **cd**

[ root@lin ~]# **cd –**

* **The final directory is /etc/sysconfig/network-scripts.**
  + **[root@lin network-scripts]#**

[12] What does the following command do?

[ root@lin ~]# **pwd**

* **This command will allow you to print the working directory.**

[13] What are the results of the last two sort commands?

[ root@lin ~]# **cat file**

orange

apple

pear

banana

[ root@lin ~]# **sort file**

[ root@lin ~]# **sort -r file**

* **The last 2 commands are for sorting out the information. The “sort file” command will take the information and sort it out Alphabetically.**
* **The “sort -r file” command will allow the information to be sorted in reverse alphabetical order.**

[14] What does the following command do?

[ root@lin ~]# **vi -R /etc/dhcpd.conf**

* This command will start the Vi of the following document dhcpd.conf file. The only thing is that it will open the Vi in “read-only mode” sort of like a view option.

[15] What does the following command do?

[ root@lin ~]# **ssh dmckay@hadrian.csusbcoyote.net**

* This will allow you to ssh into the username of “dmckay” and into the server named “hardian.csusbcoyote.net”. This of course will prompt you with a password.

[16] What does the following command do?

[ root@lin ~]# **service dhcpd start**

* This is the old command that is used on Unix, but this will start the DHCP Server.

[17] What does the following command do?

[ root@lin ~]# **free**

* This command provides information about the used and unused portions of the memory, swap space, on any computer running linux.

[18] What does the following command do?

[ root@lin ~]# **top**

* The top command displays processor activity of your linux box and also displays tasks managed by the kernel in real-time.

[19] What does the following command do?

[ root@lin ~]# **tar -zcvf backup.tar.gz /home**

* **Tar is a type of archive that CentOs uses. It will then create a backup.tar.gz under the home directory**
* **tar -c 🡪 Creates the archive**
* **tar -f 🡪 Uses archive file or device archive.**
* **tar -z 🡪 Filter the archive through gzip**
* **tar -v 🡪 Verbose; verbosely list files processed.**

[20] What does the following command do?

[ root@lin ~]# **tar -zxvf backup.tar.gz**

* **The following command is like the last one but the difference between the two is that this will decompress the file based on the command given tar -x.**

[21] What does the following command do?

[ root@lin ~]# **grep dmckay /etc/passwd**

* **The grep command will search from within the passwd file that has the keyword “dmckay”. The command allows for the person to filter through all the text by using keywords.**

[22] What does the following command do?

[ root@lin ~]# **find / -name core**

* **This is the find command, which will search for a file who’s name is core starting from the root directory.**

[23] What does the following command do?

[ root@lin ~]# **lsof -u dmckay**

* **This will list all the “List of Open Files” that will filter to exclude select login|UID sets.**

[24] What does the following command do?

[ root@lin ~]# **last -10 dmckay**

* **This will provide information about the username “dmckay” about the last 10 times that they logged into the server. It reads from a log file that is stored within /var/log/wtmp**

[25] What does the following command do?

[ root@lin ~]# **ps -aux |grep mysqld**

root 8625 6895 0 11:59 pts/1 00:00:00 mysqld

* This will take all the process that have been filtered based on ttyl sessions with the effective username and will dump them into the mysqld system.

[26] What does the following command do?

[ root@lin ~]# **kill -9 8625**

* This allows you to kill the process 8265 and with -9 command that is specified; this means that it will kill the process that is not catchable or ignorable. Sort of like a end task now command.

[27] What does the following command do?

[ root@lin ~]# **rm -rvf ~/CS300/HW**

* This will remove the file with the specified path.

[28] What does the following command do?

[ root@lin ~]# **mkdir -p CS300/HW/HW1**

* **This will create a directory with the following path.**

[29] What does the following command do?

[ root@lin ~]# **umask**

* **This controls the permissions for newly created files. It refers to a function that sets the mask, or it may refer to the mask itself.**

[30] What are the differences between following commands?

[ root@lin ~]# **su**

[ root@lin ~]# **su -**

* **The difference between the two, is that with the “su –“it allows you to login using a shell where they can assume the user using the current environment. What this means is that using “su – “it will allow you to “remote” into another user and take over their entire environment. Whereas the “su” just emulates a user for a moment, it is usually used when trying to install or in need of administrator privileges.**

**SECTION II**

[1] What is the result of following script?

