File System Access Control (ACL)
Assignment 1
CYBR371

Kamonchanok Suban Na Ayudtaya
300471606 SUBANKAMO

Vulnerability Report

Threat 1: File owner allows unauthorised rwx permission to 'Patient' files.

There is a vulnerability in how the permissions of a 'Patient' file is assigned.
Currently, when the register-patient.sh is run - the primary doctor who runs the script is given file owner permissions and they are able to grant rwx access to any user or group.
An Adversary, such as a user in the doctor group who may be motivated by money (e.g offered a bribe to allow access to patient medical history) or revenge could grant any user on the system rwx permission to a 'Patient' file where they are the owner.
This can be performed by running command setfacl -m g:"$maliciousgroup":rwx /opt/WellingtonClinic/Patients/"$filename" or setfacl -m g:"$malicioususer":r-- /opt/WellingtonClinic/Patients/"$filename".
This would violate the confidentiality, availability and integrity constraint of the patients data as the patients personal information can now be accessed, modified or deleted by anyone listed as an assigned doctor without detection or prevention.
Which could mean that the patient's data is distributed to unauthorised parties, modified resulting in inaccurate medical history or even being completely deleted off the system making all medical history lost.
The cost of this would have a negative impact on the patient's emotional well being (e.g loss of trust), physical well being (e.g prescribed wrong medication) and data privacy and may result in monetary and reputational cost for the hospital.

Proposed mitigation:

Option 1: The system could log all file/permissions changes for files within the 'Patients' directory flagging any permissions for users outside of the doctor/sudo group or any user with permissions that are not the primary or assigned doctors and file deletion as suspicious.
This could be done by writing an patient_audit.sh file that saves any changes to a log file that is regularly monitored by the admin and audited.
This would mean that any unauthorised permissions being granted can be traced and appropriately dealt with deterring violation of confidentiality, availability and integrity.
Option 2: The system could back up every file in the Patient Directory this means that if a file is deleted by an adversary we would be able to recover it.
This could be done by writing a script that saves a copy of the patient directory and its content everyday in a place only accessible by the admin.
This would reduce the impact of potential availability violation as the data in the 'Patient' files would be recoverable.
Option 3: The files in the patients directory could be encrypted so if unauthorised users are given access to a patient's file they would not be able to read it.
This could be done by encrypting the information held in each 'Patient' file.
This would reduce the possibility of confidentiality violation as if an unauthorised user was given access to the 'Patient' file they would not be able to read it given they do not have the means to decrypt it.
>All options should be used in combination to reduce the possibility and impact of confidentiality, availability and integrity constraints.

Threat 2: Giving Nurse read permission to all 'Patient' files.

There is a vulnerability in the current permissions of the ‘Patient’ file.
Currently, nurses require read permission to an all ‘Patient’ file as when they run check-medication.sh they must access the ‘Patient’ file to retrieve information.
This means that any nurse, whether motivated by financial gain, curiosity, or revenge, can access and distribute sensitive personal information and medical history by simply running cat /opt/WellingtonClinic/Patients/"$filename".
This would violate the confidentiality constraint of the patient’s data as the patient's personal/sensitive information can now be accessed/distributed by any user in the nurse group.
As a result, patients' emotional well-being, data privacy, and the hospital's monetary and reputational status could all be negatively impacted.

