POSIX File Permissions Primer

Advanced Web Development ….Instructor: Leigh Cotnoir

This is a little primer on POSIX file permissions. POSIX stands for “**P**ortable **O**perating **S**ystem **I**nterface for Uni**X**.”

POSIX permissions (perms) are set in “octal notation”. Octal notation consists of a three- or four-digit “base-8” value, where “base-8” is a number system (0-7). For a general point of reference on what that means, we are accustomed to counting in a “base-10” system (0-9), and computers are accustomed to counting in the most elemental number system, “base-2” or binary (0-1).

In base-8, we use a **3-digital octal notation** to represent a different component of the permission set:

1. User class (also referred to a “owner” class),
2. Group class,
3. “Others” class (also referred to as “everyone” or “world”).

Each of these digits is the sum of its component bits (see also Binary numeral system). As a result, specific bits add to the sum as it is represented by a numeral:

* The **read** bit adds 4 to its total (in binary 100),
* The **write** bit adds 2 to its total (in binary 010), and
* The **execute** bit adds 1 to its total (in binary 001).

OKAY, WHAT DOES THAT MEAN!?

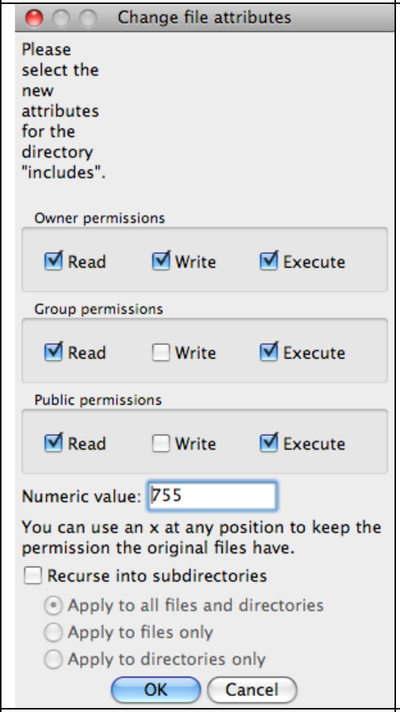
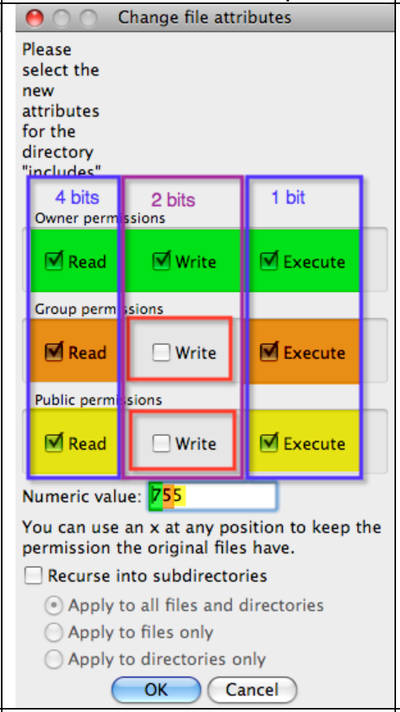
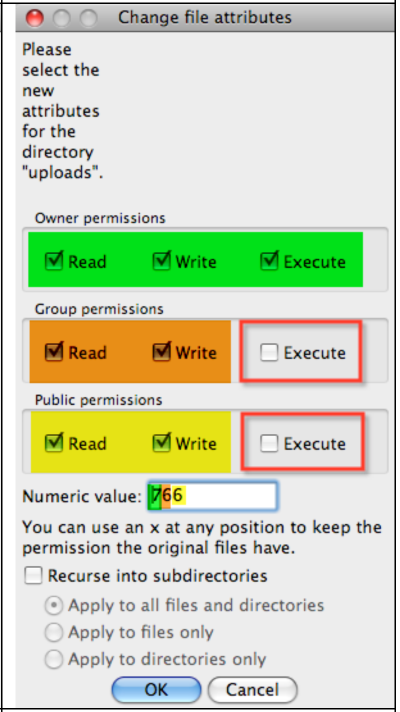
Let’s use a visual:

Value	Symbolic Representation			Bit Equivalent	Permission
0	-	-	-	0 + 0 + 0 = **0 bits**	No permission
1	-	-	x	0 + 0 + 1 = **1 bit**	Execute
2	-	w	-	0 + 2 + 0 = **2 bits**	Write
3	-	w	x	0 + 2 + 1 = **3 bits**	Write + execute
4	r	-	-	4 + 0 + 0 = **4 bits**	Read
5	r	-	x	4 + 0 + 1 = **5 bits**	Read + execute
6	r	w	-	4 + 2 + 0 = **6 bits**	Write + read
7	r	w	x	4 + 2 + 1 = **7 bits**	Read+ write + execute

If you look at the first two columns in the table above, you can compare the “symbolic representation” presence/absence of r,w,x to the binary numeric representations of each value below.

Value	Binary Numeric Representation			Bit Equivalent	Permission
0	0	0	0	0 + 0 + 0 = **0 bits**	No permission
1	0	0	1	0 + 0 + 1 = **1 bit**	Execute
2	0	1	0	0 + 2 + 0 = **2 bits**	Write
3	0	1	1	0 + 2 + 1 = **3 bits**	Write + execute
4	1	0	0	4 + 0 + 0 = **4 bits**	Read
5	1	0	1	4 + 0 + 1 = **5 bits**	Read + execute
6	1	1	0	4 + 2 + 0 = **6 bits**	Write + read
7	1	1	1	4 + 2 + 1 = **7 bits**	Read+ write + execute

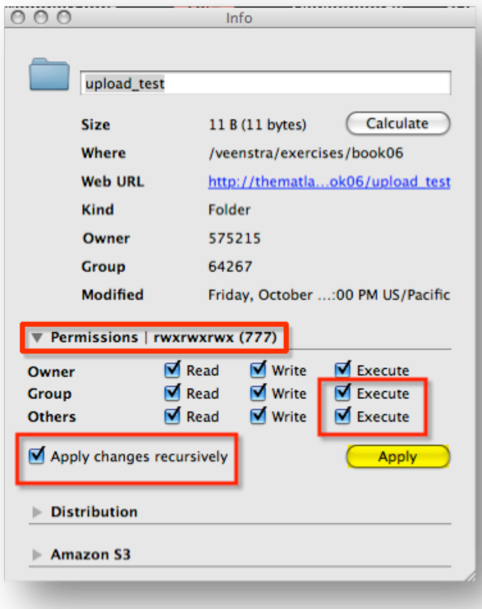
Each class (user, group, other) needs a specific permission designation so that the operating system will know how to treat the target file or directory. Let's look at some examples of permissions on directories below:

Filezilla view: 755	Filezilla view: 755 explained	Filezilla view: 766
 Change file attributes Please select the new attributes for the directory "includes". Owner permissions: Read ✓ Write ✓ Execute ✓ Group permissions: Read ✓ Write ☐ Execute ✓ Public permissions: Read ✓ Write ☐ Execute ✓ Numeric value: 755 You can use an x at any position to keep the permission the original files have. ☐ Recurse into subdirectories ● Apply to all files and directories ○ Apply to files only ○ Apply to directories only OK Cancel	 Change file attributes Please select the new attributes for the directory "includes". Owner permissions: Read ✓ Write ✓ Execute ✓ Group permissions: Read ✓ Write ☐ Execute ✓ Public permissions: Read ✓ Write ☐ Execute ✓ Numeric value: 755 You can use an x at any position to keep the permission the original files have. ☐ Recurse into subdirectories ● Apply to all files and directories ○ Apply to files only ○ Apply to directories only OK Cancel	 Change file attributes Please select the new attributes for the directory "uploads". Owner permissions: Read ✓ Write ✓ Execute ✓ Group permissions: Read ✓ Write ✓ Execute ☐ Public permissions: Read ✓ Write ✓ Execute ☐ Numeric value: 766 You can use an x at any position to keep the permission the original files have. ☐ Recurse into subdirectories ● Apply to all files and directories ○ Apply to files only ○ Apply to directories only OK Cancel
Standard permissions for web directories	Notice how the bits add up to create each number in the total octal permissions value of 755.	If we change permissions to 766, we remove the ability allow for directory contents modification while disabling the execute bit. In web directories, this means that Apache can't display image files from this directory in a browser. But it also restricts execution of other potentially harmful scripts.

