View the umask for your current shell by executing the following command:

umask  
0022

This just means that of the four possible positions to mask out (special permissions, user owner permissions, group owner permissions, and other permissions), the last two have the write permission masked out or not used. Note: The first position never has any effect as the special permissions are never set by default. As a result, the umask value of 0022 is really the same as 022.

If you create a file when the umask is set to 022, the file’s permissions is as follows:

-rw-r--r--    1 root   root       881 Feb 17 09:11 file1

If you create a directory with the same umask set, the directory’s permissions are as follows:

drwxr-xr-x    2 root   root       4096 Feb 17 14:47 dir1

To understand why these permission sets are different, think about the process of how the umask is applied. To begin with, recall the default permissions:

                     For Files              For Directories  
Maximum              rw-rw-rw-              rwxrwxrwx

Now, consider a umask value of 022. This means you are “masking out” or removing the write permissions for the group owner and others (represented by the M values here):

                     For Files              For Directories  
Maximum              rw-rw-rw-              rwxrwxrwx  
MASK              ----M--M-              ----M--M-

When those permissions are masked out, you end up getting the following permissions on new files and directories:

                     For Files              For Directories  
Maximum              rw-rw-rw-              rwxrwxrwx  
MASK              ----M--M-              ----M--M-  
Result              rw-r--r--              rw-r--r--

As you can see, the umask value changes the default permissions of a created object based on the following formula:

Maximum default value – umask value = create value

You can change the umask value to a more restrictive one with the **umask 027** command.