Proposed mitigation:
Option 1: A solution is to separate the patient’s ‘Medication’ information and the ‘Patient’ information into separate directory and file. This new medication directory/file will be where check-medication.sh would retrieve the patient’s medication history and basic information.
To do this we will revoke all nurse permissions in the Patients directory and its files. Then, we will create a new directory called "PatientMedication" and modify register-patient.sh to create a txt file "{firstname}{lastname}{birthdate}medication.txt" to be held in the PatientMedication directory.
During an appointment, the ‘Patient’ file will be updated with the full appointment information and the ‘Medication’ file updated with only the medication information.
Only primary/assigned doctors will have read and write permissions to the ‘Patient’ file, whereas the ‘Medication’ file will allow nurses to read information and primary/assigned doctors to have write permissions.
This will mean that nurses only have read permission to data that they require and is not private or sensitive to the patient. Which means that there is a reduced possibility of confidentiality being violated.
Option 2:
The ‘Patient’ file can be encrypted, This will mean the contents of the file cannot be read without decryption so we can allow nurses read permission to the ‘Patient’ file without concern of sensitive information being exposed.
To do this we will need to modify register-patient.sh to encrypt the content of the ‘Patient’ files created, create a sh script that encrypt appointment information and add it to the ‘Patient’ file, modify check-medication.sh to decrypt the ‘Patient’ file, modify searchpatient.sh to decrypt the ‘Patient’ file and create a script decrypt the ‘Patient’ file to allow doctors to read its contents.
This will mean that although nurses will have read permission of the ‘Patient’ file they will not be able to directly read the content as it would require decryption. This would secure the confidentiality of the patients data given that nurses are unable to decrypt the file.
>Option 1 may be a better choice due to the sensitivity of the system's requirements. While encrypting the ‘Patient’ file can provide an extra layer of security, there are many ways in which the file content could be lost. For instance, forgetting the passphrase or losing the private key could lead to the loss of the patient's medical history, which would be irrecoverable. Such a loss could have significant and negative consequences for the hospital, including the possibility of inappropriate treatments being performed on patients. Therefore, it may be better to choose Option 1 over encryption to ensure the safekeeping of patient data.

Threat 3: Unavailable data due to human error in ‘Patient’ files fields.

There is a vulnerability in how the permissions of a primary and assigned doctor is assigned to the ‘Patient’ file when register-patient.sh is run.

Currently, the doctor running the script inputs the primary and assigned doctor fields held in the ‘Patient’ file. The information in the ‘Patient’ file is used by searchpatient.sh to retrieve patient’s information when the script is run. If the user name of the primary or assigned doctor is incorrect or empty, no results are returned. This can lead to doctors being unable to access the information they need about their patient/s.

An Adversary, such as a user in the doctor group who may be motivated by money (e.g bribery) or revenge could intentionally omit the input doctors names when registering patients resulting in no results being returned when searchpatient.sh is executed by the primary or assigned doctor.

This violates the availability constraint of the patient's data, as the authorised users cannot access the patient's personal information when required.

Which could mean that the patient's data could be lost as doctors may not be able to find the files of their patients. This could lead to the loss of the patient's data, which would have a negative impact on their emotional and physical well-being, as they may lose trust in the hospital or be prescribed incorrect medication/treatments. Furthermore, it may result in monetary and reputational costs for the hospital.

Proposed mitigation:

Option 1: A potential solution is to automatically set the primary doctor field using $(whoami) when the ‘Patient' file is created, and then update the assigned doctor field when appointments are made.

To implement this solution, we can remove the "doctor" user input from register-patient.sh and create a new script called createappointment.sh. This script would only be run by a patient's primary doctor and would check if the primary doctor is available for an appointment. If they are available, no modifications are made to the ‘Patient' file. If they are not available, the script would prompt the doctor for an assigned doctor username, check if the assigned doctor exists, and modify the ‘Patient' file permissions to allow the assigned doctor read and write access. The assigned doctor username would also be appended to the assigned doctor field in the ‘Patient' file.

By ensuring that the primary and assigned doctor fields are accurately populated with the correct username and are not empty, we can ensure that searchpatient.sh returns accurate and available results for doctors with patients. This solution would help maintain the availability of patient data.

Option 2: A potential solution is to modify the register-patient.sh script to set the primary doctor field to $(whoami) and validate the assigned doctor input against the system users to ensure that the user exists.

We can do this by modifying register-patient.sh to add a check for the user's input of the assigned doctors field.

This would ensure that the input of register-patient.sh is accurate and not empty but does not ensure that any future modification is accurate as assigned doctors are expected to be added after the ‘Patient' file is created. Which means we can not be sure that when an appointment is made the ‘Patient' file will be updated with a valid assigned doctor (could be empty or not have correct user name).

This would mean that the data will be available for primary doctors when they run searchpatient.sh but does not ensure availability for assigned doctors.

