User Management

Managing users is essential for securing your Linux system and controlling permissions. This handout covers user creation, modification, deletion, privilege escalation, password management, and more.

---

1. Types of Users in Linux:

Linux systems have **three main types of users**:

- **Root User (`UID 0`)**
  - Highest privileges
  - Can perform any action
- **Regular Users**
  - Limited privileges
  - Can temporarily get elevated privileges via `sudo`
- **Service Users**
  - Used by services (e.g., web server)
  - Isolated from regular users for enhanced security

---

2. Important User Management Files

- `/etc/passwd`
  - Contains basic user info: username, UID, GID, description, home directory, and default shell.
  - Does **not** store passwords.
- `/etc/shadow`
  - Contains encrypted passwords and password-aging details.
  - Only readable by root or privileged users.
- `/etc/group`
  - Stores group info and membership.

---

3. Creating Users (`useradd`)

Basic syntax:

```
sudo useradd -m username
```

- `m` → Creates home directory (`/home/username`)
- `d /custom/home` → Set custom home directory
- `s /bin/bash` → Set default shell explicitly
- `g primary_group` → Set primary group
- `G secondary_groups` → Add secondary groups (comma-separated)

Example:

```
sudo useradd -m -s /bin/bash alice
```

Check newly created user details:

```
cat /etc/passwd | grep alice
```

---

4. Managing Passwords (`passwd`)

Set/change user passwords:

```
sudo passwd username
```

Display password status:

```
passwd -S username
```

Set password expiry:

```
sudo passwd -n 7 -x 30 username
```

- `n` → Minimum days before changing again.
- `x` → Maximum days password is valid.

---

5. Modifying Users (`usermod`)

Modify user details:

```
sudo usermod [options] username
```

Common options:

- `c "Full Name"` → Set user description (full name)
- `s /bin/bash` → Change default shell
- `d /new/home -m` → Move home directory to new location
- `l new_username` → Change Unix username
- `g new_primary_group` → Change primary group
- `G additional_groups` → Add secondary groups

Example (changing shell and description):

```
sudo usermod -s /bin/bash -c "Alice B" alice
```

---

6. Deleting Users (`userdel`)

Basic deletion:

```
sudo userdel username
```

Options:

- `r` →Remove home directory and mail
- `f` → Force removal even if user is logged in

Example:

```
sudo userdel -r max
```

---

7. Switching Users (`su`)

Switch to another user:

```
su username
```

Switch to root:

```
su -
```

- Requires **target user's password** (not your current user's password).

---

8. Temporarily Changing Privileges (`sudo`)

Execute commands with root privileges (temporarily):

```
sudo command
```

Examples:

- Install software → `sudo apt-get install vim`
- Open root shell → `sudo -s`

Configure sudo access via:

- `/etc/sudoers` (edited safely via `visudo`)

Allow user full sudo access:

```
username ALL=(ALL:ALL) ALL
```

Allow **passwordless sudo** for a specific command (careful ❗):

```
username ALL=(ALL) NOPASSWD: /usr/bin/apt-get
```

- Use `which` command to find exact command path (e.g., `which apt-get`).

---

9. Sudo into Another User (`sudo -u`)

Execute commands as a different user (without their password):

```
sudo -u username command
```

Start shell as another user:

```
sudo -u alice -s
```

Example (create a file as another user):

```
sudo -u alice touch /home/alice/file_sudo.txt
```

📌 Summary of Key Commands

Command	Description
`sudo useradd -m username`	Create a user with home directory
`sudo passwd username`	Set/change user's password
`sudo passwd -n 7 -x 30 username`	Set min/max password age (days)
`sudo usermod -s /bin/bash username`	Change user's shell
`sudo usermod -c "Full Name" username`	Change user's full name/description
`sudo userdel -r username`	Delete user and home directory
`su - username`	Switch to another user
`sudo -u username command`	Run command as another user
`sudo -s`	Start a root shell session
`sudo visudo`	Safely edit sudo configuration