

LAB SHEET 3

Access Control & ACLs on Linux

Madene Meriem Group 01

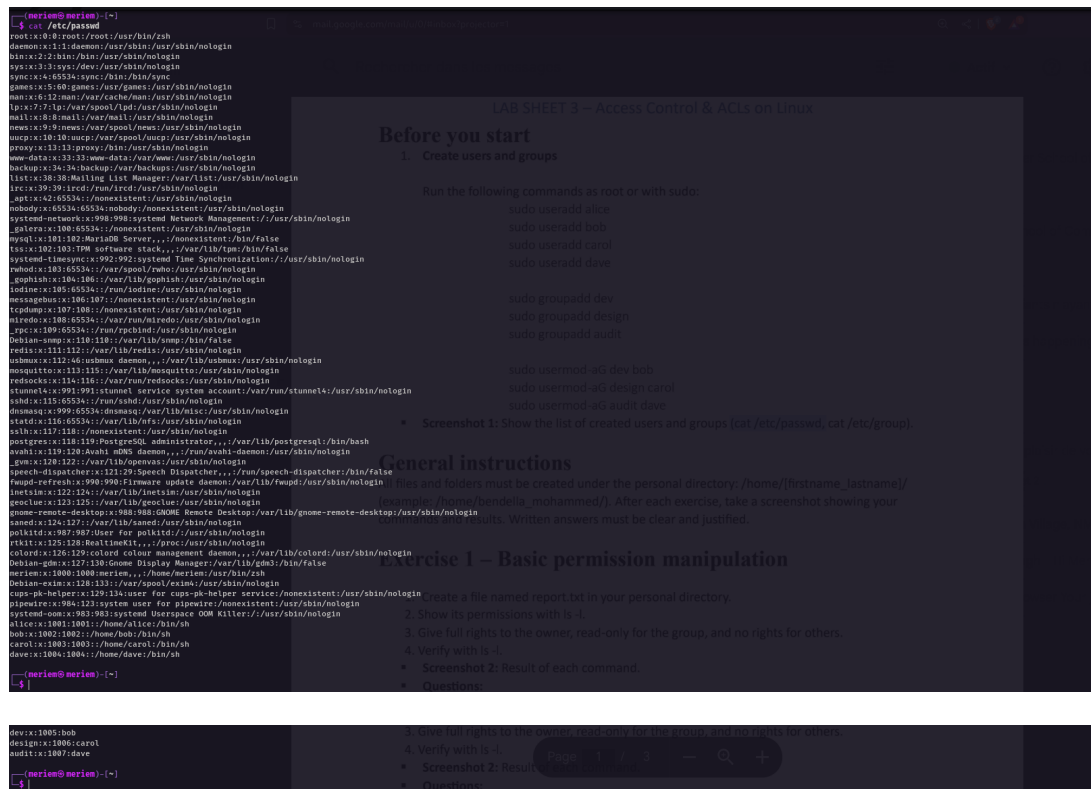
October 22, 2025

1 Setup - Users and Groups

Commands executed:

```
sudo useradd alice
sudo useradd bob
sudo useradd carol
sudo useradd dave
sudo groupadd dev
sudo groupadd design
sudo groupadd audit
sudo usermod -aG dev bob
sudo usermod -aG design carol
sudo usermod -aG audit dave
cat /etc/passwd
cat /etc/group
```

Screenshot 1:

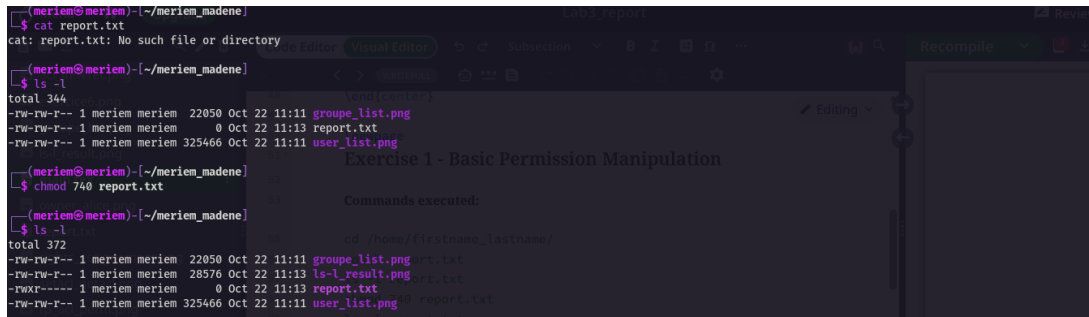


2 Exercise 1 - Basic Permission Manipulation

Commands executed:

```
cd /home/firstname_lastname/  
touch report.txt  
ls -l report.txt  
chmod 740 report.txt  
ls -l report.txt
```

Screenshot 2:



2.1 Questions

What is the numeric value corresponding to these rights?

