Due Date: 22/03/2022

Lab 1

Linux Account Management and Access Control List [10 Marks total]

Matt Romanes - 30049211

Instructions

1. Follow the instructions in this document,
2. Answer the questions in the order they appear in this document and in the labs included in this document (See each lab's document for more instructions)
3. Submit the file (.doc, .docx, .pdf) using the ecs submission system (i.e. lab1). https://apps.ecs.vuw.ac.nz/submit/CYBR371

Part 1 - Linux Account Management

Complete the lab "**Linux Account Management**" and answer the following questions:

❖ *Question 1.1 [1 Mark] - What is the numerical representation of the following permissions?*

rwxrwsr-t= 785
r-sr-x--x = 851
rwxr-Xr-x = 755
r-sr-Sr-x = 865

❖ *Question 1.2 [1 Mark] - If the umask value for a user is 035, what are the default file and directory permissions set for the user? Write the permissions and how they were calculated.*

Links that helped complete this question:

- https://www.youtube.com/watch?v=KREFnMyYlfw
- https://docs.oracle.com/cd/E19683-01/817-3814/userconcept-95347/index.html
- https://www.linuxtrainingacademy.com/all-umasks/

To calculate the umask value for files and directories, I subtracted the default directory (777) and file (666) permissions by the given umask value (035).

Directory: 777 - 035 = 742 –> rwx r-- -w-
Files: 666 - 035 = 631–> rw- — -w-

Based on the calculations above, for a **Directory** the *user* will have read, write and execute permissions, while for a **File** they will only have read and write permissions (as files are not executable by default).

For *group*, they will have no permissions for a file and read only permissions for a directory.

For *other*, they will have write permission only for both a file and a directory.

❖ *Question 1.3 [1 Mark] - If the default permissions given to files the user xyz creates are r-xr--r--, what are the default permissions set for the directories created by the user? Write the permissions and how they were calculated.*

Links that helped me complete this question:

- https://linuxize.com/post/umask-command-in-linux/

From file permission:

r-xr–r–

Default directory permission: 777

777 - 544 = 233 -> 233 is the umask value

Umask 233 gives the directory permission as follows:

	Read	**Write**	**Execute**
User	-	w	-
Group	-	w	x
Other	-	w	x

Directory permission: -w- -wx -wx

Therefore the user xyz only gets write permissions to a directory.

❖ *Question 1.4 [1.5 Marks] - Provide an example of how SUID bit is used on a Linux system and explain the rationale behind its usage.*

Links that helped me complete this question:

- https://www.youtube.com/watch?v=DF1-XRUo6OE
- https://www.liquidweb.com/kb/how-do-i-set-up-setuid-setgid-and-sticky-bits-on-linux/
- https://www.reddit.com/r/linuxquestions/comments/2o3298/why_do_we_use_suid_instead_of_just_changing_the/

SUID (Set User ID) is a permission in Linux that allows the current user to execute a file regardless of who actually owns it.

The SUID is a temporary privilege, meaning that the elevation in privilege only applies when a file with an SUID bit is executed.

SUID is useful for files like usr/bin/passwd, where users will have access to that file **only** when they want to change their password. In this scenario, they can run it as the owner of the file rather than just a user executing it, if only temporarily.

Part 2 - Linux Access Control List (ACL)

Complete the lab "**Linux Access Control List**" and answer the questions highlighted in this document in the order they appear in the lab document. Please note that the questions below are dependent on the sequence of the lab instructions and must be followed and answered step by step as they appear in the **Linux Access Control List** lab document.

❖ *Question 2.1 [1 Mark]: Write the command(s) you used to add the users with their associated provided information*

To create the users specified in the lab instructions, I used the following commands for each account:

- CYBR371: useradd cybr371 -g sudo -m -s /bin/bash
- Ben: useradd ben -g sudo -m -s /bin/bash
- David: useradd david -U -m -s /bin/bash
- Mary: useradd mary -U -m -s /bin/bash
- Masood: useradd masood -U -m -s /bin/bash