Changing an uploads folder's permissions to 766 is useful if you need enable the ability for users to simply upload materials to a server without having to display the content. It is a safer method than 777 because it prevents malware from propagating up the directory tree via script execution in the uploads folder.

If you want people to be able to upload image files, however, that are to be viewed in the site's web pages, you need to change the folder permissions to 777. If you open the permissions up to 777, it is important to write restrictions into your upload script to try and keep as many potentially harmful files out as possible. If you are only using the folder for web images, you should restrict the MIME types to web-safe files (gifs, jpgs, pngs).
Some other application files permissions views:

Cyberduck view : change from 766 to 777



Info

upload_test

Size	11 B (11 bytes)
Where	/veenstra/exercises/book06
Web URL	http://thematla...ok06/upload_test
Kind	Folder
Owner	575215
Group	64267
Modified	Friday, October ...:00 PM US/Pacific

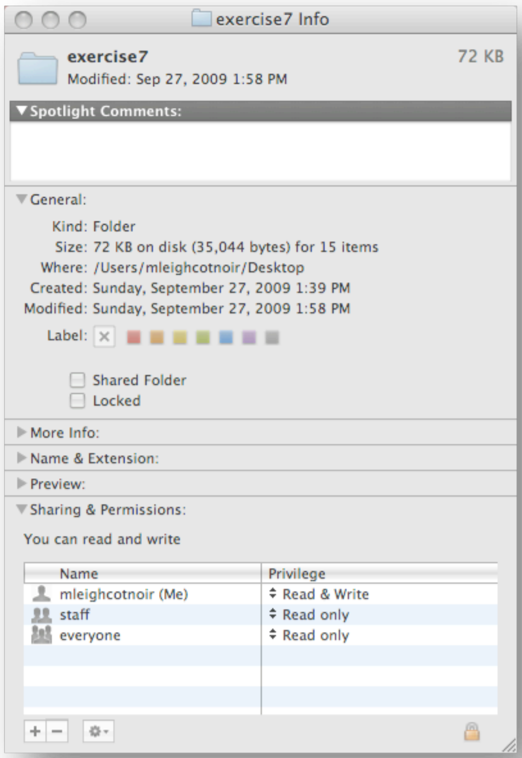
Permissions | rwxrwxrwx (777)

Owner	✓ Read	✓ Write	✓ Execute
Group	✓ Read	✓ Write	✓ Execute
Others	✓ Read	✓ Write	✓ Execute

✓ Apply changes recursively

Applying changes "recursively" means that you make all subdirectories take the same permissions.

On a Mac, OS X 10.5 using "Get Info" from the directory properties:



exercise7 Info

exercise7 72 KB
Modified: Sep 27, 2009 1:58 PM

Spotlight Comments:

General:

Kind: Folder
Size: 72 KB on disk (35,044 bytes) for 15 items
Where: /Users/mleighcotnoir/Desktop
Created: Sunday, September 27, 2009 1:39 PM
Modified: Sunday, September 27, 2009 1:58 PM
Label:

☐ Shared Folder
☐ Locked

Sharing & Permissions:

You can read and write

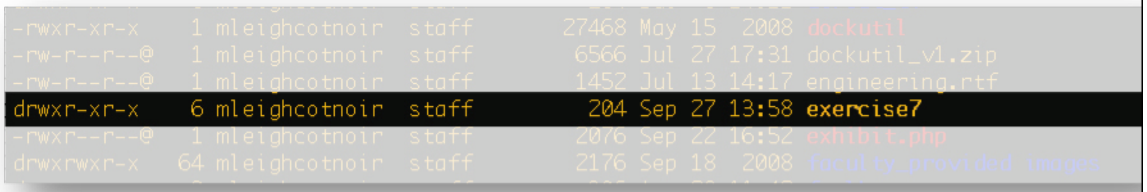
Name	Privilege
mleighcotnoir (Me)	Read & Write
staff	Read only
everyone	Read only

