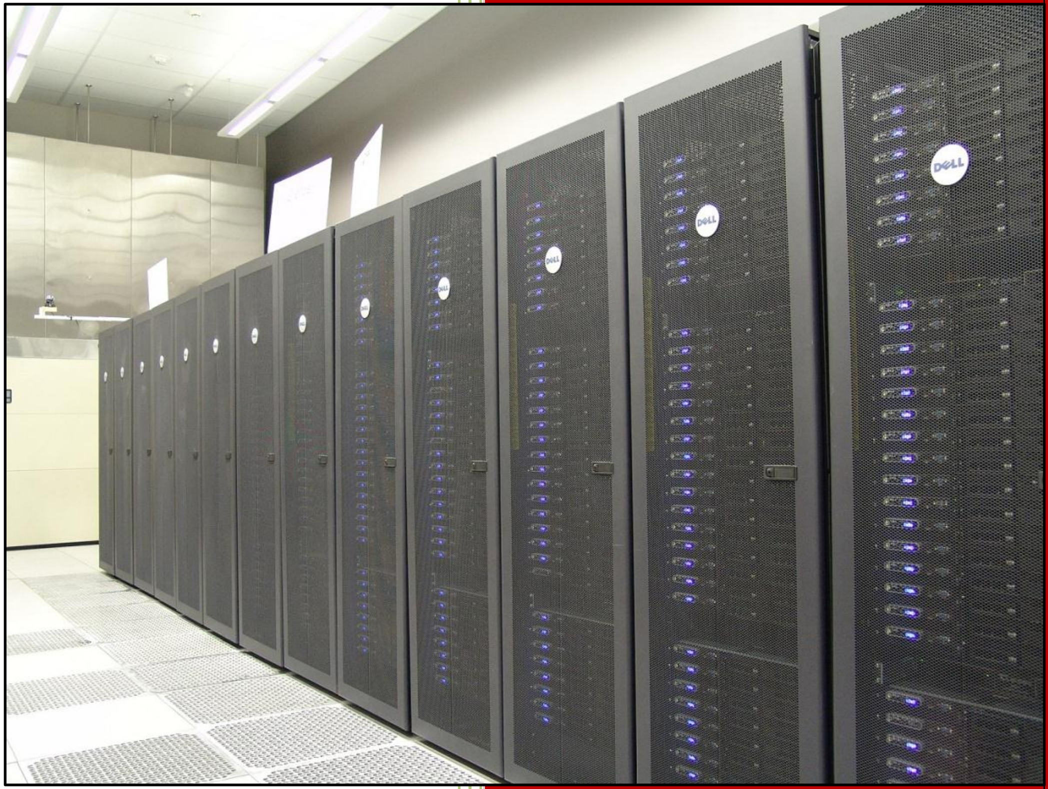
HPCC - Hrothgar

Getting Started User Guide – Setting File Access Permissions



High Performance Computing Center
Texas Tech University

Table of Contents

1. Understanding File Ownership and Permissions

In Linux system, each file and directory is owned by a specific user and group. There are three user based permission groups.

User/Owner: The person who owns the file or directory. By default, the user who creates the file or directory will become its owner.

Group: The group that owns the file. By default, when the user creates a file or directory, the group which he/she belongs to will become its group owner. All users in the group have the same access permissions to the file or directory.

Others: All other users on the system, who are not the owner and don't belong to the same group of the owner.

There are three types of access permissions on Linux: read, write, and execute. These permissions are defined separately for the file's owner, group and all other users.

Read: The read permission means that the file or directory can be opened and the contents can be read.

Write: The write permission means that the file or directory can be written or modified. The user is allowed to modify the contents of the file if the user has the write permission. However, the user is not allowed to rename or delete the file unless the user has the write permission to the directory which the file belongs to.

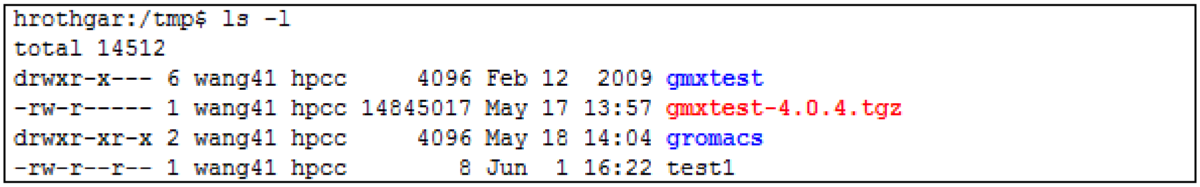
Execute: The execute permission means that the user can execute the file as a program or view the contents of the directory.

2. Viewing File Permissions

On Hrothgar, by default, users have read, write and execute permissions to the directories and files that they own. Also, the directories and files are readable and executable to other users, including the users in the same group of the owner. Basically a user is the owner of the directories /home/user-id, /lustre/work/user-id, and /lustre/scratch/user-id, as well as all files and directories under them. A user also owns the temporary files or directories in /state/partition1 on compute nodes, if their jobs create temporary output there.

