

# File Permissions

In this lab you will work with:

- The execute bits on files and directories
- The write bit on files and directories
- The sticky bit
- The set uid bit
- The stat command
- The umask command

## 1. The execute bit on files

A file is not necessarily a program, so there is no default execute bit on files. Create an empty file and list its permissions.

```
[centos@centos11 ~]$ touch file1
[centos@centos11 ~]$ ls -li file1
9017648 -rw-rw-r--. 1 centos centos 0 Apr 26 08:36 file1
[centos@centos11 ~]$
```

If you want to run it as a script you can either run it in a subshell or make it executable.

```
[centos@centos11 ~]$ echo "echo hello" > script1
[centos@centos11 ~]$ ./script1
-bash: ./script1: Permission denied
```

```
[centos@centos11 ~]$ ls -l script1
-rw-rw-r--. 1 centos centos 11 Apr 26 08:39 script1
```

(run in subshell)

```
[centos@centos11 ~]$ bash file1
hello
```

(add execute bit)

```
[centos@centos11 ~]$ chmod +x script1
[centos@centos11 ~]$ ./script1
hello
```

```
[centos@centos11 ~]$
```

## 2. The execute bit on directories.

An execute bit on a directory is needed to make the directory into your working directory.

```
[centos@centos11 ~]$ ls -l ../
total 0
drwx-----. 3 centos centos 123 Apr 26 08:39 centos
drwx-----. 2 dictu-11 dictu-11 62 Apr 26 08:24 dictu-11
```

```
[centos@centos11 ~]$ sudo touch ../dictu-11/file1
```

```
sudo chmod a+r ../dictu-11
```

If you only have read permission, you can list the filenames  
But not access the inodes to which the filenames refer.

```
[centos@centos11 ~]$ ls -l ../dictu-11
ls: cannot access ../dictu-11/file1: Permission denied
total 0
-????????? ? ? ? ? ? ? file1
```

Make the directory executable and try it again...

```
[centos@centos11 ~]$ sudo chmod a+x ../dictu-11
[centos@centos11 ~]$ ls -l ../
total 0
drwx-----. 3 centos centos 139 Apr 26 08:46 centos
drwxr-xr-x. 2 dictu-11 dictu-11 75 Apr 26 08:48 dictu-11
```

## 3. The write bit on files and directories.

Using sudo make a file in another user's directory writable. Now you can change its content.

```
[centos@centos11 ~]$ sudo chmod a+w ../dictu-11/file1
[centos@centos11 ~]$ echo centos_can_write > ../dictu-11/file1
[centos@centos11 ~]$ cat ../dictu-11/file1
centos_can_write
```

Try to remove the file. Why is that not possible?

```
[centos@centos11 ~]$ rm ../dictu-11/file1
rm: cannot remove '../dictu-11/file1': Permission denied
```

#### 4. World Writable Directories.

A world writable directory supports writing files to all users. The danger is that users can remove each other's files.

```
[centos@centos11 ~]$ touch /tmp/centosfile
```

Switch to user dictu-11

```
[centos@centos11 ~]$ su - dictu-11
Password:
Last failed login: Mon Apr 26 08:15:08 EDT 2021 on pts/0
There were 2 failed login attempts since the last successful login.
[dictu-11@centos11 ~]$ ls -ld /tmp
drwxrwxrwt. 7 root root 111 Apr 26 09:03 /tmp
```

Can you remove the file that is owned by centos?

```
[dictu-11@centos11 ~]$ rm /tmp/centosfile
rm: remove write-protected regular empty file '/tmp/centosfile'? y
rm: cannot remove '/tmp/centosfile': Operation not permitted
```

What if you remove the sticky bit from /tmp?

```
[dictu-11@centos11 ~]$ exit
Logout
[centos@centos11 ~]$ sudo chmod o-t /tmp
[sudo] password for centos:
[centos@centos11 ~]$ ls -ld /tmp
drwxrwxrwx. 7 root root 111 Apr 26 09:03 /tmp
```

Try and remove it once more...

```
[centos@centos11 ~]$ su - dictu-11
Password:
```

```
Last login: Mon Apr 26 09:09:32 EDT 2021 on pts/0
[dictu-11@centos11 ~]$ rm /tmp/centosfile
rm: remove write-protected regular empty file `/tmp/centosfile'? y
[dictu-11@centos11 ~]$
```

#### 4. SUID and SGID

The set uid bit will change the effective userid of a process to the id of the owner of the program. So if you run the su command or for example the passwd command, the forked process will run as root. Sometimes that is necessary. The SGID has the same functionality but then with regards to the group id.

