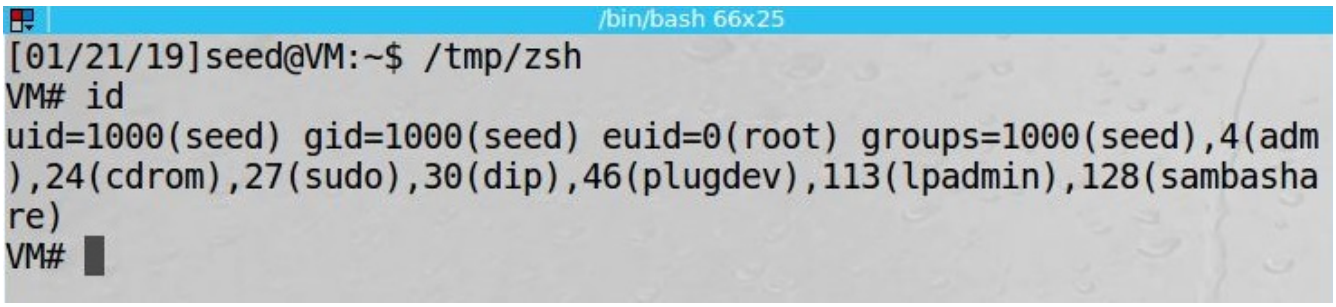
Ashutosh Mishra
1601CS06
Secure System Design – Assignment 1

Question 1: passwd, chsh, su, sudo need to be SetUID programs because when we use these commands, we are modifying important files which only root has write access to. We do not want to give root privilege to the user but we want to give root privilege to the programs. Hence, these programs need to be SetUID. If they weren't SetUID, we would have to either request the root for each of the commands, or grant root privileges to the user.

Before copying	After copying
passwd: changes the password	passwd: Authentication token manipulation error
chsh: changes the shell	chsh: Cannot change ID to root.
su: switches to root user	su: Authentication failure
sudo: executes command as root	sudo: /home/seed/ssd/sudo must be owned by uid 0 and have the setuid bit set

After copying, no program runs correctly. All programs throw some sort of authentication failure.

Question 2(a): Yes, after copying /bin/zsh to /tmp/zsh, and executing as normal user, we get a root shell.

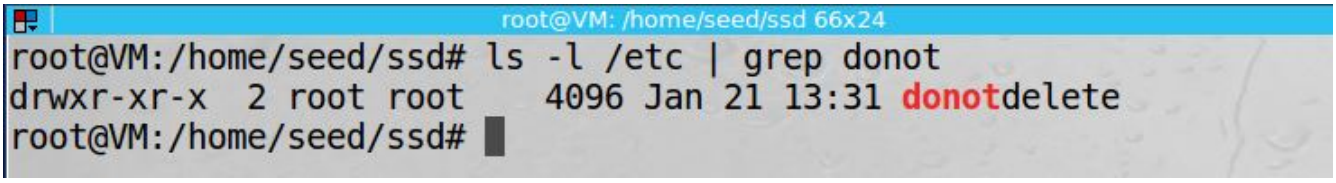


Question 2(b): No, this time we do not get root shell as normal user.

This is because bash can detect when it has been started as a SetUID root. It resets our EUID to our RUID and we do not get root privilege.

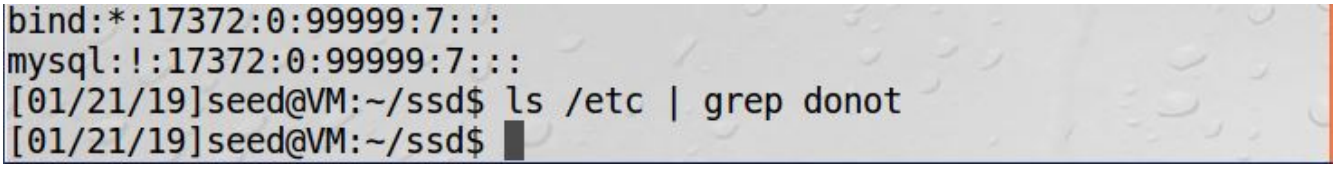
Question 3(a): No, this program is not safe. Let us say that root created a directory called "donotdelete" in /etc/. If Bob runs the following command, he can delete this directory:

Before Command:



Run command: ./a.out "/etc/shadow;rmdir /etc/donotdelete"

After Command:

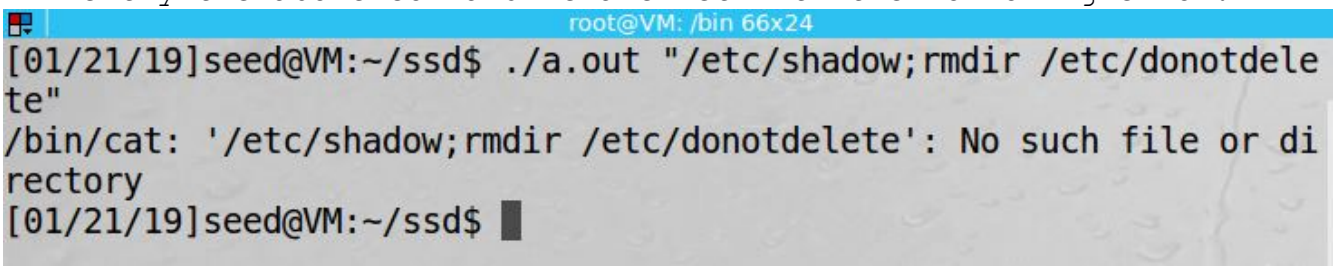


Although we can read the /etc/shadow file, we can also delete a directory which is created by root.

This is because the input is not sanitized and since the program is SetUID with root as user, Bob can basically execute any command.

Question 3(b): If we put q = 1 in the program, this attack does not work anymore. This attack fails to work because execve sanitizes its inputs and does not allow extra commands to be executed.

If we try the above command we are met with the following error:



Question 4: After resetting the sh link to bash again, when we try the same attack, it does not work. It throws the following error:

bash detects that it has been started SUID root (UID!=EUID) and uses its root power to throw this power away, resetting EUID to UID. Thus, we do not get root privileges and instead get only our user privileges.