# Chapter 31 Title - Notes

## 31.2 Introduction

Linux systems form collections of users called **groups**, whose members share common purposes. To further that end, they share certain files/directories and maintain some common privileges. This separates them from other on system, sometimes collectively called **world**. Using groups aids collaborative projects enormously.

## 31.3 Learning Objectives:

- Explain why it is useful to have Linux users belong to one or more groups.
- Use utlilities such as **groupadd**, **groupdel**, **groupmod**, and **usermod** to create, remove, and manipulate groups and their membership.
- Describe User Private Groups.
- Explain the concept of group membership.

## 31.4 Groups

Groups: collections of users whose members share some common purpose within Linux systems. Share certain files/directories, maintain some common privileges. Separates them from other on the system, sometimes collectively called the world. Using groups aids collaborative projects enormously. Users belong to one or more groups.

Groups defined in `/etc/group`, which has same role for groups as `/etc/passwd` has for users. Each line of the file looks like:

```
groupname:password:GID:user1, user2,...
```

where:

- `groupname` is the name of the group
- `password` is the password place holder. Group passwords may be set, but only if the `/etc/gshadow` file exists
- `GID` is the group identifier. Values between 0 and 99 are for system groups. Values between 100 and `GID_MIN` (as defined in `/etc/login.defs` and usually the same as `UID_MIN`) are considered special. Values over `GID_MIN` are for **UPG** (**U**ser **P**rivate **G**roups)
- `user1, user2,...` is a comma-separated list of users who are members of the group. The user need not be listed here is this groups is the user's principal group

## 31.5 Group Management

Group accounts may be managed/maintained with:

- **groupadd**: Add a new group
- **groupdel**: Remove a group
- **groupmod**: Modify a group's properties
- **usermod**: Modify a user's group memberships (add or remove).

Can also edit `/etc/group` directly, but better to use **vigr** utility, which is generally symbolically linked to **vipw** utility mentioned earlier.

These group manipulation utilities modify `/etc/group` and (if it exists) `/etc/gshadow`, and may only be executed by root:

**Examples:**

```
$ sudo groupadd -r -g 215 staff
$ sudo groupmod -g 101 blah
$ sudo groupdel newgroup
$ sudo usermod -G student,group1,group2 student
```

**Note**: Be very careful with **usermod -G** command: the group list that follows is the complete list of groups, not just the changes. Any supplementail groups left out will be gone! Non-destructive use should utilize `-a` option, which will preserve pre-existing group memberships when adding new ones.

## 31.6 User Private Groups

Linux uses User Private Groups (UPG).

Idea behind UPGs: each user will have his/her own group. However, UPGs are **not guaranteed** to be private. Additional members may be added to someone's private group in `/etc/group`.

By default, users whose accounts are created with **useradd** have: primary `GID` = `UID` and group name is also identical to user name.

As specified in `/etc/profile`, **umask** is set to `002` for all users created with UPG. Under this scheme, user files thus created with permissions `664` ( `rw-rw-r--` ) and directories with `775` ( `rwxrwxr-x` ). Will discuss **umask** in next section.

## 31.7 Group Membership

A Linux user has one **primary** group; listed in `/etc/passwd` and will also be listed in `/etc/group`. User may belong to between 0 and 15 **secondary groups**.

Primary group is the `GID` that is used whenever the user creates files or directories. Membership in other, secondary, groups grant user additional permissions.

Group membership can be identified by running either of the following commands:

```
$ groups [user1 user2 ...]
$ id -Gn [user1 user2 ...]
```

With no arguments, either command reports on current user. Note: default groups can differ by distribution:

On CentOS:

```
[student@CentOS7 ~]$ groups
student
[student@CentOS7 ~]$
```

On Ubuntu:

```
student@ubuntu:~$ groups
student adm cdrom sudo dip plugdev lpadmin sambashare libvirt
student@ubuntu:~$
```

##

Back to top

Previous Chapter - Table of Contents - Next Chapter