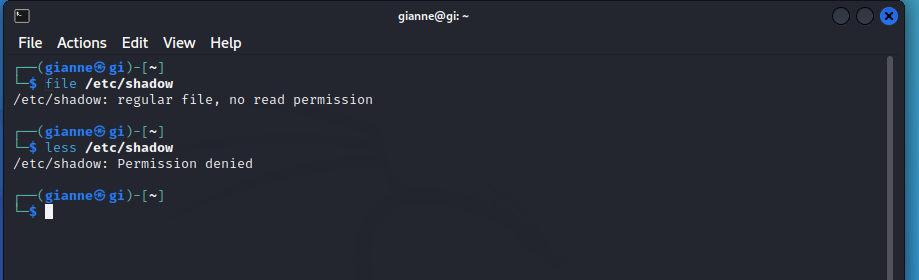
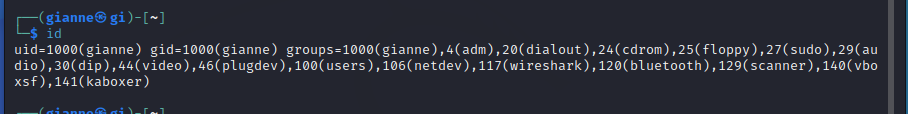
**Activity 9 - Permissions**

**Owners, Group, Members and Everybody Else**

****

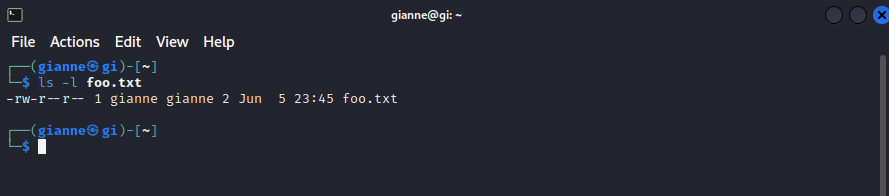
The reason for this error message is that, as regular users, we do not have permission to read this file.

****

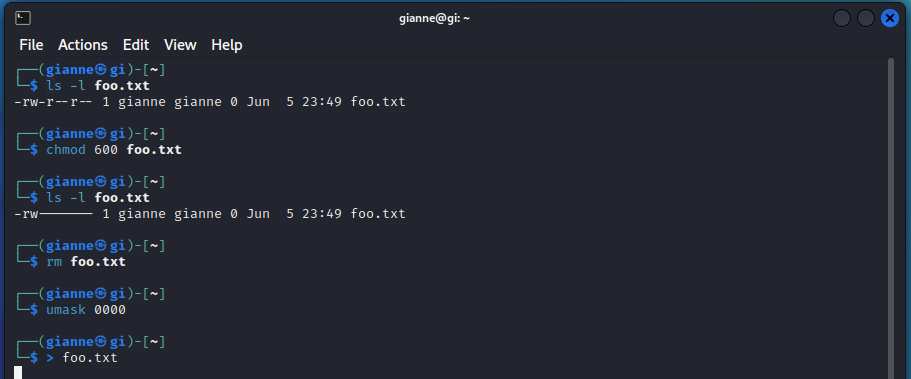
Let's look at the output. When user accounts are created, users are assigned a number called a user ID or uid which is then, for the sake of the humans, mapped to a username. The user is assigned a primary group ID or gid and may belong to additional groups. The above example is from a Fedora system. On other systems, such as Ubuntu, the output may look a little different:

**Reading, Writing, And Executing -** Access rights to files and directories are defined in terms of read access, write access, and execution access. If we look at the output of the ls command, we can get some clue as to how this is implemented:





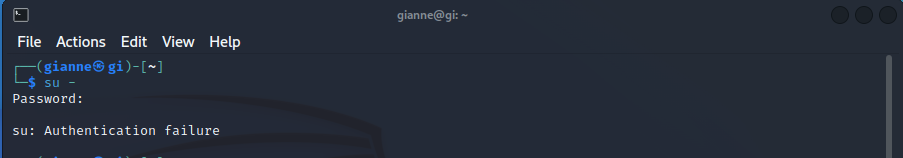


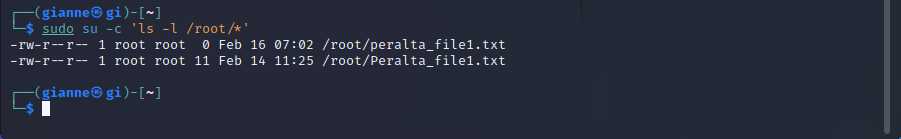
**umask** – Set Default Permissions



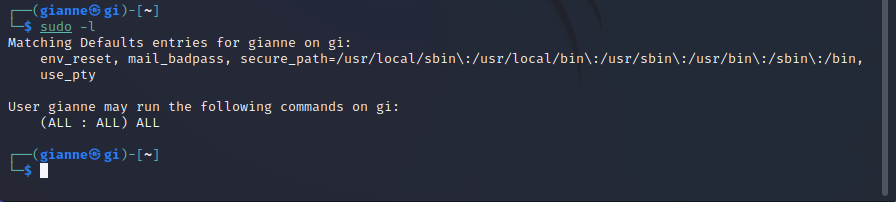
**Changing Identities Executing -** At various times, we may find it necessary to take on the identity of another user. Often we want to gain super-user privileges to carry out some administrative task, but it is also possible to “become” another regular user for such things as testing an account. There are three ways to take on an alternate identity:

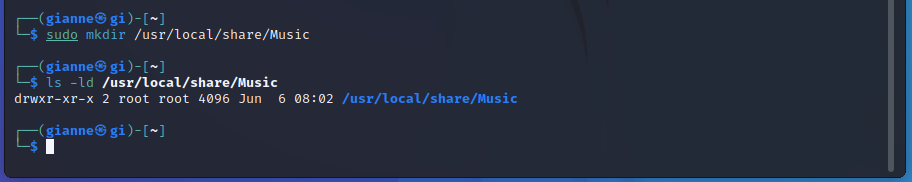
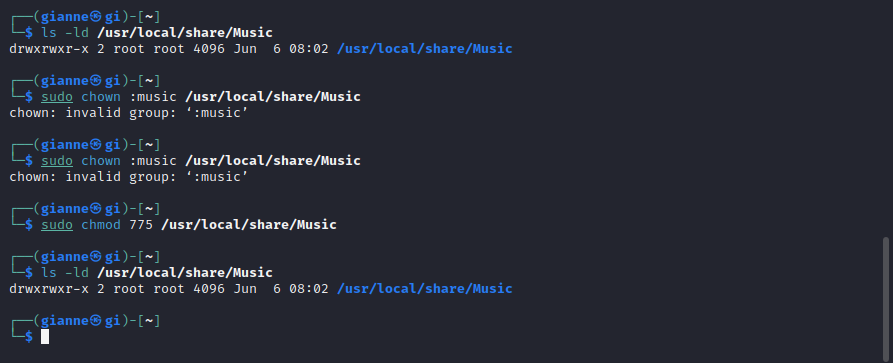
**su** – Run A Shell With Substitute User And Group IDs

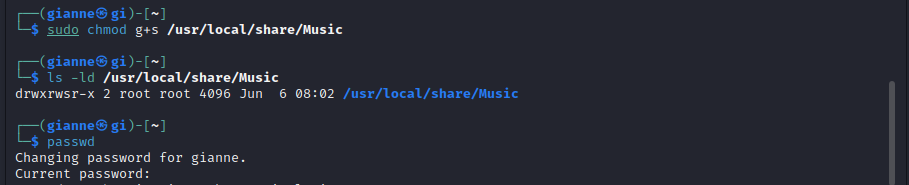




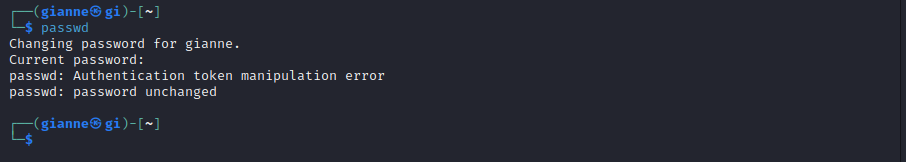
**sudo –** Execute A Command As Another User



**Exercising Our Privileges -** Now that we have learned how this permissions thing works, it's time to show it off. We are going to demonstrate the solution to a common problem — setting up a shared direc- tory.

****

**Changing Your Password** - The last topic we'll cover in this chapter is setting passwords for yourself (and for other users if you have access to superuser privileges.) To set or change a password, the passwd command is used. The command syntax looks like this:

****