The numeric value is 740: owner = 7 for all the rights / group = 4 for read only / others = 0 for no rights

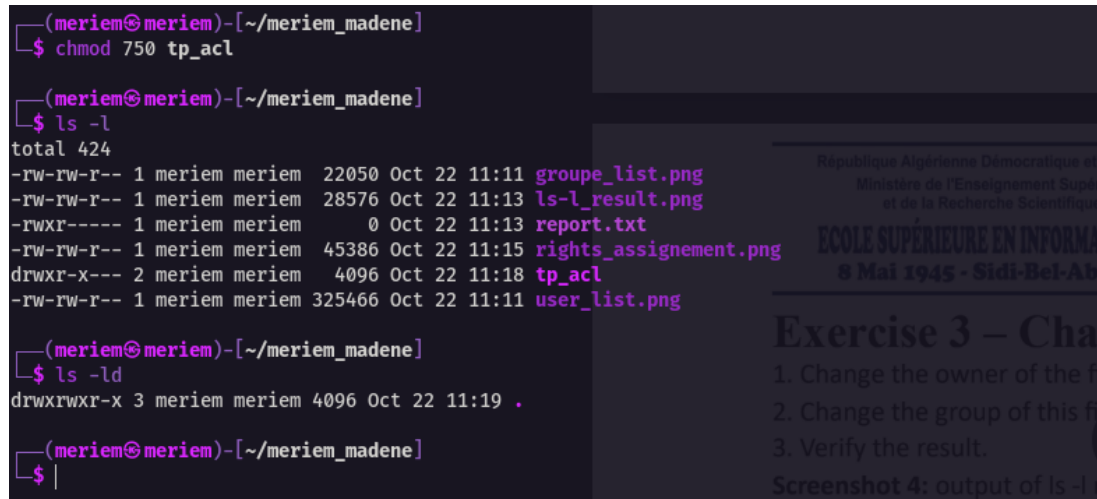
Is the principle of least privilege respected? Why? Yes the principle is respected here because we are giving to each user the only privileges they need .

3 Exercise 2 - Directory Management

Commands executed:

```
mkdir tp_acl
chmod 750 tp_acl
ls -ld tp_acl
```

Screenshot 3:



```
(meriem@meriem)-[~/meriem_madene]
$ chmod 750 tp_acl

(meriem@meriem)-[~/meriem_madene]
$ ls -l
total 424
-rw-rw-r-- 1 meriem meriem 22050 Oct 22 11:11 groupe_list.png
-rw-rw-r-- 1 meriem meriem 28576 Oct 22 11:13 ls-l_result.png
-rwxr----- 1 meriem meriem 0 Oct 22 11:13 report.txt
-rw-rw-r-- 1 meriem meriem 45386 Oct 22 11:15 rights_assignment.png
drwxr-x--- 2 meriem meriem 4096 Oct 22 11:18 tp_acl
-rw-rw-r-- 1 meriem meriem 325466 Oct 22 11:11 user_list.png

(meriem@meriem)-[~/meriem_madene]
$ ls -ld
drwxrwxr-x 3 meriem meriem 4096 Oct 22 11:19 .

(meriem@meriem)-[~/meriem_madene]
$ |
```

3.1 Question

What is the difference between the execute right on a file and on a directory?

Execute on a file allows running it as a program. Execute on a directory allows accessing its contents (entering with `cd` and listing files). Without execute on a directory, you cannot access files inside even if you have read permission.

