

## 27.4. practice: standard file permissions

1. As normal user, create a directory ~/permissions. Create a file owned by yourself in there.
2. Copy a file owned by root from /etc/ to your permissions dir, who owns this file now ?
3. As root, create a file in the users ~/permissions directory.
4. As normal user, look at who owns this file created by root.
5. Change the ownership of all files in ~/permissions to yourself.
6. Make sure you have all rights to these files, and others can only read.
7. With chmod, is 770 the same as rwxrwx--- ?
8. With chmod, is 664 the same as r-xr-xr-- ?
9. With chmod, is 400 the same as r----- ?
10. With chmod, is 734 the same as rwxr-xr-- ?
- 11a. Display the umask in octal and in symbolic form.
- 11b. Set the umask to 077, but use the symbolic format to set it. Verify that this works.
12. Create a file as root, give only read to others. Can a normal user read this file ? Test writing to this file with vi.
- 13a. Create a file as normal user, give only read to others. Can another normal user read this file ? Test writing to this file with vi.
- 13b. Can root read this file ? Can root write to this file with vi ?
14. Create a directory that belongs to a group, where every member of that group can read and write to files, and create files. Make sure that people can only delete their own files.

## 27.5. solution: standard file permissions

1. As normal user, create a directory ~/permissions. Create a file owned by yourself in there.

```
mkdir ~/permissions ; touch ~/permissions/myfile.txt
```

2. Copy a file owned by root from /etc/ to your permissions dir, who owns this file now ?

```
cp /etc/hosts ~/permissions/
```

3. The copy is owned by you. True/False  
True

4. As root, create a file in the users ~/permissions directory.  
NOT INCLUDED need guidance on exact syntax

```
(become root)# touch /home/username/permissions/rootfile
```

5. As normal user, look at who owns this file created by root.  
NOT INCLUDED since this is dependent on prior question, need guidance on syntax

```
ls -l ~/permissions
```

The file created by root is owned by root.

6. Change the ownership of all files in ~/permissions to yourself.  
NOTE: Asterisk below allows any characters

```
chown user ~/permissions/*
```

You cannot become owner of the file that belongs to root.  
NOTE: Does this mean that the command above will fail?  
NOT INCLUDED

7. Make sure you have all rights to these files, and others can only read.  
NOT INCLUDED  
How should this be worded given there are 2 commands, one for files and one for directories

```
chmod 644 (on files)
```

```
chmod 755 (on directories)
```

8. With chmod, is 770 the same as

rw-rwx--- ? yes

9. With chmod, is 664 the same as

r-xr-xr-- ? No

10. With chmod, is 400 the same as

r----- ? yes

11. With chmod, is 734 the same as

rw-xr-xr-- ? no

11a. Display the umask in octal and in symbolic form.

```
umask ; umask -S
```

11b. Set the umask to 077, but use the symbolic format to set it.

Verify that this works. THIS PART NOT INCLUDED

```
umask -S u=rwx,g=
```

Please confirm that there are only 2 blanks above, one after 'umask' and one after '-S'

12. Create a file as root, give only read to others. Can a normal user read this file ? Test writing to this file with vi.

NOT INCLUDED

Multiple lines

```
(become root)
```

```
# echo hello > /home/username/root.txt
```

```
# chmod 744 /home/username/root.txt
```

```
(become user)
```

```
vi ~/root.txt
```

13a. Create a file as normal user, give only read to others. Can another normal user read this file ? Test writing to this file with vi.

NOT INCLUDED need guidance on exact syntax

```
echo hello > file ; chmod 744 file
```

Yes, others can read this file

13b. Can root read this file ? Can root write to this file with vi ?

Yes, root can read and write to this file. Permissions do not apply to root.

NOT INCLUDED until prior question is added

14. Create a directory that belongs to a group, where every member of that group can read and write to files, and create files. Make sure that people can only delete their own files.

NOT INCLUDED need guidance on correct wording e.g. Create a directory that belongs to a group called 'project 42' etc.

Also, multiple commands - break up into several connected questions?

```
mkdir /home/project42 ; groupadd project42
```

```
chgrp project42 /home/project42 ; chmod 775 /home/project42
```

You can not yet do the last part of this exercise...