

Project 1-

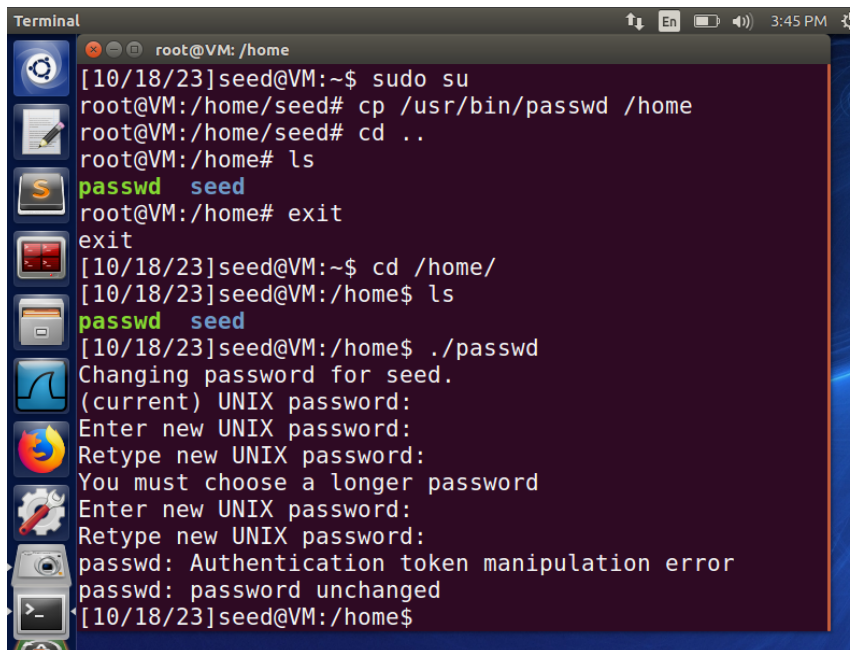
Team members-

1.Shitil Shetty

2.Vishwajeet Kulkarni

3.Pawan Sai Krishna Reddy Kerelly

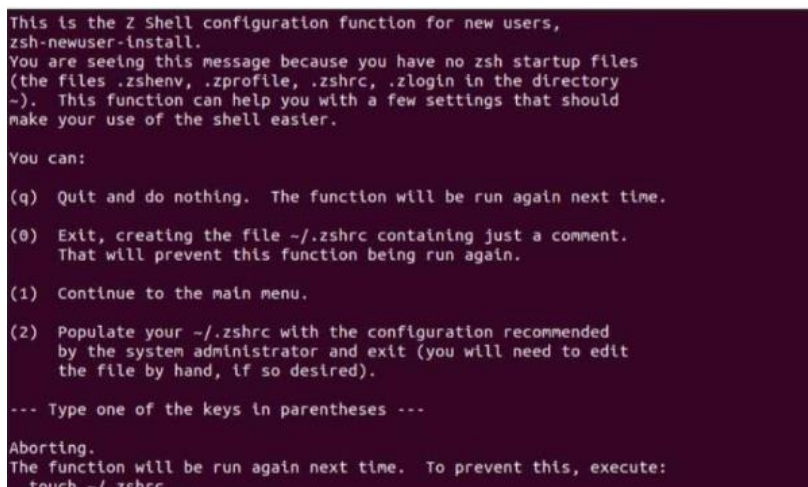
Problem 2-

A terminal window titled 'Terminal' with a dark background and light text. The prompt is 'root@VM: /home'. The user 'seed' runs 'sudo su' to become root. Then, as root, they run 'cp /usr/bin/passwd /home' and 'cd ..' to move to the home directory. They then run 'ls' and see 'passwd seed'. They run 'exit' to return to the 'seed' user prompt. Then they run 'cd /home/' and 'ls', seeing 'passwd seed'. Finally, they run './passwd', which prompts for the current password, then a new password, and then a confirmation. The new password is rejected because it's too short. They try again, but the confirmation fails with 'passwd: Authentication token manipulation error'. The password remains unchanged.

```
Terminal
root@VM: /home
[10/18/23]seed@VM:~$ sudo su
root@VM:/home/seed# cp /usr/bin/passwd /home
root@VM:/home/seed# cd ..
root@VM:/home# ls
passwd seed
root@VM:/home# exit
exit
[10/18/23]seed@VM:~$ cd /home/
[10/18/23]seed@VM:/home$ ls
passwd seed
[10/18/23]seed@VM:/home$ ./passwd
Changing password for seed.
(current) UNIX password:
Enter new UNIX password:
Retype new UNIX password:
You must choose a longer password
Enter new UNIX password:
Retype new UNIX password:
passwd: Authentication token manipulation error