❖ *Question 2.2 [1 Mark]: Login as user "**ben**" and write a command to append the line "`This line is from the user ben`" to `myfile.txt` file in `cybr371`'s home directory (use absolute path). Write the command you used to append the line and explain the output (i.e. did you manage to append the line? Explain why the command was successful and/or why it failed).*

As per the lab instructions I logged in as the user ben and attempted to echo a message to myfile.txt in cybr371's home directory. This was the result:

```
cybr371@Ubuntu:~$ umask
0022
cybr371@Ubuntu:~$ umask 002
cybr371@Ubuntu:~$ umask
0002
cybr371@Ubuntu:~$ touch myfile.txt
cybr371@Ubuntu:~$ echo This file was created by user cybr371 >> myfile.txt
cybr371@Ubuntu:~$ ls -l
total 16
-rw-r--r-- 1 cybr371 sudo 8445 Apr 16  2012 examples.desktop
-rw-rw-r-- 1 cybr371 sudo   38 Mar 19 03:43 myfile.txt
cybr371@Ubuntu:~$ getfacl myfile.txt
# file: myfile.txt
# owner: cybr371
# group: sudo
user::rw-
group::rw-
other::r--

cybr371@Ubuntu:~$ su - ben
Password:
ben@Ubuntu:~$ echo This line is from the user ben >> /home/cybr371/myfile.txt
ben@Ubuntu:~$
```

As shown above, writing the message into the file from the user ben was successful because users within the same group as cybr 371 (sudo, in this case) had read AND write permissions, therefore Ubuntu allowed writing to the file from the user ben.

❖ *Question 2.3 [1 Mark]: Login as user "`david`" now and write a command to add a line "`This line is from the user david`" to `myfile.txt` file in `cybr371`'s home directory. (Write the command and explain why the operation is either successful or not).*

```
ben@Ubuntu:~$ su - david
Password:
david@Ubuntu:~$ echo This line is from the user david >> /home/cybr371/myfile.tx
t
-su: /home/cybr371/myfile.txt: Permission denied
david@Ubuntu:~$
```

The attempt to write to the file from the user david was unsuccessful. This is because, as shown in the screenshot for question 2.1, users in the other category only have read permissions to the file, hence why Ubuntu denied the user david permission to write to myfile.txt.

❖ *Question 2.4 [1 Mark] – Write a command to use ACL to **deny** all access (read, write and execute) to `myfile.txt` for user `david`.*

Thanks to this forum for helping me with this question:
https://askubuntu.com/questions/609194/how-do-i-prevent-one-user-in-particular-from-accessing-my-home-directory

```
cybr371@Ubuntu:~$ setfacl -m u:david:000  myfile.txt && getfacl myfile.txt
# file: myfile.txt
# owner: cybr371
# group: sudo
user::rw-
user:david:---
group::rw-
mask::rw-
other::r--
```

❖ *Question 2.5 [1 Mark] – Login as user `masood` and issue a command to read the content of the file `mytext.txt` in the `cybr371`'s home directory. Can the user `masood` read the file? Write the commands and explain the output of the command.*

```
cybr371@Ubuntu:~$ echo This file is to be read by Masood >> test/newfile.txt
cybr371@Ubuntu:~$ cat test/newfile.txt
This file is to be read by Masood
cybr371@Ubuntu:~$ su - masood
Password:
masood@Ubuntu:~$ cat /home/cybr371/test/newfile.txt
This file is to be read by Masood
masood@Ubuntu:~$
```

Note: I echoed a message to the file before reading it as user Masood

Cat command: https://riptutorial.com/ubuntu/example/17617/reading-a-text-file

To read the contents of newfile.txt from cybr371's test directory as the user Masood, I called the cat command which read the entire file and displayed it in the terminal. As shown above, the cat command simply outputs the content of newfile.txt inside the terminal itself.

❖ *Question 2.6 [0.5 Mark]: Write a command to create an ACL for user `mary` with write and execute permissions only on the file `myfile.txt`.*

```
cybr371@Ubuntu:~$ setfacl -m u:mary:wx myfile.txt && getfacl myfile.txt
# file: myfile.txt
# owner: cybr371
# group: sudo
user::rw-
user:david:---
user:mary:-wx
group::rw-
mask::rwx
other::r--
```

***Once done, please make sure to **end the reservation**