# User, Groups and sudo su

Prabhjeet Singh

# User and Groups

- Who logged into the system?
  We can find a user by 3 commands. - $ who, $user, $w

└─$ who

kali tty7 2022-08-27 21:25 (:0)

└─$ w

18:14:21 up 22:37, 1 user, load average: 0.03, 0.03, 0.00

USER TTY FROM LOGIN@ IDLE JCPU PCPU WHAT

kali tty7 :0 Sat21 44:48m 1:51 0.51s xfce4-session

└─$ users

Kali

$less etc/passwd

$less etc/group

$ ls -la /home -> to display ownership of the particular directory.

# File and Directory permission

**$ ls -l**

total 28

-rw-r--r-- 1 kali kali  114 Aug 29 06:08  combine1.txt

-rw-r--r-- 1 kali kali    0 Aug 29 08:17  file

-rw-r--r-- 1 kali kali  104 Aug 29 17:14  file1_hard.txt

lrwxrwxrwx 1 kali kali    9 Aug 29 17:13  file1_soft.txt -> file1.txt

-rw-r--r-- 1 kali kali   44 Aug 29 06:07  file2.txt

-rw-r--r-- 1 kali kali   21 Aug 29 07:03  file3.txt

-rw-r--r-- 1 kali kali    0 Aug 29 08:17 'file   space.txt'

-rw-r--r-- 1 kali kali    0 Aug 29 08:17 'file space.txt'

drwxr-xr-x 2 kali kali 4096 Aug 29 18:38  my_dir1

drwxr-xr-x 2 kali kali 4096 Aug 29 18:38  my_dir2

-rw-r--r-- 1 kali kali    0 Aug 29 08:17  space.txt

-rw-r--r-- 1 kali kali  524 Aug 29 17:19  zip__backup.zip

1. First place holder for the type for file it is – and directory – d
2. Next 3 char for the permissions of the **owner** of the file or directory.
3. Next 3 char for the permissions of the **users of the same group** as owner of the file or directory.
4. Next 3 char for the permission of **other users**.

# Change the permission of file or directory with chmod command

$chmod g-w file1.txt → it will remove write permission of the group

$chmod a=,u=r file1.txt → it will remove all permissions for everyone and add read permission to the owner.

**Octal permission**

Read – 4

Write – 2

Execute - 1

6 -> only read and write

7 - > read, write and execute

4 -> only read

$chmod 664 file1.txt → owners has read and write, group has read and write and others Have only read.

**Change owner and group of the file**

$chown bob file1.txt → $chown <new owner> <file name> → it will change the owner of file

$chgrp john file1.txt → it will change the group

**SUDO and su command**

**sudo** - Execute command as root or admin user only once. sudo allows a permitted user to execute a command as the superuser or another user, as specified by the security policy.

**su** → switch user. it will change the current user.

su allows commands to be run with a substitute user and group ID. When called with no user specified, su defaults to running an interactive shell as root. When user is specified, additional arguments can be supplied, in which case they are passed to the shell

**$sudo -u bob cat file1.txt**

Enter password and command will work.

**su commnd example**

(kali㉿kali)-[~/linux_learning]

└─**$ su root**
130 ×

Password:

┌──(root 💀 kali)-[/home/kali/linux_learning]

└─# ls

combine1.txt file1_hard.txt file2.txt 'file space.txt' my_dir1 space.txt

file file1_soft.txt file3.txt 'file space.txt' my_dir2 zip__backup.zip

┌──(root 💀 kali)-[/home/kali/linux_learning]

└─# exit

┌──(kali㉿kali)-[~/linux_learning]

└─$

## Change password - $passwd

The passwd command changes passwords for user accounts. A normal user may only change the password for their own account, while the superuser may change the password for any account. passwd also changes the account or associated password validity period.

**Change password for current user**

$passwd

**Change password for the other user**

$sudo passwd bob