passwd: password unchanged
[10/18/23]seed@VM:/home$
```

Non rooted users are unable to change passwords as it is not recommended. The system restricts root users from modifying their passwords.

(b1)

A terminal window showing the output of the 'zsh-newuser-install' function. It explains that the user has no zsh startup files and offers options to either quit, exit, continue, or populate the .zshrc file with recommended settings. The user is prompted to type one of the keys in parentheses. The terminal shows the user typing '(q)' and the function aborting.

```
This is the Z Shell configuration function for new users,
zsh-newuser-install.
You are seeing this message because you have no zsh startup files
(the files .zshenv, .zprofile, .zshrc, .zlogin in the directory
~). This function can help you with a few settings that should
make your use of the shell easier.

You can:

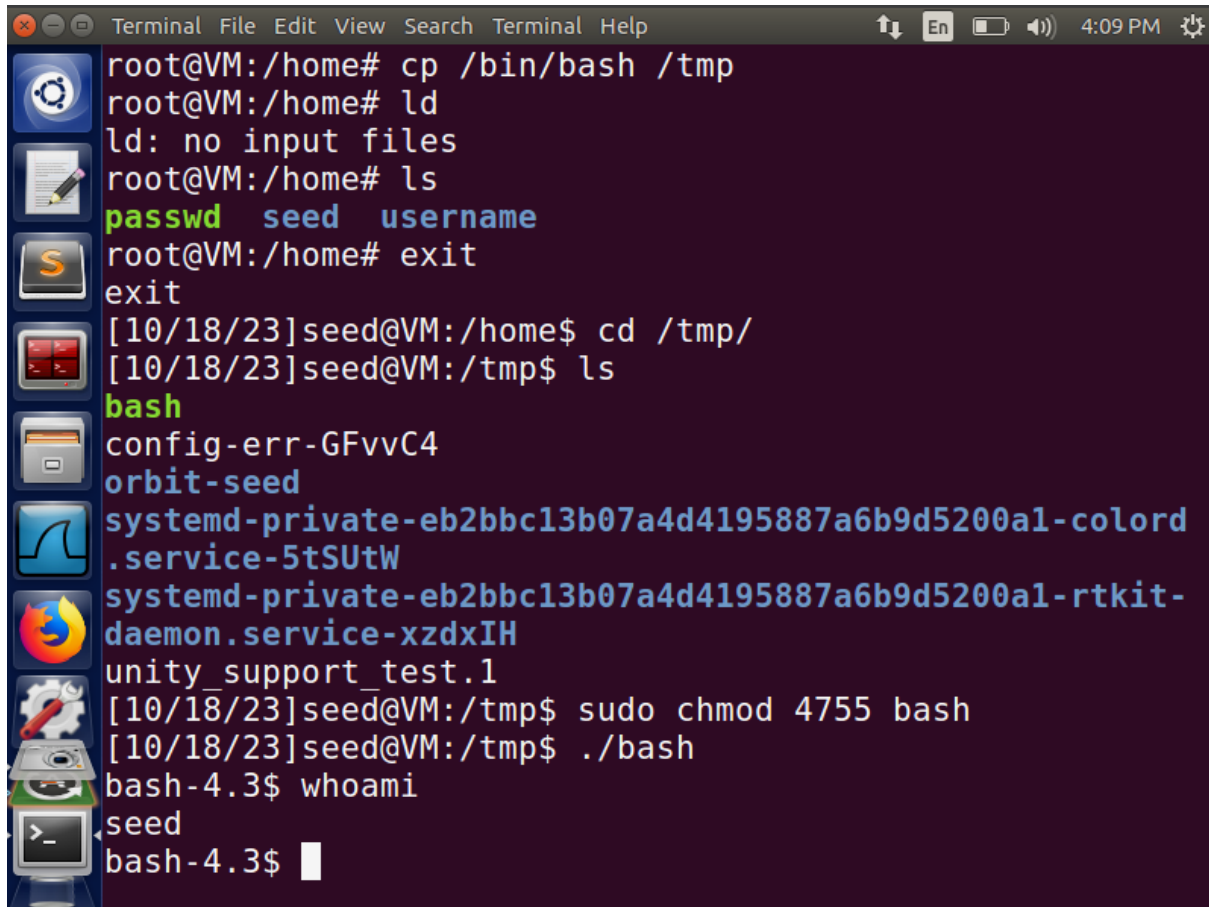
(q) Quit and do nothing. The function will be run again next time.
(0) Exit, creating the file ~/.zshrc containing just a comment.
    That will prevent this function being run again.
(1) Continue to the main menu.
(2) Populate your ~/.zshrc with the configuration recommended
    by the system administrator and exit (you will need to edit
    the file by hand, if so desired).

--- Type one of the keys in parentheses ---

Aborting.
The function will be run again next time. To prevent this, execute:
touch ~/.zshrc
```

