Sri Lanka Institute of Information Technology
Introduction to Cyber Security - IE2022
Lab Sheet 9
Year 2, Semester 1

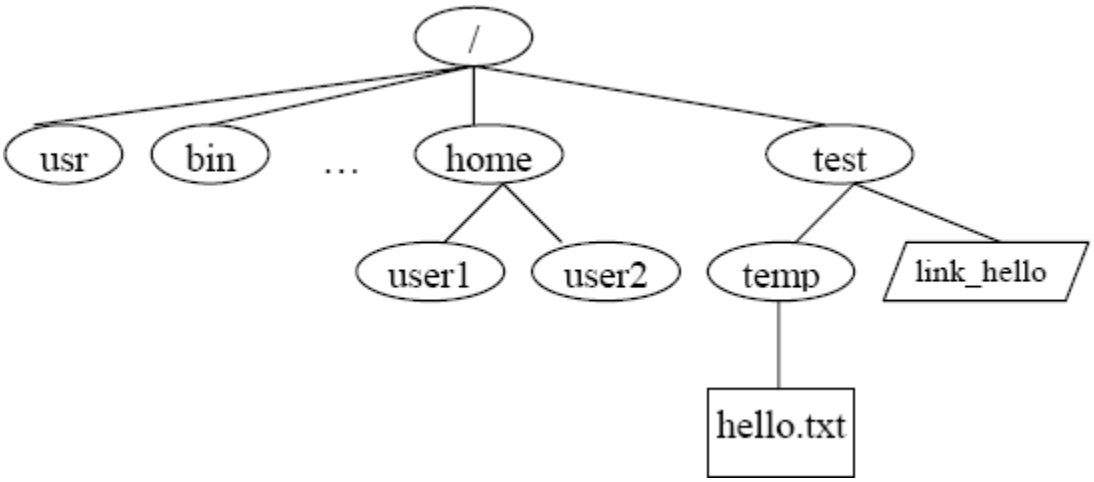
Access Control

Required Equipment/Software

- **Kali Linux**

Unix

Unix File Hierarchy



Ownership and Permissions

Permission Bits

Extra			Owner			Group			Others		
su	sg	t	r	w	x	r	w	x	r	w	x

Sri Lanka Institute of Information Technology
Introduction to Cyber Security - IE2022
Lab Sheet 9
Year 2, Semester 1

Unix Lab Procedures

1. **Setting up File Structure and User Space.**

The objective of this exercise is to set up the file hierarchy structure and the users that are required for the exercises in this section. The su command is used to switch users.

a. Login as root

b. Use useradd command to create two new users user1 and user2 as follows:

- useradd -m user1 -g users -p user1
- useradd -m user2 -g users -p user2

c. Check user information with the id command. Note the uid, gid for each output.

- id user1
- id user2

d. Create a directory structure

- mkdir /test
- mkdir /test/temp

e. Switch user roles as user1 and then back to root using the su command

- whoami
- su user1
- su OR su root

f. Create a new file as root user and change group ownership as well as user ownership of the file.

- touch /home/user2/HelloWorld
- ls -l /home/user2/HelloWorld (observe owner and group)
- chgrp users /home/user2/HelloWorld
- chown user2:users /home/users/HelloWorld
- ls -l /home/user2/HelloWorld (observe owner and group)

2. ***Questions.***

- ***Explain what chgrp and chown do?***
- ***What do -g and -p options mean?***

3. **Differences in File and Folder Permissions.**

The objective of the following exercises would be to see the differences in file and folder permissions. The chmod command will be used to change file and directory permission to demonstrate the slight differences in permissions for files and directories.

a. Observe the result of ls and cd commands

- cd /
- ls -l
- ls -al /home
- Switch to user1 using su user1
- ls -al /home/user2 (Can you list directory?)
- cd /home/user2 (Can you change directory?)

b. Change directory permissions of user2 directory and try again as user1.