In the "Sharing and Permissions" above, we see that the

- User is "mleighcotnoir"

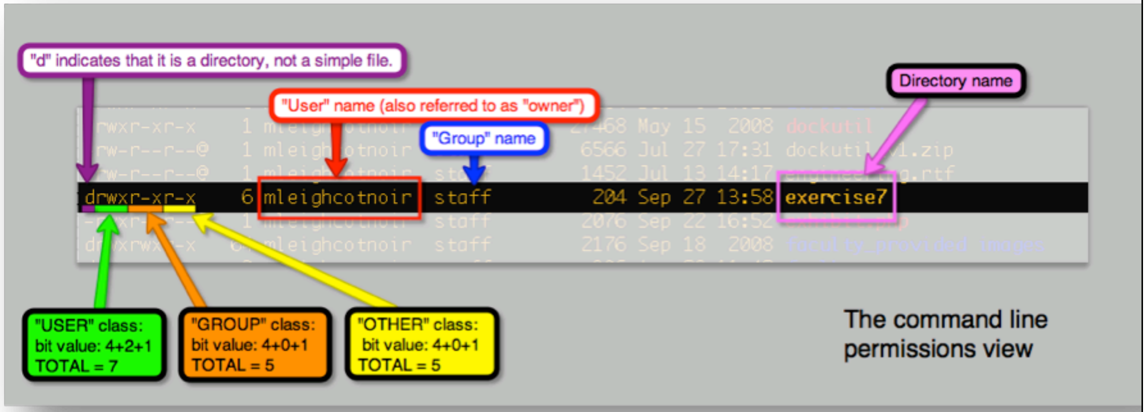
- Group is "staff"
- Others (is always "everyone")

Note, however, that on a Mac, it does not tell us anything about the execute bit using "Get Info." It only shows us simple permission properties. You have to use a special program or go into the command line to get more detail on the permissions if changing the execute bit.



```
-rwxr-xr-x     1 mleighcotnoir  staff      27468 May 15  2008 dockutil
-rw-r--r--@    1 mleighcotnoir  staff       6566 Jul 27 17:31 dockutil_v1.zip
-rw-r--r--@    1 mleighcotnoir  staff       1452 Jul 13 14:17 engineering.rtf
drwxr-xr-x     6 mleighcotnoir  staff        204 Sep 27 13:58 exercise7
-rwxr--r--@    1 mleighcotnoir  staff       2076 Sep 22 16:52 exhibit.php
drwxrwxr-x    64 mleighcotnoir  staff       2176 Sep 18  2008 faculty_provided images
```

Above the command line properties of the same directory called "exercise7" as in the Mac example above. Below are the details of how to read these details.



```
drwxr-xr-x     6 mleighcotnoir  staff        204 Sep 27 13:58 exercise7
```

"d" indicates that it is a directory, not a simple file.
"User" name (also referred to as "owner")
"Group" name
Directory name
"USER" class: bit value: 4+2+1 TOTAL = 7
"GROUP" class: bit value: 4+0+1 TOTAL = 5
"OTHER" class: bit value: 4+0+1 TOTAL = 5
The command line permissions view

You will notice in the above example that the execute bit is enabled, even though the Mac "Get Info" panel doesn't give us this information. To get this information in the command line, you must use the Terminal application in the /Applications/Utilities/ folder. To get the information on a folder, go to its parent directory and issue a list command with a "switch" to show the "long" view. If we assume that you wanted to get the info on the main web-serving folder on a Mac computer, located at "/Library/WebServer/Documents/" then you would type the following into the command line:

```
cd /Library/WebServer/
ls –l
```

This would return the permissions for all directories in the WebServer folder, including the "Documents" folder, which is where the main default web-serving folder is for Macs.

SPECIAL NOTE: If you review the fopen() function, please note that the r,w,x letters behave slightly differently as parameter commands for the fopen() function!