* Linux accounts and groups (users and permissions pt 1)

* Linux account.

3 user categories.

① Superuser account

- Root user - unrestricted permissions
- for administrative task: need to login as root user or execute the command as root (sudo command)

②. user account

- a regular user we create a login.
- eg tom - /home/tom.

③ service account

- Relevant for Linux Server Distro
- each service will get its own user
- mysql user will start mysql application.
- Best practice for security.
- Don't run services with root user.

per server or per computer.

- 1 root user.
- multiple regular user and service users.

Why multiple standard users?

- Share computer
- many companies use windows for their employees
- usually employee can login to your account

on every computer
- same thing with universities.
- In windows, it is able to centrally manage users.
  - admins add users to the system. all computer are connected to the system
  - only access to your home folder.

So you can Login to any hardware that is connected to this system. no hardware bound.

Linux doesn't have centrally managed system
- user accounts are managed on that specific hardware

* multiple users on a server.
  - for linux having multi-user is imp for server.
  - usually teams administer a server. Why not just shared user?

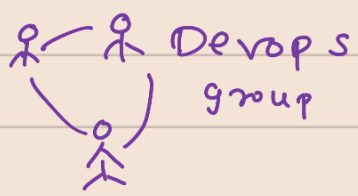
Why having user for each team member is imp.
- They need a non root user
- Permissions per team member
- Traceability - who did what on the system?

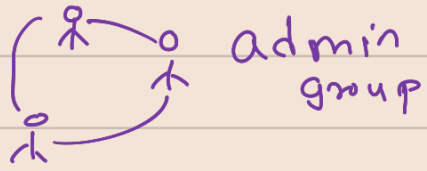
* Groups and permissions.

How to manage permissions
- 2 level of permissions

① User level
   give permission to user directly.
② Group level

- Group users in Linux groups.
- Give permission to the group.
- The way to go, if you manage multiple users





- users are added to the group.
- permissions to that group.

each user can also be a part of multiple group

* user management in practice.

① all the users, system has is actually stored.

/etc/passwd

- stores users account information
- everyone can read it, but only root user can change the file.

cat /etc/passwd.

o/p → each line represents a user.

eg- yash : x : 1000 : 1000 : yash,,, : /home/yash : /bin/bash

yash → username

x → password (encrypted) → /etc/shadow (stored)

1000 → user ID. 0 → for root

1000 → Group ID

yash,,, → user ID info.

/home/yash - user home directory.

/bin/bash - user Default Shell

② Create a new user

sudo adduser tom

② Change password

```
sudo passwd tom
```

③ Change user login

```
su - tom
```

④ login as root

```
su -
```

⑤ Create a group.

```
sudo groupadd devops
cat /etc/group → check all the groups.
```

⑥ Change primary group of user Tom to devops.

```
sudo usermod -g devops tom
```

devops → group name the user should be in

tom → user name

```
sudo delgroup tom
```

⑦ adding user to multiple group

```
sudo usermod -G admin tom
```

-G → overwrites the whole secondary group list.

if you want to add user to a new secondary group in addition to the existing one

```
sudo usermod -aG newgroup tom
```

⑧ current user belongs to how many group.

```
groups
```

⑨ particular user belongs to how many groups

```
groups tom
```

⑩ add user by specifying the group at the creation time.

sudo adduser -G devops Nic

(11). remove a particular user from a group.

sudo gpasswd -d Nic devops

-d → delete
Nic → user
devops → group