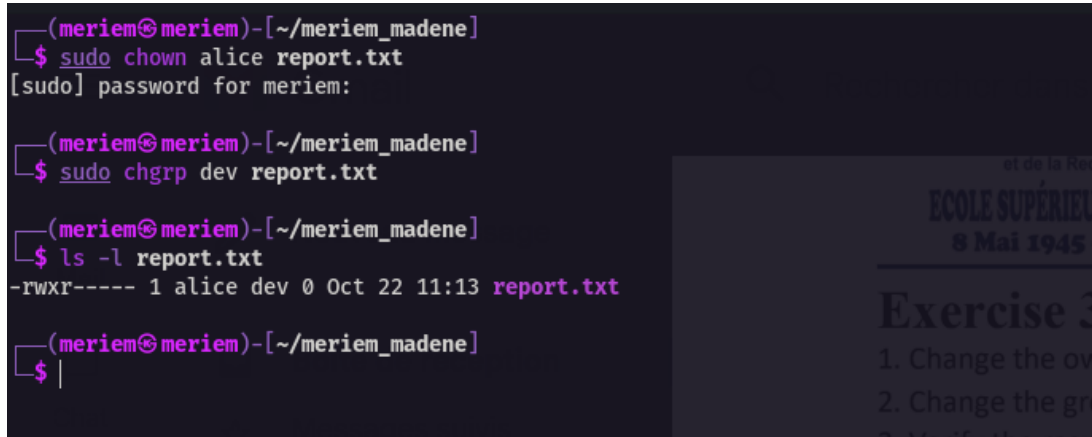
Execute on file allows executing the file itself, while the execute in the folder allows access to the content of the folder like `cd` command, because if we remove `exe` on `dir` you won't access the files inside

4 Exercise 3 - Change Owner and Group

Commands executed:

```
sudo chown alice report.txt
sudo chgrp dev report.txt
ls -l report.txt
```

Screenshot 4:



```
(meriem@meriem)-[~/meriem_madene]
$ sudo chown alice report.txt
[sudo] password for meriem:

(meriam@meriem)-[~/meriem_madene]
$ sudo chgrp dev report.txt

(meriam@meriem)-[~/meriem_madene]
$ ls -l report.txt
-rwxr----- 1 alice dev 0 Oct 22 11:13 report.txt

(meriam@meriem)-[~/meriem_madene]
$
```

4.1 Question

What is the difference between `chown` and `chgrp`?

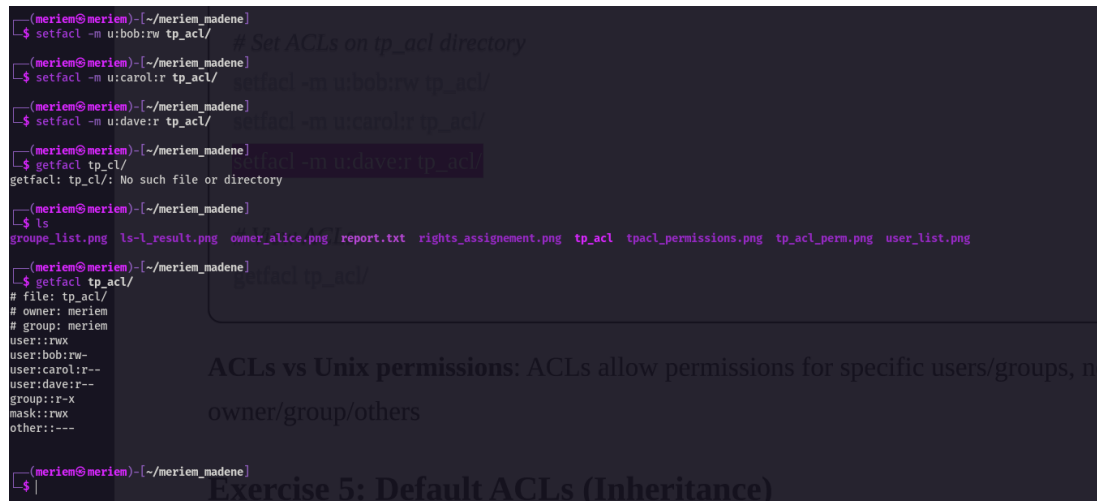
`chown` changes the owner of a file, while `chgrp` changes only the group. You can also use `chown` to change both at once with the syntax: `chown user:group file`.