#!/bin/bash

#

if [ -e ./sysctl.conf ]; then

IPF=$(grep "^net.ipv4.ip\_forward =" ./sysctl.conf);

if [ "$IPF" = "net.ipv4.ip\_forward = 1" ]; then

echo "IP Forwarding is enabled previously.";

exit;

fi

sed -i 's/net.ipv4.ip\_forward = 0/net.ipv4.ip\_forward = 1/' ./sysctl.conf;

echo "IP Forwarding is enabled just now.";

fi

exit;

* **The following script sets up the system for Forwarding. It takes the ipv4 from the sysctl.conf file and enables the port forwarding. It switches from a 0 to a 1.**

[2] What does the following script will do?

#!/bin/bash

#

echo -n "Enter your name: ";

DEPT=”IT Department”;

read NAME;

echo "Hello $NAME” | festival --tts;

echo "Welcome to $DEPT” | festival --tts;

exit

* **When the script runs, it will input the name from the user and it will take the variable that was setup “IT Department” and it will out the following Hello Yousef, Welcome to IT Department.**

[3] What does the following script will do?

#!/bin/bash

#

cat << DHCP\_HEADER > ./dhcpd.conf

#

# DHCP Server Configuration file.

# see /usr/share/doc/dhcp\*/dhcpd.conf.sample

#

ddns-update-style interim;

ignore client-updates;

subnet 139.182.148.0 netmask 255.255.254.0 {

option routers 139.182.148.1;

option subnet-mask 255.255.254.0;

option domain-name "cse.csusb.edu";

option domain-name-servers 139.182.2.1, 139.182.154.13;

option time-offset -28800;

default-lease-time 21600;

max-lease-time 43200;

}

DHCP\_HEADER

* **This sets up the DHCP server with the following configuration. It sets up a subnet and the subnet mask. It inputs the following information of the router, domain name, and domain name servers, the time that it will allow for an IP address to be leased. This script sets up the DHCP server to allow other machines to pull an IP address from it, and how it will communicate with the outside world.**

[4] Put comments in the following script: Replace “**Comments here”** with explanation

#!/bin/bash

# Date:

# Author:

# Description:

#

# **This script will allow for IP Forwarding**

if [ -e ./sysctl.conf ]; then

# **The command below will take the current ./sysctl.conf file and create a new file called “sysctl.conf.bak” which is considered a backup.**

cp ./sysctl.conf ./sysctl.conf.bak;

# **This will enable the IP forwarding without rebooting the system.**

IPF=$(grep "^net.ipv4.ip\_forward =" ./sysctl.conf);

# **This is an if statement: If System Forwarding has already been enabled then the output will be “IP Forwarding is enabled previously” and will then end the if statement and exit.**

if [ "$IPF" = "net.ipv4.ip\_forward = 1" ]; then

echo "IP Forwarding is enabled previously.";

exit;

fi

# **This if statement executes only if the top one failed, which executes the IP Forwarding from disabled to enabled and returns a message “IP Forwarding is enabled just now” and then exits the if statement.**

sed -i 's/net.ipv4.ip\_forward = 0/net.ipv4.ip\_forward = 1/' ./sysctl.conf;

echo "IP Forwarding is enabled just now.";

exit;

fi

# **This echo command tells the user that it could not find the “sysctl.conf” file and this only executes when the 2 if statements above it did not resolve.**

echo "sysctl.conf file not found!";

exit;

[5] Write a bash script that generate the dhcpd.conf configuration file as following:

*The script will take the range of IP addresses and ping each IP addresses.*

*Get the MAC addresses from arp table and create DHCP registration based on MAC address.*

*So that DHCP server gives out network configuration information to clients based on MAC addresses.*

#

# DHCP Server Configuration file.

# see /usr/share/doc/dhcp\*/dhcpd.conf.sample

#

ddns-update-style interim;

ignore client-updates;

subnet 139.182.148.0 netmask 255.255.254.0 {

option routers 139.182.148.1;

option subnet-mask 255.255.254.0;

option domain-name "cse.csusb.edu";

option domain-name-servers 139.182.2.1, 139.182.154.13;

option time-offset -28800;

default-lease-time 21600;

max-lease-time 43200;

}

host coyote-120 {

hardware ethernet 2c:27:d7:3b:4d:65;

fixed-address 139.182.148.120;

}

host coyote-121 {

hardware ethernet 2c:27:d7:3b:49:df;

fixed-address 139.182.148.121;

}

……………

host coyote-130 {

hardware ethernet 2c:27:d7:3b:4d:58;

fixed-address 139.182.148.130;

}