



ALBUKHARY INTERNATIONAL UNIVERSITY

SCHOOL OF COMPUTING & INFORMATICS

Course Title : OPERATING SYSTEMS
Course Code : CCC 2123

LAB EXERCISE (WEEK 8)

Instructions:

Using the Command Line Interface (CLI) for Linux that we have learned and used so far, please execute the following tasks:

Task 1: Permission Setup

- Create a new directory named "Permission_Lab."
- Inside this directory, create a file named "secure_file.txt."
- Set the permissions to allow the owner full access, the group read-only access, and others no access.

```
haryani@kali: ~/Permission_Lab
File Actions Edit View Help
(haryani@kali)-[~]
$ mkdir Permission_Lab
(haryani@kali)-[~]
$ ls
CLI      Documents  Lab6      OS        Practice  Subdirectory2  fruits
CLI_Lab_Week6  Downloads  Music     Permission_Lab  Public    Templates      newDirectory
Desktop   File_Lab   New_Directory  Pictures     Subdirectory1  Videos        test
(haryani@kali)-[~]
$ cd Permission_Lab
(haryani@kali)-[~/Permission_Lab]
$ touch secure_file.txt
(haryani@kali)-[~/Permission_Lab]
$ chmod 700 securefile.txt
chmod: cannot access 'securefile.txt': No such file or directory
(haryani@kali)-[~/Permission_Lab]
$ chmod 700 secure_file.txt
(haryani@kali)-[~/Permission_Lab]
$ ls -l secure_file.txt
-rwx----- 1 haryani haryani 0 Dec 28 20:33 secure_file.txt
(haryani@kali)-[~/Permission_Lab]
$
```

```
# Create a new directory named "Permission_Lab"
mkdir Permission_Lab
```

```
# Move into the directory
```

```
cd Permission_Lab
```

```
# Create a file named "secure_file.txt"
touch secure_file.txt
```

```
# Set the permissions
chmod 700 secure_file.txt
```

Task 2: Ownership Change

- Change the ownership of "secure_file.txt" to another user on the system.
- Verify the ownership change.

```
(haryani@kali)~[~/Permission_Lab]
$ sudo adduser new --allow-all-names
[sudo] password for haryani:
info: Adding user `new' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `new' (1001) ...
info: Adding new user `new' (1001) with group `new (1001)' ...
info: Creating home directory `/home/new' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for new
Enter the new value, or press ENTER for the default
  Full Name []: Haryani
    Room Number []: 123
    Work Phone []: 1234
    Home Phone []: 12345
    Other []: 123456
Is the information correct? [Y/n] y
info: Adding new user `new' to supplemental / extra groups `users' ...
info: Adding user `new' to group `users' ...

(haryani@kali)~[~/Permission_Lab]
$ sudo chown new:new secure_file.txt

(haryani@kali)~[~/Permission_Lab]
$ ls -l secure_file.txt
-rwx----- 1 new new 0 Dec 28 20:48 secure_file.txt

(haryani@kali)~[~/Permission_Lab]
$
```

Before giving an ownership, we need to know who the user is.

```
#Create new user call "new:
Sudo adduser new - --allow-all-names
Enter password to the system, then enter all required information.
```

```
# Change the ownership of "secure_file.txt" to another user
sudo chown new:new secure_file.txt
```