>Option 1 is better as it ensures the availability of data for both primary and assigned doctors assuming assigned doctors are populated after a patient is registered.

Task 2

	register-patient.sh	searchpatient.sh	check-medication.sh	patients folder	WellingtonClinic folder	Masood Mansoori 2001 file	LanceBourne1970 file
All Doctors	r-x	r-x	-	rwx	--x	-	-
DrMaryT	r-x	r-x	-	rwx	--x	rw-	rw-
DrMandyS	r-x	r-x	-	rwx	--x	-	rw-
DrEliM	r-x	r-x	-	rwx	--x	-	-
All Nurses	-	-	r-x	--x	--x	r--	r–

Assumption

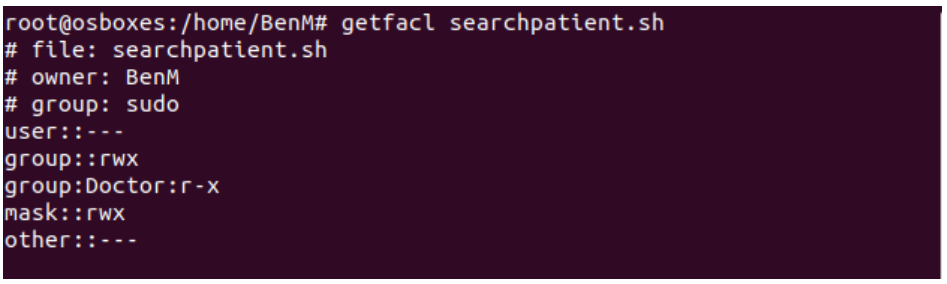
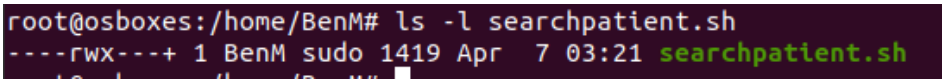
>This ACL is based on the entire system description, the assignment mentioned primary and assigned doctors are able to modify a patient's file therefore we allow them rw- permissions to patient files where they are primary or assigned doctors.

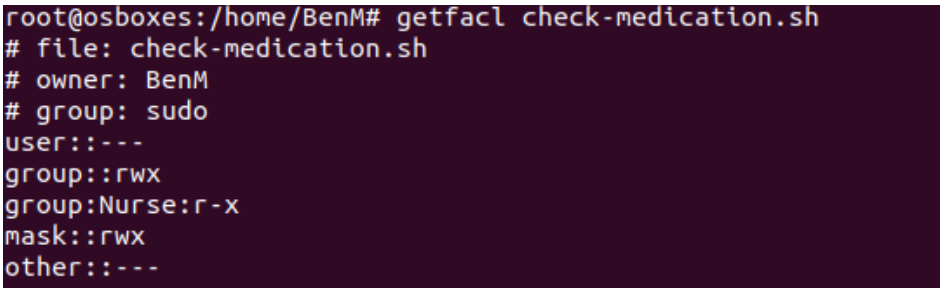
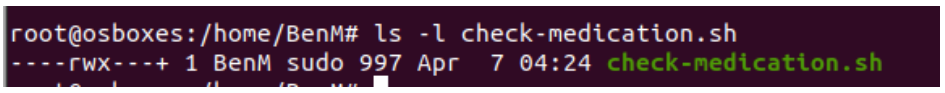
If this ACL is only based only on the described sh scripts then primary and assigned doctors would only have r-- permissions to a patient's file.

>Nurses are able to read the entire patient's file which contradicts the requirement that nurses should not have access to the patient's basic information nor the entire medical records.