There are some 12 command in /usr/bin/ that have the suid bit set.

Test the su command and see how the *eid* changes.

```
[centos@centos11 ~]$ whoami
centos
[centos@centos11 ~]$ su
Password:
[root@centos11 centos]# whoami
root
```

Remove the suid bit and the su command no longer works.

```
[root@centos11 centos]# chmod u-s /usr/bin/su
[root@centos11 centos]# exit
```

```
[centos@centos11 ~]$ su
Password:
su: Authentication failure
```

To change your password in the /etc/shadow file, you need root privileges. Remove the suid bit from the passwd command.

```
[centos@centos11 ~]$ sudo chmod u-s /bin/passwd
```

```
[centos@centos11 ~]$ passwd
Changing password for user centos.
Changing password for centos.
(current) UNIX password:
New password:
Retype new password:
passwd: Authentication token manipulation error
```

```
[centos@centos11 ~]$ sudo chmod u+s /bin/passwd
```

## 5. The stat command

```
[centos@centos11 ~]$ touch file2
```

```
[centos@centos11 ~]$ stat file2
  File: 'file2'
  Size: 0          Blocks: 0          IO Block: 4096   regular empty
file
Device: fd00h/64768d Inode: 9017648    Links: 1
Access: (0664/-rw-rw-r--)Uid: ( 1000/ centos)  Gid: ( 1000/ centos)
Context: unconfined_u:object_r:user_home_t:s0
Access: 2021-04-26 15:33:35.419947571 +0200
Modify: 2021-04-26 15:33:35.419947571 +0200
Change: 2021-04-26 15:33:35.419947571 +0200
  Birth: -
```

```
[centos@centos11 ~]$ chmod 777 file2
```

```
[centos@centos11 ~]$ stat file2
  File: 'file2'
  Size: 0          Blocks: 0          IO Block: 4096   regular empty file
Device: fd00h/64768d Inode: 9017648    Links: 1
Access: (0777/-rwxrwxrwx)  Uid: ( 1000/ centos)  Gid: ( 1000/
centos)
Context: unconfined_u:object_r:user_home_t:s0
Access: 2021-04-26 15:33:35.419947571 +0200
Modify: 2021-04-26 15:33:35.419947571 +0200
Change: 2021-04-26 15:33:47.342110848 +0200
  Birth: -
```

What happens to the Access time when you **cat** the file?

## 6. The umask command

The umask command set the default permissions for files and directories. What are the default permissions for files?

```
[centos@centos11 ~]$ touch file3
[centos@centos11 ~]$ ls -l file3
-rw-rw-r--. 1 centos centos 0 Apr 26 15:43 file3
```

What happens if you set the umask to all zeroes?

```
[centos@centos11 ~]$ umask 000
[centos@centos11 ~]$ touch file4
[centos@centos11 ~]$ ls -l file4
```

And what about directories?

First change it back to the user's defaults.

```
[centos@centos11 ~]$ umask 0022
[centos@centos11 ~]$ mkdir dir1
[centos@centos11 ~]$ ls -ld dir1
drwxr-xr-x. 2 centos centos 6 Apr 26 15:44 dir1
```

And then to all zeroes again.

```
[centos@centos11 ~]$ umask 0000
[centos@centos11 ~]$ mkdir dir2
[centos@centos11 ~]$ ls -ld dir2
drwxrwxrwx. 2 centos centos 6 Apr 26 15:45 dir2
```