When we execute zsh with SetUID it verifies the owner details. If we execute it as seed we gain access, with root privileges.

(b2)



```
root@VM:/home# cp /bin/bash /tmp
root@VM:/home# ld
ld: no input files
root@VM:/home# ls
passwd seed username
root@VM:/home# exit
exit
[10/18/23]seed@VM:/home$ cd /tmp/
[10/18/23]seed@VM:/tmp$ ls
bash
config-err-GFvvC4
orbit-seed
systemd-private-eb2bbc13b07a4d4195887a6b9d5200a1-colord
.service-5tSUtW
systemd-private-eb2bbc13b07a4d4195887a6b9d5200a1-rtkit-
daemon.service-xzdxIH
unity_support_test.1
[10/18/23]seed@VM:/tmp$ sudo chmod 4755 bash
[10/18/23]seed@VM:/tmp$ ./bash
bash-4.3$ whoami
seed
bash-4.3$
```

Bash verifies the information and permissions of the users regardless of whether they have root access and 4755 permissions. However it's important to note that this functionality is not supported in zsh.

(c1)

No. The system can be easily manipulated. So, giving SetUID permission must be done carefully.

`PATH= /home/seed/tmp1:$PATH`

`touch ls`

`nano ls`

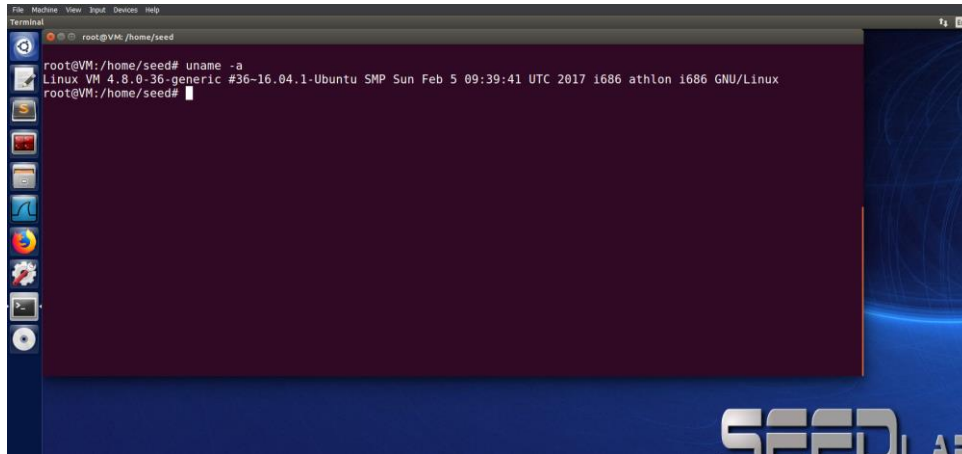
`chmod+x ls`

`bad_ls`

(c2)

The permissions set by the executor will not be preserved when using `/bin/bash`. The commands executed are specific to the seed level. Do not affect the root level

(c3)



```
File Edit View Input Devices Help
Terminal
root@VM: /home/seed
root@VM:/home/seed# uname -a
Linux VM 4.8.0-36-generic #36-16.04.1-Ubuntu SMP Sun Feb 5 09:39:41 UTC 2017 i686 athlon i686 GNU/Linux
root@VM:/home/seed#
```

Problem 3-

There's a security vulnerability, in the scenario you described that could allow someone who's on the no fly list to attempt air travel.

Here's a step by step breakdown of how such an attack could occur,How Someone on the No Fly List Could Try to Fly.

1. Creating a Fake Boarding Pass

The attacker could create a boarding pass resembling an HTML page using resources or software. This fabricated boarding pass would include a barcode, passenger information and flight details.

i.e- fake ticket generator online can be used with the same template which airlines uses.

2. Printing the fake Boarding Pass

After creating it the attacker would print out the boarding pass on paper. The printed pass might also feature a barcode representing the reservation since genuine boarding passes usually have two dimensional barcodes.

3. Arriving at the Airport

Once inside the airport the individual would approach the security checkpoint. Present their identification card along with the printed counterfeit boarding pass.

4. Security Check Process

Its possible that security personnel may not have access, to the governments no fly list.