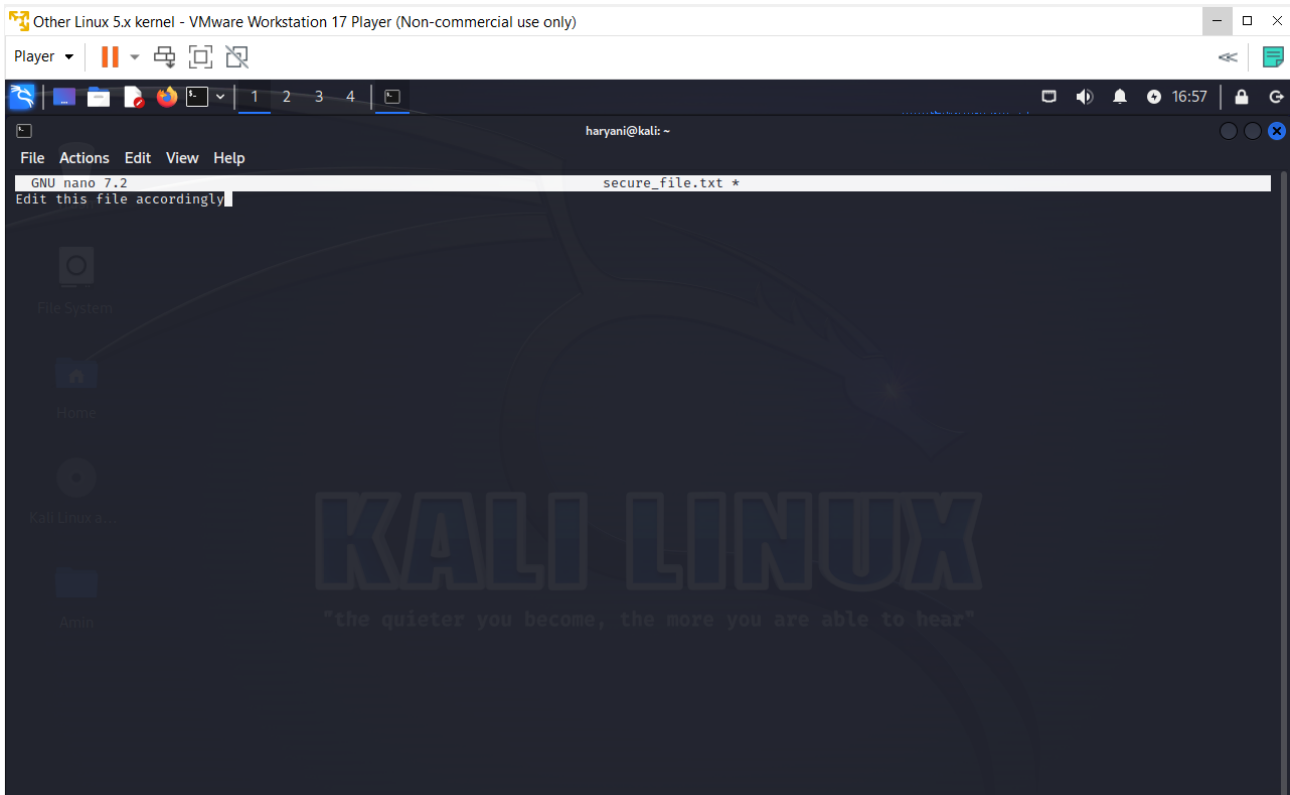
```
# Verify the ownership change
ls -l secure_file.txt
```

Task 3: Permission Testing

- Attempt to edit "secure_file.txt" as the owner and a different user.
- Note the outcomes and any error messages received.

Attempt to edit "secure_file.txt" as the owner
nano secure_file.txt

Attempt to edit "secure_file.txt" as a different user
sudo nano secure_file.txt



*Note: The difference between nano file.txt and sudo nano file.txt lies in the level of permissions and access to the file.

nano file.txt:

- When you use this command without sudo, you are opening the file file.txt in the nano text editor with your current user's permissions.
- You can edit the file and save changes, but you are restricted by the permissions associated with your user account. If the file requires elevated privileges (e.g., it's owned by another user or is in a system directory), you might encounter permission issues.

sudo nano file.txt:

- The sudo command stands for "superuser do" and is used to execute a command with elevated privileges.
- When you use sudo nano file.txt, you are opening the file with superuser or root permissions. This allows you to edit and save changes to files that require higher privileges than your user account possesses.
- Be cautious when using sudo as it grants you elevated access, and modifications made with it can have system-wide consequences.