LexiConn Knowledgebase

Unix / Linux Permissions

$(function(){ $(".Box").css("display","none"); }); Unix- and Linux-based operating systems are designed to serve as multi-user environments and apply both Ownership and Permissions to every file and directory in the filesystem to ensure that file sharing can be accomplished effectively and that private files are kept private .

Ownership

Every file and directory is assigned to an owner and to a group. In this example, you will notice the user and group ownership designation next to a typical file (assuming your domain is abc.com and your username is abcd) :

-rwxr-xr-- 1 abc abcgrp 2690 May 13 10:43 index.html

Permissions

Now that an owner and a group is assigned to each file and directory, File Permissions can be assigned to define what members of each group can do with a file or directory. Permissions are defined as Read, Write, and Execute for three different entities:

1. The Owner
2. The Group
3. Other (any who is not the owner and is not a member of the assigned group)

Read, Write, and Execute permissions are indicated by rwx respectively. Thus, the string of letters at the beginning of the previous example - rwxr-xr-- indicate that the owner, abc can read the file, write to the file, and execute the file (the first three letters - rwx), the Group, abcgrp can read the file and execute it, but they cannot write to the file (the next three letters - r-x), and Other, (also the identifier of any normal user accessing your site from the web) can read the file, but cannot write to or execute it (the final three letters - r--).

-rwxr-xr-- 1 abc abcgrp 2690 May 13 10:43 index.html

When applied to a directory, permissions control access to all of the files within the directory. A letter 'd' precedes the permissions definition to indicate a directory, as in the example below. In this case, not even the Owner has write permissions; so, although the owner can delete any files he has write permissions for from within the directory, he/she may not delete the directory.

dr-xr-xr-- 16 abc abcgrp 4096 May 29 12:24 www

There is a numerical shorthand used frequently when referring to permissions that is based on binary code.

In binary code, the first column represents the numeral 1, the second column represents the numeral 2, and the third column represents the numeral 4. Thus:

Binary: 111 = 7 --> (1 + 2 + 4)
Binary: 001 = 4 --> (0 + 0 + 4)
And Binary: 101 = 5 --> (1 + 0 + 4)

In the same way, binary numbers can be used to represent permissions (0 = false, 1 = true):

rwx: 111 = 7 --> (1 + 2 + 4)
r-x: 101 = 5 --> (1 + 0 + 4)
-wx: 011 = 6 --> (0 + 2 + 4)

Thus, a file with permissions of rwxr-xr-- would have a numerical equivalent of 751:

Owner | Group | Other rwx | r-x | r-- 111 | 101 | 100 1+2+4 | 1+0+4 | 1+0+0 ---------- 7 | 5 | 1

Here is a brief conversion chart:

1 --x execute 2 -w- write 3 -wx write and execute 4 r-- read 5 r-x read and execute 6 rw- read and write 7 rwx read, write and execute Most web-accessible directories are set to 755 and most html files are set to 711 (or 755) by default.

https://support.lexiconn.com/kb/questions/202/