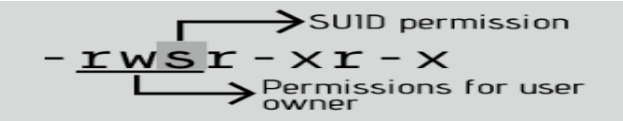
Module III: Permission and Advance Permission

Part II

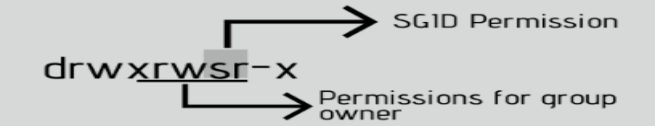
Concepts on SUID, SGID and STICKY BIT directories for collaboration:

SUID (Set User ID), SGID (Set Group ID), and the Sticky Bit are special flags that can be set on files and directories in Red Hat Enterprise Linux (RHEL) to control access and permissions. These flags can be used to facilitate collaboration among users in a shared environment.

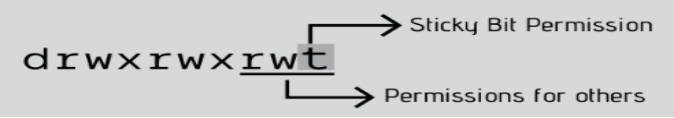
SUID is a special type of file permission that allows a user to execute a file with the permissions of the file's owner. This can be useful in situations where a user needs to execute a file or program that requires elevated privileges, such as a backup script or a program that manages system resources. For example, the passwd command is owned by the root user, but users are allowed to execute it because it has the SUID bit set. When a user runs passwd, it runs with the privileges of the root user, allowing the user to change their own password.



SGID is similar to SUID, but it applies to directories instead of files. When the SGID bit is set on a directory, new files created within that directory will inherit the group ownership of the directory, rather than the group ownership of the user who created the file. This can be useful in situations where multiple users need to collaborate on a project and share access to files and directories.



The Sticky Bit is a special flag that can be set on directories, which controls the ability of users to delete or rename files within that directory. When the Sticky Bit is set on a directory, only the owner of the file, the owner of the directory, or the root user can delete or rename files within that directory. This can be useful in situations where a shared directory is being used for important files that should not be accidentally deleted or modified.



In summary, SUID, SGID, and the Sticky Bit can be used to control access and permissions on files and directories in a shared environment, and can be used to facilitate collaboration among users by allowing them to share access to files and directories while still maintaining control over who can access, modify, or delete them.

In Red Hat Enterprise Linux (RHEL), you can set the SUID, SGID, and Sticky Bit permissions on files and directories using the **chmod** command.

1. To set the SUID bit on a file, use the command **chmod u+s filename**. For example, to set the SUID bit on a file named "myscript.sh", use the command **chmod u+s myscript.sh**. This will allow the file to be executed with the permissions of the file's owner.
2. To set the SGID bit on a directory, use the command **chmod g+s directoryname**. For example, to set the SGID bit on a directory named "mydir", use the command **chmod g+s mydir**. This will allow files created within the directory to inherit the group ownership of the directory.

3. To set the Sticky Bit on a directory, use the command **chmod +t directoryname**. For example, to set the Sticky Bit on a directory named "shareddir", use the command **chmod +t shareddir**. This will prevent users from deleting files that they do not own within the directory.

It's important to note that when using the chmod command, you can also use numeric notation to set the permissions. For example, to set the SUID, SGID and Sticky Bit on a file, you can use the command **chmod 6711 filename**. The first number is the owner permissions, the second is the group permissions, and the third is the other permissions. The number 7 is for setting all permissions (rwx) and 1 for execute only.

It's also important to note that using SUID, SGID, and Sticky Bit can have security implications. Be sure to use them with caution and only when necessary. It's always a good idea to review the permissions of files and directories on your system to ensure that they are set correctly and securely.

The following commands can be used to set special file attributes in RHEL (Red Hat Enterprise Linux):

1. SUID (Set User ID) - **chmod u+s filename**
2. SGID (Set Group ID) - **chmod g+s filename**
3. Sticky Bit - **chmod +t filename**
4. Remove SUID - **chmod u-s filename**
5. Remove SGID - **chmod g-s filename**
6. Remove Sticky Bit - **chmod -t filename**

In these commands, **filename** is the name of the file for which you want to set or remove the special attribute. The **u+s**, **g+s**, and **+t** options set the SUID, SGID, and Sticky Bit attributes, respectively, while the **u-s**, **g-s**, and **-t** options remove these attributes.

Special File Attributes:

Just beyond regular rwx/ugo permissions are file attributes. Such attributes can help you control what anyone can do with different files. While the lsattr command lists current file attributes, the chattr command can help you change those attributes. For example, the following command protects /etc/fstab from accidental deletion, even by the root administrative user:

chattr +i /etc/fstab

With that attribute, if you try to delete that file as the root administrative user, you'll get the following response:

rm /etc/fstab
rm: remove regular file `/etc/fstab'? y
rm: cannot remove `/etc/fstab': Operation not permitted

The lsattr command shows how the previous chattr +i command added the immutable attribute to /etc/fstab:

lsattr /etc/fstab
----i--------e- /etc/fstab

Of course, the root administrative user can unset that attribute with the following command. Nevertheless, the initial refusal to delete the file should at least give pause to that administrator before changes are made:

chattr -i /etc/fstab