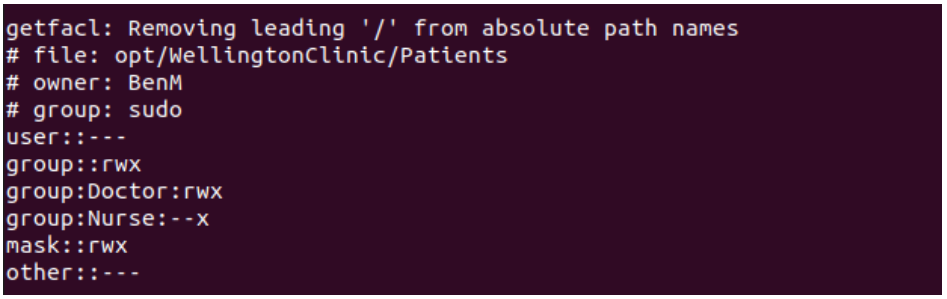
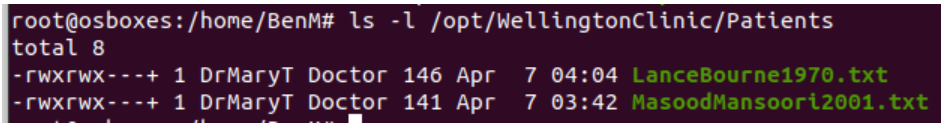
Given that medication information and required patients are held in the patient file there is no way to give them partial access to the file.

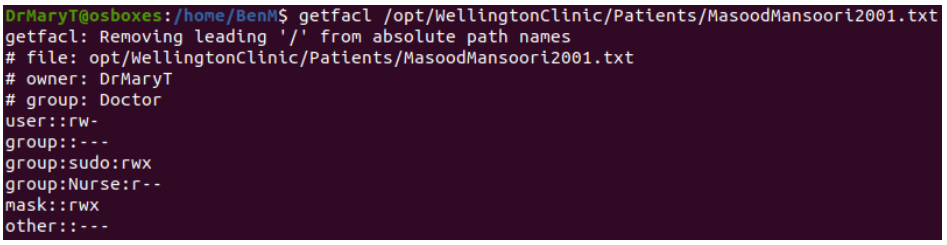
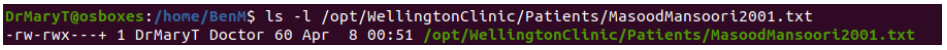
Task 3

/home/BenM/searchpatient.sh	
Set Permissions	chown BenM:sudo /home/BenM/searchpatient.sh chmod 070 /home/BenM/searchpatient.sh setfacl -m g:Doctor:r-x /home/BenM/searchpatient.sh
getfacl	
Ls -l	

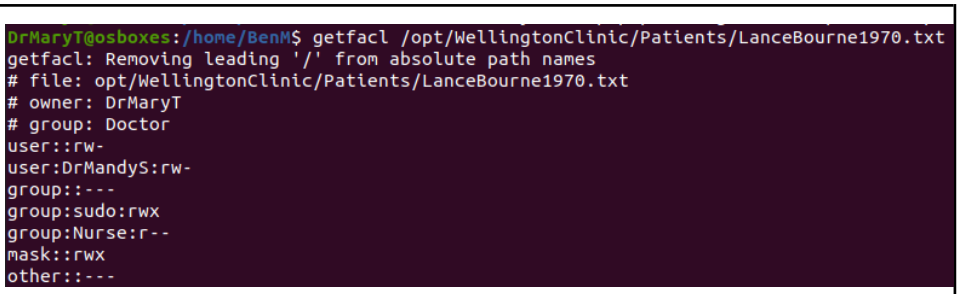
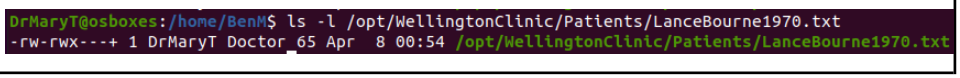
/home/BenM/check-medication.sh	
Set Permissions	chown BenM:sudo /home/BenM/check-medication.sh chmod 070 /home/BenM/check-medication.sh setfacl -m g:Nurse:r-x /home/BenM/check-medication.sh
getfacl	
Ls -l	