Users can view the permissions for their files or directories by typing the command

ls -l



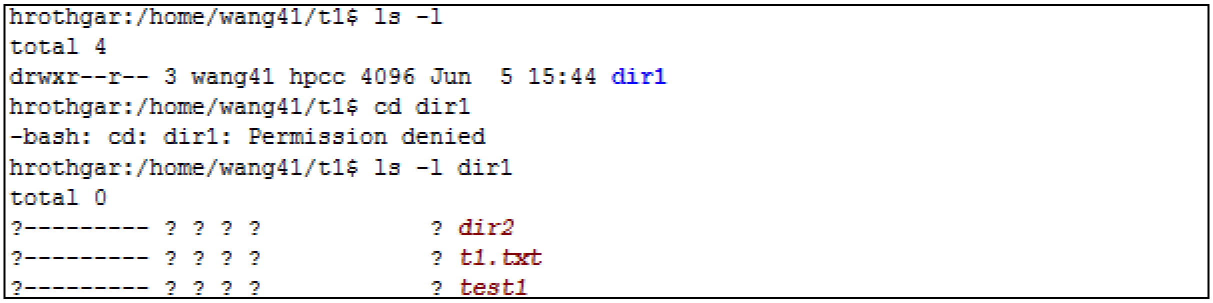
```
hrothgar:/tmp$ ls -l
total 14512
drwxr-x--- 6 wang41 hpcc     4096 Feb 12  2009 gmxtest
-rw-r----- 1 wang41 hpcc 14845017 May 17 13:57 gmxtest-4.0.4.tgz
drwxr-xr-x 2 wang41 hpcc     4096 May 18 14:04 gromacs
-rw-r--r-- 1 wang41 hpcc        8 Jun  1 16:22 test1
```

In this output example, the first character shows the file's type, d means a directory and – means a regular file. The following set of three characters is for the owner permissions. The second set of three characters is for the group permissions. And the third set of three characters is for all users' permission. The followed number shows the number of links (for directory entries that refer to the files). The following shows the owner of the file and the group the file belongs to. The rest show the file's size in bytes, date and time of last modification, and the filename.

There are two special cases for the directory access permissions.

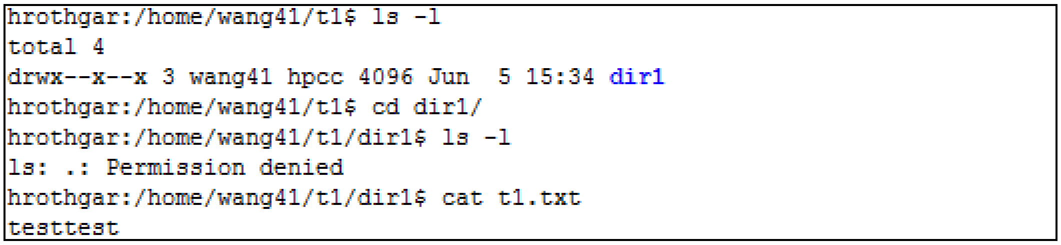
1. For a directory with only the read permission, the users can only see the names of the inside files by using the *ls* command, but not *cd* into the directory.

   In this example, the users refer to all users except the owner of the directory.



```
hrothgar:/home/wang41/t1$ ls -l
total 4
drwxr--r-- 3 wang41 hpcc 4096 Jun  5 15:44 dir1
hrothgar:/home/wang41/t1$ cd dir1
-bash: cd: dir1: Permission denied
hrothgar:/home/wang41/t1$ ls -l dir1
total 0
?--------- ? ? ? ?            ? dir2
?--------- ? ? ? ?            ? t1.txt
?--------- ? ? ? ?            ? test1
```

2. For a directory with only the execute permission, the users can *cd* into the directory, but cannot list its contents by using the *ls* command. However, if the users know a file's name in the directory, the users can view the file's content.



```
hrothgar:/home/wang41/t1$ ls -l
total 4
drwx--x--x 3 wang41 hpcc 4096 Jun  5 15:34 dir1
hrothgar:/home/wang41/t1$ cd dir1/
hrothgar:/home/wang41/t1/dir1$ ls -l
ls: .: Permission denied
hrothgar:/home/wang41/t1/dir1$ cat t1.txt
testtest
```

3. Setting File Permissions

Each of the types of access permissions has an associated numeric value.

- read = 4
- write = 2
- execute = 1

The values for the access rights for each of the groups is added together to obtain a value between 0 and 7 which can be used to assign or modify permissions using the *chmod* (change mode) command.

- rwe = 4 (read) + 2 (write) + 1 (execute) = 7
- rw = 4 (read) + 2 (write) = 6
- re = 4 (read) + 1 (execute) = 5

If you are concerned about the permission settings, you can change the permission by using command "*chmod*" with appropriate options.

For example,

```
chmod 755 filename
```

Change the file's permissions to `-rwxr-xr-x`. The owner would have full permissions (read, write, and execute), the group and other users would have read and execute permissions.

```
chmod -R 755 directory_name
```

Change the directory and all its files permissions to `drwxr-xr-x`. To the directory and the files it contains, the owner would have full permissions (read, write, and execute), the group and other users would have read and execute permissions. (Note: even the group and others have read and execute permissions to the directory, other kinds of limited access permissions can be set for its specific files.)

```
chmod 750 filename
```

Change the file’s permissions to -rwxr-x---.   The owner would have full permissions (read, write, and execute), the group would have read and execute permissions, and all other users wouldn’t have any permissions.

```
chmod 700 filename
```

Change the file’s permissions to -rwx------. Only the owner would have full permissions. The group and all others wouldn’t have any permission.

```
chmod 754 filename
```

Change the file’s permissions to -rwxr-xr--. The owner would have full permissions (read, write, and execute), the group would have read and execute permissions, and all others would only have the read permission.

```
chmod 744 filename
```

Change the file’s permissions to -rwxr--r--.   The owner would have full permissions (read, write, and execute), the group and all others would have the read permission.

Last updated: 06/04/2012

For Additional Assistance Contact: hpccsupport@ttu.edu

For Comments/Suggestions on user guide hpcc@ttu.edu