5 Exercise 4 - ACL: Fine-grained Access Control

Commands executed:

```
sudo apt install acl -y
setfacl -m u:bob:rw tp_acl/
setfacl -m u:carol:r tp_acl/
setfacl -m u:dave:r tp_acl/
getfacl tp_acl/
```

Screenshot 5:



```
(meriem@meriem)~/meriem_madene
-$ setfacl -m u:bob:rw tp_acl/
(meriem@meriem)~/meriem_madene
-$ setfacl -m u:carol:r tp_acl/
(meriem@meriem)~/meriem_madene
-$ setfacl -m u:dave:r tp_acl/
(meriem@meriem)~/meriem_madene
-$ getfacl tp_acl/
getfacl: tp_acl/: No such file or directory
(meriem@meriem)~/meriem_madene
-$ ls
groupe_list.png ls-l_result.png owner_alice.png report.txt rights_assignment.png tp_acl tp_acl_permissions.png tp_acl_perm.png user_list.png
(meriem@meriem)~/meriem_madene
-$ getfacl tp_acl/
# file: tp_acl/
# owner: meriem
# group: meriem
user::rw-
user:bob:rw-
user:carol:r--
user:dave:r--
group::r-x
mask::rwx
other::---
```

5.1 Question

What is the difference between an ACL and classic Unix permissions?

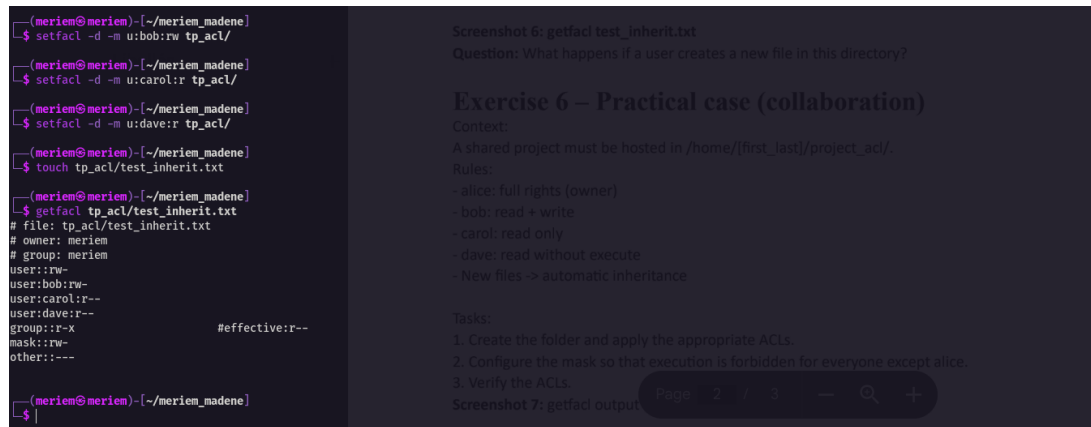
Classic Unix permissions only support three permission levels (owner, group, others). ACLs allow setting specific permissions for multiple individual users and groups, providing fine-grained access control beyond the traditional model.

6 Exercise 5 - ACL Inheritance

Commands executed:

```
setfacl -m d:u:bob:rw tp_acl/  
setfacl -m d:u:carol:r tp_acl/  
setfacl -m d:u:dave:r tp_acl/  
touch tp_acl/test_inherit.txt  
getfacl tp_acl/test_inherit.txt
```

Screenshot 6:



6.1 Question

What happens if a user creates a new file in this directory?

When a user creates a new file in this directory, it automatically inherits the default ACLs that were set on the parent directory. This ensures consistent permissions for all new files without manual configuration.