The security personnel will perform their security procedures, such, as verifying the name on the boarding pass with the ID. If the information matches they may allow the individual to proceed.

5. Boarding the Flight

The attacker will proceed through security. Reach the boarding gate. The airline staff will scan the barcode on the boarding pass, at the gate to validate the reservation. If it is a barcode they may grant permission for the person to board the airplane.

Which additional security measures should be implemented in order to eliminate this vulnerability?

To address this vulnerability and prevent individuals, on the no fly list from attempting to fly we can implement security measures

1. Real time integration of the No Fly List

Airlines and airport security should have access to the government maintained no fly list. This will enable identification of individuals on the list during reservation and boarding processes.

2. Enhanced security for boarding passes

Airlines should adopt procedures for creating and distributing boarding passes. To minimize the risk of tickets boarding passes should be digitally. Designed with tamper proof features.

3. Improved barcode security

Two barcodes on boarding passes should incorporate encrypted data and digital signatures to prevent manipulation or forgery attempts.

4. Implementation of verification

Utilizing biometric authentication methods like fingerprint or facial recognition can help validate passengers identities reducing reliance on printed boarding passes at security checkpoints.

These additional security measures aim to tackle the mentioned vulnerability while enhancing safety protocols, in air travel.

5. Implementing Random Security Checks

In areas, throughout the airport it is important to conduct security checks as a deterrent against individuals attempting to exploit any vulnerabilities. This includes conducting inspections at the exit gate.

6. Enhancing Passenger Screening

It is crucial to employ advanced passenger screening techniques that consider factors beyond a persons identity, such as their travel history in order to identify potential security concerns.

7.Coordination Among Agencies

To improve security measures and successfully counter threats, strong coordination and information exchange between government organizations, airlines, and airports are required.

By implementing these security measures and ensuring communication among all stakeholders we can significantly reduce the identified vulnerability and minimize the risk of individuals, on the no fly list attempting to board flights.