CYBR371 Assignment 1

Raashna Chand 300607575

Q1

	setup.sh	grades.xlsx	final/ questions.pdf	final/ solutions.pdf	final/ student000/ answers.docx	final/ student001/ answers.docx
arman	---	rw-	rw-	rw-	r--	r--
ilona	---	rw-	r--	r--	r--	r--
student000	---	---	r--	---	rw-	---
student001	---	---	r--	---	---	rw-
student002	---	---	r--	---	---	---

Q3

I wasn't able to use basic access control for this as it didn't make sense to do so. Each file and directory had a different set of permissions depending on which group the user belonged to, and using chown would not work on individual groups. So I used extended access control lists (setfacl) for all groups and users.

I set-up the directory structure the same way as the case study. In case the users hadn't been created yet, I decided to create the users as well as the groups. I created the directories and set ACLs according to the guidelines.

```
vboxuser@vboxuser:/opt$ ls -al
total 20
drwxr-xr-x   4 root root 4096 Apr 12 20:15 .
drwxr-xr-x  20 root root 4096 Apr 12 19:05 ..
drwxrwxr-x+ 11 root root 4096 Apr 12 19:33 cybr371
-rwxr--r--+  1 root root 3032 Apr 12 20:15 setup.sh
drwxr-xr-x   8 root root 4096 Apr 12 19:29 VBoxGuestAdditions-7.0.14
```

Here we see the output of ls -al for the directory opt. We see that extended ACLs have been used on cybr371 and setup.sh, as indicated by the + sign beside the permissions.

```
root@vboxuser:/opt# getfacl setup.sh
# file: setup.sh
# owner: root
# group: root
user::rwx
group::r--
mask::r--
other::---

root@vboxuser:/opt#
```

```
root@vboxuser:/opt# getfacl cybr371
# file: cybr371
# owner: root
# group: root
user::rwx
group::r-x
group:lecturer:rwx
group:tutor:rwx
group:student:r-x
mask::rwx
other::r-x
```

No one can access setup.sh aside from root.

Everyone except students and others have full permissions for the directory. This makes sure students do not tamper with the file structure.

```
vboxuser@vboxuser:/opt/cybr371$ ls -al
total 44
drwxrwxr-x+ 11 root root 4096 Apr 12 19:33 .
drwxr-xr-x   4 root root 4096 Apr 12 20:15 ..
drwxr-xr-x  95 root root 4096 Apr 12 19:33 assignment1
drwxr-xr-x  95 root root 4096 Apr 12 19:34 assignment2
drwxr-xr-x  95 root root 4096 Apr 12 19:34 final
-rw-rw-r--+  1 root root    0 Apr 12 20:15 grades.xlsx
drwxr-xr-x  95 root root 4096 Apr 12 19:33 lab1
drwxr-xr-x  95 root root 4096 Apr 12 19:33 lab2
drwxr-xr-x  95 root root 4096 Apr 12 19:33 lab3
drwxr-xr-x  95 root root 4096 Apr 12 19:33 lab4
drwxr-xr-x  95 root root 4096 Apr 12 19:33 lab5
drwxr-xr-x  95 root root 4096 Apr 12 19:34 midterm
```

```
root@vboxuser:/opt/cybr371# getfacl grades.xlsx
# file: grades.xlsx
# owner: root
# group: root
user::rw-
group::r--
group:lecturer:rw-
group:tutor:rw-
mask::rw-
other::---
```

An ACL on grades.xlsx. Only the owner (root), lecturers and tutors can edit, not anyone else.