/opt/WellingtonClinic/Patients	
Set Permissions	chown BenM:sudo /opt/WellingtonClinic/Patients chmod 070 /opt/WellingtonClinic/Patients setfacl -m g:Doctor:rwx /opt/WellingtonClinic/Patients setfacl -m g:Nurse:--x /opt/WellingtonClinic/Patients

getfacl	
Ls -l	

/opt/WellingtonClinic/Patients/MasoodMansoori2001.txt	
Set Permissions	chmod 600 /opt/WellingtonClinic/Patients/MasoodMansoori2001.txt setfacl -m g:sudo:rw- /opt/WellingtonClinic/Patients/MasoodMansoori2001.txt setfacl -m g:Nurse:r-- /opt/WellingtonClinic/Patients/MasoodMansoori2001.txt
getfacl	
Ls -l	

/opt/WellingtonClinic/Patients/LanceBourne1970.txt	
Set Permissions	chmod 600 /opt/WellingtonClinic/Patients/LanceBourne1970.txt setfacl -m u:DrMandyS:rw- /opt/WellingtonClinic/Patients/LanceBourne1970.txt setfacl -m g:sudo:rwx /opt/WellingtonClinic/Patients/LanceBourne1970.txt setfacl -m g:Nurse:r-- /opt/WellingtonClinic/Patients/LanceBourne1970.txt

getfacl	 DrMaryT@osboxes:/home/BenM$ getfacl /opt/WellingtonClinic/Patients/LanceBourne1970.txt getfacl: Removing leading '/' from absolute path names # file: opt/WellingtonClinic/Patients/LanceBourne1970.txt # owner: DrMaryT # group: Doctor user::rw- user:DrMandyS:rw- group::--- group:sudo:rwx group:Nurse:r-- mask::rwx other::---
Ls -l	 DrMaryT@osboxes:/home/BenM$ ls -l /opt/WellingtonClinic/Patients/LanceBourne1970.txt -rw-rwx---+ 1 DrMaryT Doctor 65 Apr 8 00:54 /opt/WellingtonClinic/Patients/LanceBourne1970.txt

Appendix

Q1.) create-directory-staff.sh

```
[sudo] password for osboxes:
root@osboxes:/home/osboxes# ./create-directory-staff.sh
uid=1001(BenM) gid=27(sudo) groups=27(sudo)
uid=1002(DrMaryT) gid=1001(Doctor) groups=1001(Doctor)
uid=1003(DrMandyS) gid=1001(Doctor) groups=1001(Doctor)
uid=1004(DrEliM) gid=1001(Doctor) groups=1001(Doctor)
uid=1005(LuciaB) gid=1002(Nurse) groups=1002(Nurse)
uid=1006(PhilM) gid=1002(Nurse) groups=1002(Nurse)
```

```
getfacl: Removing leading '/' from absolute path names
# file: opt/WellingtonClinic
# owner: BenM
# group: sudo
user::---
group::rwx
group:Doctor:--x
group:Nurse:--x
mask::rwx
other::---

getfacl: Removing leading '/' from absolute path names
# file: opt/WellingtonClinic/Patients
# owner: BenM
# group: sudo
user::---
group::rwx
group:Doctor:rwx
group:Nurse:--x
mask::rwx
other::---
```

Q4.) register-patient.sh

```
DrMaryT@osboxes:/home/BenM$ ./register-patient.sh
Enter the following information about the patient:
First name: Masood
Last name: Mansoori
Year of birth: 2001
Phone number: 081039475
Email: masood.mans@mail.com
Doctor(s) (~primaryDoctor,#assignedDoctor(s)...): ~DrMaryT
Creating patient file...
getfacl: Removing leading '/' from absolute path names
# file: opt/WellingtonClinic/Patients/MasoodMansoori2001.txt
# owner: DrMaryT
# group: Doctor
user::rw-
group::---
group:sudo:rwx
group:Nurse:r--
mask::rwx
other::---
```

```
DrMaryT@osboxes:/home/BenM$ cat /opt/WellingtonClinic/Patients/MasoodMansoori2001.txt
Masood,Mansoori,2001,081039475,masood.mans@mail.com,~DrMaryTDrMaryT@osboxes:/home/BenM$
```

```
DrMaryT@osboxes:/home/BenM$ ./register-patient.sh
Enter the following information about the patient:
First name: Lance
Last name: Bourne
Year of birth: 1970
Phone number: 0543836456
Email: lancb@outlook.com
Doctor(s) (~primaryDoctor,#assignedDoctor(s)...): ~DrMaryT,#DrMandyS
Creating patient file...
```

```
getfacl: Removing leading '/' from absolute path names
# file: opt/WellingtonClinic/Patients/LanceBourne1970.txt
# owner: DrMaryT
# group: Doctor
user::rw-
user:DrMandyS:rw-
group::---
group:sudo:rwx
group:Nurse:r--
mask::rwx
other::---

DrMaryT@osboxes:/home/BenM$ cat /opt/WellingtonClinic/Patients/LanceBourne1970.txt
Lance,Bourne,1970,0543836456,lancb@outlook.com,~DrMaryT,#DrMandySDrMaryT@osboxes:/home/BenM$
```

