**Matthew Austin**

**Assignment 4**

**CIS356** – Spring Term 2018

**Point Value**: 100 points

**Submission Instruction**

Please submit your assignment 4 to your instructor in class on the due date and submit your hw4\_lastname\_firstname.pdf or .docx on Schoology.

**Description**

1. Is it really that bad to turn off a UNIX or Linux system with the power button on the computer case? What about unplugging the computer from the wall? Explain your answer. See if you can determine the likelihood of a bad outcome by doing Internet research. (**10 points**)
   1. Yes, you may corrupt your hard drive. Computers need to be turned off properly so that files can be closed and processed can be completed. Shutting the computer off via the power button the case surprises the computer disrupting process as well as RAM which only holds memory as long as power is supplied. All of those things will contribute to you probably corrupting and losing all your data.
2. Please list the six major phases of system bootstrapping process (**6 points**)? You need to describe what each phase does? (**14 points**)
   1. bootloader → Initialization of Kernel (OS) → Creation of kernel (init, kjournaled, kswapd)→ Detect and configuration of hardware → single-user mode → startup script
   2. **1.)** reading the bootloader from the Master Boot Record, **2.)** Initializing the kernel for the next step **3.)** Creation of the kernel, along with init, kjournled and kswaped, **4.)** Detect hardware that is connected to the computer along with their configuration, **5.)** Loading the set default user mode **6.)** Then loads up startup scripts which is time-saving and can do multiple tasks
3. Please explain the concept of run levels. List the run levels defined on your Amazon Ubuntu server and briefly describe each? (**10 points**)
   1. A runlevel is one of the modes that a Unix -based operating system will run in. Each runlevel has a certain number of services stopped or started, giving the user control over the behavior of the machine. After the Linux kernel has booted, the init program reads the /etc/inittab file to determine the behavior for each runlevel. Unless the user sets another value as the kernel boot parameter it will always run default user run-level 1

|  |  |  |
| --- | --- | --- |
| Run Level | Mode | Action |
| 0 | Halt | Shuts down System |
| 1 | Single-User Mode | Does not configure network interfaces, start daemons, or allow non-root logins |
| 2 | Multi-User Mode | Does not configure network interfaces or start daemons. |
| 3 | Multi-User Mode with Networking | Starts the system normally. |
| 4 | Undefined | Not used/User-definable |
| 5 | X11 | As runlevel 3 + display manager(X) |
| 6 | Reboot | Reboots the system |

1. Assume that there is a program/executable file called “Create\_Directories.sh” at (/home/Ubuntu). The content of the script is the following:

#!/bin/bash

mkdir /home/ubuntu/directory\_one

mkdir /home/ubuntu/directory\_two

mkdir /home/ubuntu/directory\_three

mkdir /home/ubuntu/directory\_four

mkdir /home/ubuntu/directory\_five

Please show how you would use systemd, unit file, and target, to glue it into the system to run the script automatically at boot time. Assume that the system’s default target is *multi-user.target*. (**20 points**)

1. Ensure that the .sh file is executable and is in /home/Ubuntu/Create\_Directories.sh
2. copy an existing service file and paste a file named createDirectories.service in lib/systemd/system
3. Modify the file as following:

[Unit]

Description=

After=multi-user.target

[Service]

WorkingDirectory=/home/Ubuntu

ExecStart=/bin/bash /home/Ubuntu/Create\_Directories.sh

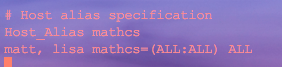
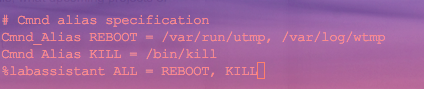
Restart=always

[Install]

WantedBy=multi-user.target

4.) $sudo systemctl enable createDirectories

5.) $sudo systemctl isolate reboot.target (restart system)

1. Please create two entries in the /etc/sudoers file:
   1. One entry that allows users *matt* and *lisa* to do everything like the root users on the machine named *mathcs*. (**10 points**)
   2. One entry that allows the group *labassistant* to kill jobs/processes and reboot the local machine. (**10 points**)h

1. Please explain what is the traditional UNIX Access Control (**5 points**) and what is the Role-Based Access Control (**5 points**)
   1. At the early stage of Unix, some general rules that shaped the Access Control system: Objects (e.g., files and processes) have owners. Owners have broad control over their objects, You own new objects that you create, The special user account called “root” can act as the owner of any object, Only root can perform certain sensitive administrative operations.
   2. Role-based access control is a method of regulating access to computer or network resources based on the roles of individual users within an enterprise. In this context, access is the ability of an individual user to perform a specific task, such as view, create, or modify a file.
2. Why is it better to use command *sudo* rather thanlog in the *root account* for system administrative tasks. (**10 points**)
   1. If several people have access to the root account, you won’t be able to tell who used it and when.
   2. It doesn’t record the command executed as root, but it creates a log entry that states who became root and when.
   3. No accountability.