We will explore assignment1 as representative of all the other directories present in cybr371.

```
vboxuser@vboxuser:/opt/cybr371/assignment1$ ls -al
total 380
drwxr-xr-x  95 root root 4096 Apr 12 19:33 .
drwxrwxr-x+ 11 root root 4096 Apr 12 19:33 ..
-rw-rw-r--+  1 root root    0 Apr 12 20:31 questions.pdf
-rw-rw-r--+  1 root root    0 Apr 12 20:31 solutions.pdf
drwxrwx---+  2 root root 4096 Apr 12 19:33 student001
drwxrwx---+  2 root root 4096 Apr 12 19:33 student002
drwxrwx---+  2 root root 4096 Apr 12 19:33 student003
drwxrwx---+  2 root root 4096 Apr 12 19:33 student004
drwxrwx---+  2 root root 4096 Apr 12 19:33 student005
drwxrwx---+  2 root root 4096 Apr 12 19:33 student006
drwxrwx---+  2 root root 4096 Apr 12 19:33 student007
drwxrwx---+  2 root root 4096 Apr 12 19:33 student008
drwxrwx---+  2 root root 4096 Apr 12 19:33 student009
drwxrwx---+  2 root root 4096 Apr 12 19:33 student010
drwxrwx---+  2 root root 4096 Apr 12 19:33 student011
drwxrwx---+  2 root root 4096 Apr 12 19:33 student012
drwxrwx---+  2 root root 4096 Apr 12 19:33 student013
drwxrwx---+  2 root root 4096 Apr 12 19:33 student014
drwxrwx---+  2 root root 4096 Apr 12 19:33 student015
drwxrwx---+  2 root root 4096 Apr 12 19:33 student016
drwxrwx---+  2 root root 4096 Apr 12 19:33 student017
drwxrwx---+  2 root root 4096 Apr 12 19:33 student018
drwxrwx---+  2 root root 4096 Apr 12 19:33 student019
drwxrwx---+  2 root root 4096 Apr 12 19:33 student020
drwxrwx---+  2 root root 4096 Apr 12 19:33 student021
drwxrwx---+  2 root root 4096 Apr 12 19:33 student022
```

We can see in this image that extended ACLs have been applied to all the files and directories. We will look at

the questions and solutions file, student001 and its answers.docx file.

```
root@vboxuser:/opt/cybr371/assignment1# getfacl questions.pdf
# file: questions.pdf
# owner: root
# group: root
user::rw-
group::r--
group:lecturer:rw-
group:tutor:r--
group:student:r--
mask::rw-
other::r--
```

```
root@vboxuser:/opt/cybr371/assignment1# getfacl solutions.pdf
# file: solutions.pdf
# owner: root
# group: root
user::rw-
group::r--
group:lecturer:rw-
group:tutor:r--
mask::rw-
other::---
```

Lecturers can read and write to questions.pdf, tutors and students can only read. Lecturers can read and write to solutions.pdf, tutors can view, but no one else can read it (including students).

```
root@vboxuser:/opt/cybr371/assignment1# getfacl student001
# file: student001
# owner: root
# group: root
user::rwx
user:student001:rwx
group::r-x
group:lecturer:r-x
group:tutor:r-x
mask::rwx
other::---
```

For the directory student001, lecturers and tutors can list the files present, and student001 can also add and delete files to the directory, but no one else can, including other students.

```
root@vboxuser:/opt/cybr371/assignment1/student001# ls -al
total 8
drwxrwx---+  2 root root 4096 Apr 12 19:33 .
drwxr-xr-x  95 root root 4096 Apr 12 19:33 ..
-rw-rw----+  1 root root    0 Apr 12 20:40 answers.docx
root@vboxuser:/opt/cybr371/assignment1/student001#
```

```
root@vboxuser:/opt/cybr371/assignment1/student001# getfacl answers.docx
# file: answers.docx
# owner: root
# group: root
user::rw-
user:student001:rw-
group::r--
group:lecturer:r--
group:tutor:r--
mask::rw-
other::---
```

ACL on answers.docx. Student001 can read and write to file, lecturers and tutors can read contents, but no one else can access the file in any way.

```
vboxuser@vboxuser:/opt/cybr371/assignment1/student001$ umask
0002
```

Umask value is the default 0002, default permissions for the user are rw- for files and rwx for directories, for the groups are rw- for files and rwx for directories, for other are r-- for files and r-x for directories. Might be more appropriate to have a value of 0007 where “others” have no permissions at all.