Q5.) searchpatient.sh

```
DrMaryT@osboxes:/home/BenM$ ./searchpatient.sh
Doctor Patients
DrMaryT Lance Bourne, Masood Mansoori
```

```
DrMandyS@osboxes:/home/BenM$ ./searchpatient.sh
Doctor Patients
DrMandyS Lance Bourne
```

```
DrEliM@osboxes:/home/BenM$ ./searchpatient.sh
Doctor Patients
DrEliM -
```

Q6.) check-medication.sh

```
LuciaB@osboxes:/home/BenM$ ./check-medication.sh
Enter patient's first name: Lance
Enter patient's last name: Bourne
Enter patient's year of birth: 1970
Patient   Primary Doctor  Assigned Doctor(s)
Lance Bourne    DrMaryT    DrMandyS

Date of Visit Attended Doctor Medication Dosage
10/5/2021 DrMaryT ibprufen 4 per day
04/3/2022 DrMandyS Ivermectin 2 per day
07/9/2022 DrMaryT vitaminC 1000mg per day
```

```
LuciaB@osboxes:/home/BenM$ ./check-medication.sh
Enter patient's first name: Masood
Enter patient's last name: Mansoori
Enter patient's year of birth: 2001
Patient   Primary Doctor  Assigned Doctor(s)
Masood Mansoori    DrMaryT

Date of Visit Attended Doctor Medication Dosage
11/2/2021 DrMaryT scratchicilin 2 per day
```

Q8.) Implement two of your proposed solutions, and illustrate their effectiveness.

Threat 3: Unavailable data due to human error in 'Patient' files fields.

Option 1:

Q8-Threat3-Updated-register-patient.sh

```
DrMaryT@osboxes:/home/BenM$ ./register-patient.sh
Enter the following information about the patient:
First name: Masood
Last name: Mansoori
Year of birth: 2001
Phone number: 081039475
Email: masood.mans@mail.com
Creating patient file...
```

```
DrMaryT@osboxes:/home/BenM$ getfacl /opt/WellingtonClinic/Patients/MasoodMansoori2001.txt
getfacl: Removing leading '/' from absolute path names
# file: opt/WellingtonClinic/Patients/MasoodMansoori2001.txt
# owner: DrMaryT
# group: Doctor
user::rw-
user:DrMandyS:rw-
group::---
group:sudo:rwx
group:Nurse:r--
mask::rwx
other::---
```

Createappointment.sh

```
DrMaryT@osboxes:/home/BenM$ ./create-appointment.sh
Enter patient first name: Masood
Enter patient last name: Mansoori
Enter patient year of birth: 2001
Enter date of appointment (yyyy-mm-dd):
Is the primary doctor available for the appointment? (yes/no): no
/opt/WellingtonClinic/Patients/MasoodMansoori2001.txt
Enter assigned doctor username: DrMandyS
getfacl: Removing leading '/' from absolute path names
# file: opt/WellingtonClinic/Patients/MasoodMansoori2001.txt
# owner: DrMaryT
# group: Doctor
user::rw-
user:DrMandyS:rw-
group::---
group:sudo:rwx
group:Nurse:r--
mask::rwx
other::---
```

```
DrMaryT@osboxes:/home/BenM$ cat /opt/WellingtonClinic/Patients/MasoodMansoori2001.txt
Masood,Mansoori,2001,081039475,masood.mans@mail.com,~DrMaryT,#DrMandyS
DrMaryT@osboxes:/home/BenM$
```

Threat 2: Giving Nurse read permission to all 'Patient' files.