- su root
- chmod 740 /home/user2
- (Can you list or change directory?)
- su root
- chmod 750 /home/user2
- (Can you list or change directory?)
- touch /home/user2/hello12.txt(Can you create a new file?)
- su root
- chmod 770 /home/user2
- su user1
- (Can you create a new file?)
- ls -l /home/user2

4. Question.

- ***What are the directory permissions for user1, user2 ?***

Sri Lanka Institute of Information Technology
Introduction to Cyber Security - IE2022
Lab Sheet 9
Year 2, Semester 1

5. Alternative Syntax for chmod Command.

You are expected to learn both the ways to use chmod. The access permissions for the file hello.txt is to set the su bit only, allow all access permissions to owner, read and execute rights to the group and only read rights to others. In other works the 12 bit permission required on the file hello.txt is as follows: "100 111 101 100." This can be achieved in several ways using chmod command:

- chmod 4754 hello.txt
- chmod u+srwx,g+rx,o+r hello.txt
- chmod u=srwx,g=rx,o=r hello.txt

6. New Text Files and Linking Files.

Unix supports two kinds of link files--a hard link and a symbolic link. A hard link is a file with the actual address space of some ordinary file's data blocks. A symbolic link is just a reference to another file. It contains the pathname to some other file.

a. In the /test/temp/ directory, as root user, create a new text file ("hello") and fill it with some text using touch, pico, vi etc.
b. Create a link link_hello in the test folder pointing to hello.txt in the temp folder (refer to file structure in introduction)

- cd /
- ln -s /test/temp/hello /test/link_hello
- ***Is there any difference in file permissions of link_hello and hello?***
- ***cat /test/link_hello What is the output?***

Sri Lanka Institute of Information Technology
Introduction to Cyber Security - IE2022
Lab Sheet 9
Year 2, Semester 1

7. Default file permissions and Group Access Control.

Whenever a new file is created using the C program,default permissions can be assigned to it. UNIX system allows the user to filter out unwanted permissions by default. This default setting can be set by the user using the umask command. It is a system call that is also recognized by the shell. The command takes the permissions set during file creation and performs a bitwise AND to the bitwise negation of mask value. Some common umask values are 077 (only user has permissions), 022 (only owner can write), 002 (only owner and group members can write), etc.

a. In a terminal window, make sure you are a root user. If not the root user, then switch back to root user (use your password to switch).

b. Use umask command to check the current mask permission and assign a new mask.

- umask
- ***What is the current mask? How is it interpreted? (try umask -S or the man pages)***
- cd /test
- touch testmask1
- ls -al
- ***What are the permissions of the file testmask1***
- umask 0077
- touch testmask2
- ***What are the permissions of the file testmask2***
- ***What is the effect of setting mask value to 0000?***

8. Setuid Bit, Setgid Bit and Sticky Bit.

As explained in the background above, the highest three bits of the permission bits represent the setuid bit, setgid bit and the sticky bit. If the setuid bit is set then the uid will always be set to the owner of the file during execution. If the setuid bit is not set then the uid will be the user who executes the process. Similarly, if the setgid bit is set then the gid will be set to the group that owns the file during execution. If the setgid bit is not set then the gid will be the group that executes the process. The sticky bit is set to keep processes in the main memory. In the following exercise, the objective is to demonstrate how processes are affected when the setuid bit is set. The exercise must be begun with root privileges.

- which touch
- ls -l /bin/touch
- chmod 4755 /bin/touch
- ls -l /bin/touch
- ls -l /home/user2
- chmod 700 /home/user2/HelloWorld
- ls -l /home/user2 (observe timestamp and permissions)
- su user1
- touch /home/user2/HelloWorld
- ls -l /home/user2 (observe timestamp)
- su root
- chmod 0755 /bin/touch
- su user1
- touch /home/user2/HelloWorld

9. Question.

- ***Why is permission denied?***

10. **Restore the System.**

After the series of exercises, it is most essential that the system is restored to its normal state so that other students may undertake the exercises again. Below are the series of commands that are expected to restore the system to its original form.

- su root
- umask 0022
- chmod 0755 /bin/touch
- userdel user1
- userdel user2
- rm -rf /home/user1
- rm -rf /home/user2
- rm -rf /test
- rm -rf /home/test/