7 Exercise 6 - Practical Case

Commands executed:

```
mkdir project_acl
sudo chown alice project_acl
setfacl -m u:alice:rxw project_acl/
setfacl -m u:bob:rw project_acl/
setfacl -m u:carol:r project_acl/
setfacl -m u:dave:r project_acl/
setfacl -m d:u:alice:rxw project_acl/
setfacl -m d:u:bob:rw project_acl/
setfacl -m d:u:carol:r project_acl/
setfacl -m d:u:dave:r project_acl/
setfacl -m m::rw project_acl/
getfacl project_acl/
```

Screenshot 7:

The screenshot shows a terminal window on the left and a document on the right. The terminal window displays the following commands and output:

```
(meriem@meriem)-[~/meriem_madene]
$ sudo setfacl -m u:alice:rxw project_acl/
(meriem@meriem)-[~/meriem_madene]
$ sudo setfacl -m u:bob:rw project_acl/
(meriem@meriem)-[~/meriem_madene]
$ sudo setfacl -m u:carol:r project_acl/
(meriem@meriem)-[~/meriem_madene]
$ sudo setfacl -m u:dave:r project_acl/
(meriem@meriem)-[~/meriem_madene]
$ sudo setfacl -d -m u:alice:rxw project_acl/
(meriem@meriem)-[~/meriem_madene]
$ sudo setfacl -d -m u:bob:rw project_acl/
(meriem@meriem)-[~/meriem_madene]
$ sudo setfacl -d -m u:carol:r project_acl/
(meriem@meriem)-[~/meriem_madene]
$ sudo setfacl -d -m u:dave:r project_acl/
(meriem@meriem)-[~/meriem_madene]
$ setfacl -m m::rw- project_acl/
setfacl: project_acl/: Operation not permitted
(meriem@meriem)-[~/meriem_madene]
$ sudo setfacl -m m::rw- project_acl/
(meriem@meriem)-[~/meriem_madene]
$ sudo getfacl project_acl/
# file: project_acl/
# owner: alice
# group: alice
user::rxw
user:alice:rxw          #Effective:rw-
user:bob:rw-
user:carol:r--
user:dave:r--
group::rxw              #Effective:rw-
group:alice:rxw
group:bob:rw-
group:carol:r--
group:dave:r--
mask::rw-
mask:alice:rw-
mask:bob:rw-
mask:carol:r--
mask:dave:r--
other::r-x
other:alice:r-x
other:bob:r-x
other:carol:r-x
other:dave:r-x
default:user::rxw
default:user:alice:rxw
default:user:bob:rw-
default:user:carol:r--
default:user:dave:r--
default:group::rxw
default:group:alice:rxw
default:group:bob:rw-
default:group:carol:r--
default:group:dave:r--
default:mask::rw-
default:mask:alice:rw-
default:mask:bob:rw-
default:mask:carol:r--
default:mask:dave:r--
default:other::r-x
default:other:alice:r-x
default:other:bob:r-x
default:other:carol:r-x
default:other:dave:r-x
```

The document on the right is titled "Exercise 5 – ACL inheritance (default ACLs)" and contains the following text:

1. Add default ACLs so that all new files created inside `tp_acl/` automatically inherit the above ACL (in ex 4):

2. Create a file `test_inherit.txt` inside that folder and check its ACLs.

Screenshot 6: `getfacl test_inherit.txt`

Question: What happens if a user creates a new file in this directory?

Exercise 6 – Practical case (collaboration)

Context:
A shared project must be hosted in `/home/[first_last]/project_acl/`.

Rules:

- alice: full rights (owner)
- bob: read + write
- carol: read only
- dave: read without execute
- New files -> automatic inheritance

Tasks:

1. Create the folder and apply the appropriate ACLs.
2. Configure the mask so that execution is forbidden for everyone except alice.
3. Verify the ACLs.

Screenshot 7: `getfacl` output

8 Exercise 7 - Attack Simulation and Remediation

Commands executed:


```
ssh student@<metasploitable_ip>
cd /home/student/
touch secret.txt
chmod 777 secret.txt
ls -l secret.txt

# From another user
su - otheruser
echo "unauthorized access" >> /home/student/secret.txt
cat /home/student/secret.txt

# Fix (as student)
chmod 600 secret.txt
ls -l secret.txt

# Test fix
su - otheruser
cat /home/student/secret.txt
```

Screenshot 10:



screenshot10.png

8.1 Questions

What is the security flaw here?

The security flaw is that `chmod 777` gives full permissions (read, write, execute) to everyone on the system. Any user can read sensitive data, modify the file content, or delete it entirely. This violates confidentiality and integrity principles.

How to fix it? (Apply least privilege)

Apply `chmod 600` to restrict access to only the owner. This gives read and write permissions only to the file owner, with no access for group or others, following the principle of least privilege.