Q7

	register-patient.sh	check-medication.sh	patients	patients/RickSanchez1950	patients/SummerSmith200
All Doctors	--x	--x	rwx	---	---
drloun	--x	--x	rwx	---	rw-
drlstethosc	--x	--x	rwx	rw-	rw-
All nurses	---	r-s	r-x	---	---
others	---	---	---	---	---

Q8

Again, I wasn't able to use basic access control for this as it didn't make sense to do so. Each file and directory had a different set of permissions depending on which group the user belonged to, and using chown would not work on individual groups. So I used extended access control lists (setfacl) for all groups and users.

I set-up the directory structure the same way as the case study. I created groups, users, directories and set ACLs according to the guidelines.

I made sure that upon creation of a patient file, only the primary doctor would be allowed to edit the patient's file. As there is no way to subsequently check the secondary doctor with a shell script specified in the requirements, I made the assumption that this would be taken care of by the administrator.

In check-medication.sh, I came up with a subpar method to get the names of the primary and secondary doctors. I used the names of the existing doctors in if-else statements that checked if the doctors existed. This will not be suitable should more doctors be put in place.

Regardless, here is what setup.sh does.

```
$ pwd
/opt
$ ls -al
total 20
drwxr-xr-x   4 root root 4096 Apr 13 20:34 .
drwxr-xr-x  20 root root 4096 Apr 12 19:07 ..
-rwxr--r--   1 root root 2773 Apr 13 16:37 setup.sh
drwxr-xr-x   8 root root 4096 Apr 13 15:32 VBoxGuestAdditions-7.0.14
drwxrwx---+  3 root root 4096 Apr 13 20:01 WellingtonClinic
$
```

We see in /opt that we have created WellingtonClinic.

```
$ cd WellingtonClinic
$ ls -al
total 16
drwxrwx---+ 3 root root 4096 Apr 13 20:01 .
drwxr-xr-x  4 root root 4096 Apr 13 20:34 ..
-rw-rwx---+ 1 root root    0 Apr 13 16:38 check-medication.sh
drwxrwx---+ 2 root root 4096 Apr 13 20:01 patients
-rw-rwx---+ 1 root root 1183 Apr 13 20:01 register-patient.sh
$
```

ACLs for all files and directories present.

```
$ getfacl check-medication.sh
# file: check-medication.sh
# owner: root
# group: root
user::rw-
user:administrator:rwx
group::r--
group:doctor:--x
group:nurse:r-x
mask::rwx
other::---
```

```
$ getfacl patients
# file: patients
# owner: root
# group: root
user::rwx
user:administrator:rwx
group::r-x
group:doctor:rwx
group:nurse:r-x
mask::rwx
other::---
```

We see that for check-medication.sh the administrator has full permissions, doctors can execute the file, and nurses can read and execute the file. One cannot execute a file without also having read permissions, and since this is the file nurses use to check medication for a patient, they have this permission. For the patients directory, again administrator has full permissions, but so do doctors while nurses can only list the contents.

```
$ pwd
/opt/WellingtonClinic/patients
$ ls -al
total 20
drwxrwx---+ 2 root root 4096 Apr 13 20:38 .
drwxrwx---+ 3 root root 4096 Apr 13 20:01 ..
-rw-rwx---+ 1 root root   36 Apr 13 20:01 ajsdlasldja123
-rw-rwx---+ 1 root root   99 Apr 13 20:37 RickSanchez1950
-rw-rwx---+ 1 root root  194 Apr 13 20:38 SummerSmith2007
$
```

In the patients directory, we see ACLs applied to the patient files.

```
$ getfacl RickSanchez1950
# file: RickSanchez1950
# owner: root
# group: root
user::rw-
user:administrator:rwx
user:drloun:rw-
group::r--
mask::rwx
other::---

$ getfacl SummerSmith2007
# file: SummerSmith2007
# owner: root
# group: root
user::rw-
user:administrator:rwx
user:drloun:rw-
user:drstethosc:rw-
group::r--
mask::rwx
other::---
```

Rick’s patient file sees Dr Lou Ngevity have read and write permissions. Summer’s has both Dr Lou Ngevity and Dr Stethos Cope. No one else apart from the admin can read and write the file.

```
$ umask
0002
```

Again, the default umask applies to the system.