Option 1:

Q8-Threat2-Updated-Register-Patient.sh

```
DrMaryT@osboxes:/home/BenM$ ./register-patient.sh
Enter the following information about the patient:
First name: marina
Last name: suban
Year of birth: 2000
Phone number: m
Email: m
Doctor(s) (~primaryDoctor,#assignedDoctor(s)...): ~DrMaryT,#DrMandyS
Creating patient file...
getfacl: Removing leading '/' from absolute path names
# file: opt/WellingtonClinic/PatientMedication/marinasuban2000medication.txt
# owner: DrMaryT
# group: Doctor
user::-w-
user:DrMandyS:-w-
group::---
group:sudo:rwx
group:Nurse:r--
mask::rwx
other::---

getfacl: Removing leading '/' from absolute path names
# file: opt/WellingtonClinic/Patients/marinasuban2000.txt
# owner: DrMaryT
# group: Doctor
user::rw-
user:DrMandyS:rw-
group::---
group:sudo:rwx
mask::rwx
other::---

/opt/WellingtonClinic/PatientMedication/marinasuban2000medication.txt created successfully!
/opt/WellingtonClinic/Patients/marinasuban2000.txt created successfully!
```

```
LuciaB@osboxes:/home/BenM$ cat /opt/WellingtonClinic/Patients/marinasuban2000.txt
cat: /opt/WellingtonClinic/Patients/marinasuban2000.txt: Permission denied
```

```
LuciaB@osboxes:/home/BenM$ cat /opt/WellingtonClinic/PatientMedication/marinasuban2000medication.txt
marina,suban,~DrMaryT,#DrMandySLuciaB@osboxes:/home/BenM$
```

```
DrMaryT@osboxes:/home/BenM$ echo -e "11/2/2021,DrMaryT,scratchicilin,2 per day" >> /opt/WellingtonClinic/P
atientMedication/marinasuban2000medication.txt
```

```
LuciaB@osboxes:/home/BenM$ cat /opt/WellingtonClinic/PatientMedication/marinasuban2000medication.txt
marina,suban,~DrMaryT,#DrMandyS
11/2/2021,DrMaryT,scratchicilin,2 per day
```

Q8-Threat2-Updated-checkmedication.sh

```
LuciaB@osboxes:/home/BenM$ ./check-medication.sh
Enter patient's first name: marina
Enter patient's last name: suban
Enter patient's year of birth: 2000
Patient   Primary Doctor  Assigned Doctor(s)
marina suban    DrMaryT   DrMandyS

Date of Visit Attended Doctor Medication Dosage
11/2/2021,DrMaryT,scratchicilin,2 per day
```

Q9.) audit.sh

```
root@osboxes:/home/BenM# ./audit.sh
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libauparse0
Suggested packages:
  audispd-plugins
The following NEW packages will be installed:
  auditd libauparse0
0 upgraded, 2 newly installed, 0 to remove and 379 not upgraded.
Need to get 246 kB of archives.
After this operation, 864 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us.archive.ubuntu.com/ubuntu focal/main amd64 libauparse0 amd64 1:2.8.5-2ubuntu6 [49.8 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu focal/main amd64 auditd amd64 1:2.8.5-2ubuntu6 [196 kB]
Fetched 246 kB in 3s (95.3 kB/s)
Selecting previously unselected package libauparse0:amd64.
(Reading database ... 142607 files and directories currently installed.)
Preparing to unpack .../libauparse0_1%3a2.8.5-2ubuntu6_amd64.deb ...
Unpacking libauparse0:amd64 (1:2.8.5-2ubuntu6) ...
Selecting previously unselected package auditd.
Preparing to unpack .../auditd_1%3a2.8.5-2ubuntu6_amd64.deb ...
Unpacking auditd (1:2.8.5-2ubuntu6) ...
Setting up libauparse0:amd64 (1:2.8.5-2ubuntu6) ...
Setting up auditd (1:2.8.5-2ubuntu6) ...
Created symlink /etc/systemd/system/multi-user.target.wants/
auditd.service → /lib/systemd/system/auditd.service.
Processing triggers for systemd (245.4-4ubuntu3.15) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.2) ...
========================================
Username Object Operation Date
 interest were events
```

```
root@osboxes:/home/BenM# mkdir /opt/WellingtonClinic/TestDirectory
root@osboxes:/home/BenM# ./audit.sh
Reading package lists... Done
Building dependency tree
Reading state information... Done
auditd is already the newest version (1:2.8.5-2ubuntu6).
0 upgraded, 0 newly installed, 0 to remove and 379 not upgraded.
========================================
Username Object Operation Date
osboxes /opt/WellingtonClinic/TestDirectory mkdir 04/07/2023
```

