LINUX FOUNDATION

TRAINING

File Attributes

A Linux Foundation Training Publication
www.training.linuxfoundation.org

Overview

The Linux operating systems allows for a high degree of control over access to files and directories. Each file and directory has attributes that determine access at the owner, group, and world level. These attributes are used to determine whether a user or process can read, write, and execute a file. In this context, the "world" consists of all users who are neither the owner of the file or belong to the group owner of the file.

File access, or permission, is granted hierarchically. Group access overrides universal access, and owner access overrides group access. The permissions on a given file or directory are often represented numerically as an octal.

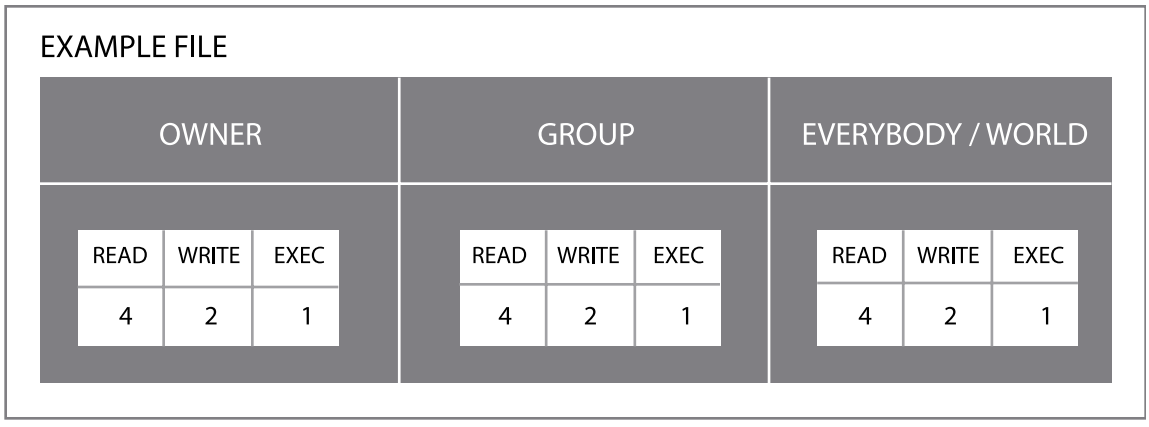
Key Ideas

Permission levels: Each file and directory has permissions set at three levels: owner, group, and world. The permissions at each level can be represented together as an octal (three digit number with one number each for owner, group, and world permissions).

Permission types: Permission is given to perform actions on files and directories: read, write, and execute. To calculate the octal representation of what permissions are allowed, each action is worth a certain number or points. Read permission = 4, Write = 2, Execute = 1. For example, if permissions allow for read, write, and execute, that is represented as a 7.

Permissions are simbolized as follows:

r --> read permissions
w --> write permissions
x --> execute permissions



ls -l: The first 10 characters in each line in the output of this command indicate the file type (1st character), owner permissions (2nd-4th characters), group owner permissions (5th-7th), and world permissions (8th-10th).

Let's look at an example:

drwxr-xr-x. 3 root root 18 Feb 11 22:49 mnt

1st character: Type of file (d = directory)

2nd - 4th characters: Owner permissions (read, write, execute = 7)

5th - 7th characters: Group permissions (read and execute, but not write = 5)

8th - 10th characters: World permissions (read and execute, but not write = 5)

Owner: root
Group: root

chmod: The chmod command is used to change the permissions of a file or directory.

chown: The chown command is used to change the permissions of a file or directory.

Example Scenario

The following series of examples shows different scenarios where you can discover and change file attributes.

Now Do It

1. Find the file attributes of the /home directory.
2. Find the octal representation of the permissions of /etc/hosts
3. If a file's permissions are represented by 755, what actions can members of the group owner take?
4. Create a file in /root called star.txt, and change its ownership to a different user.
5. Change the permissions of of star.txt so that the owner, group owner, and world can read and execute.

If you remember nothing else...

Chmod 777 means that everyone can read, write, and execute the file.

Answer Key

1. Find the file attributes of the home directory
   # ls -ld /home
   drwxr-xr-x. 3 root root 21 Feb 8 13:09 home

2. Octal representation of /etc/hosts: 644

3. 755 means members of the owner's group can read and execute

4. Create a file and change it's ownership:
   # touch /root/star.txt
   # chown USER /root/start.txt
     (where USER is replaced with an existing user)
   No output on successful execution of command

5. Change permissions so everyone can read, write execute:
   # chmod 777 /root/star.txt
   No output on successful execution of command.

LINUX FOUNDATION

Get Certified!

Get more information at http://training.linuxfoundation.org/certification/lfcs