Q10). extract-user-info.sh

```
root@osboxes:/home/BenM# ./extract-user-info.sh
Enter the username: DrMaryT
Username: DrMaryT
Groups: Doctor
UserID: 1002
Group(s) ID: 1001
Home Directory: /home/DrMaryT
Shadow file?: Yes
Hashing Algorithm Used? SHA-512
Date of last password change: 07/04/2023
root@osboxes:/home/BenM# ./extract-user-info.sh
Enter the username: DrMandyS
Username: DrMandyS
Groups: Doctor
UserID: 1003
Group(s) ID: 1001
Home Directory: /home/DrMandyS
Shadow file?: Yes
Hashing Algorithm Used? SHA-512
Date of last password change: 07/04/2023
root@osboxes:/home/BenM# ./extract-user-info.sh
Enter the username: LuciaB
Username: LuciaB
Groups: Nurse
UserID: 1005
Group(s) ID: 1002
Home Directory: /home/LuciaB
Shadow file?: Yes
Hashing Algorithm Used? SHA-512
Date of last password change: 07/04/2023
```

Q11). find-sguid.sh

```
DrMaryT@osboxes:~$ chmod 4077 DirectorysuidS
DrMaryT@osboxes:~$ chmod 4777 Directorysuids
DrMaryT@osboxes:~$ chmod 2777 Directorysgids
DrMaryT@osboxes:~$ chmod 2707 DirectorysgidS
DrMaryT@osboxes:~$ chmod 4077 DirectorysgidS/suidS
DrMaryT@osboxes:~$ chmod 4777 DirectorysgidS/suids
DrMaryT@osboxes:~$ chmod 2777 DirectorysgidS/sgids
DrMaryT@osboxes:~$ chmod 2707 DirectorysgidS/sgidS
DrMaryT@osboxes:~$ cd /home/BenM
DrMaryT@osboxes:/home/BenM$ ./find-sguid.sh
File or directory | Type | Permission | Permission/Octal | Note
-----------------------------------------------------------------------
find: ‘/home/DrMaryT/DirectorysuidS’: Permission denied
/home/DrMaryT | Directory | drwxr-xr-x | 0755 | -
/home/DrMaryT/.bashrc | File | -rw-r--r-- | 0644 | -
/home/DrMaryT/Directorysuids | Directory | drwsrwxrwx | 4777 | *suspicious
/home/DrMaryT/DirectorysgidS | Directory | drwx--Srwx | 2707 | *suspicious
/home/DrMaryT/DirectorysgidS/suids | File | -rwsrwxrwx | 4777 | *suspicious
/home/DrMaryT/DirectorysgidS/sgids | File | -rwxrwsrwx | 2777 | *suspicious
/home/DrMaryT/DirectorysgidS/sgidS | File | -rwx--Srwx | 2707 | *suspicious
/home/DrMaryT/DirectorysgidS/suidS | File | ---Srwxrwx | 4077 | *suspicious
/home/DrMaryT/.bash_logout | File | -rw-r--r-- | 0644 | -
/home/DrMaryT/Directorysgids | Directory | drwxrwsrwx | 2777 | *suspicious
/home/DrMaryT/.profile | File | -rw-r--r-- | 0644 | -
/home/DrMaryT/DirectorysuidS | Directory | d--Srwxrwx | 4077 | *suspicious
DrMaryT@osboxes:/home/BenM$
```