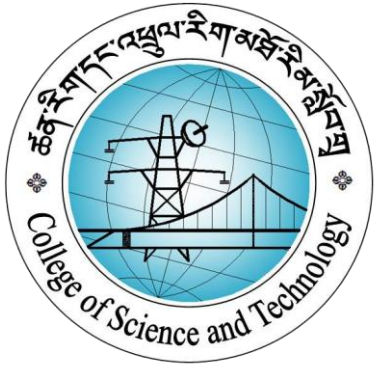

LESSON – 17-2

SSH SERVER CONFIGURATION

LEARNING OUTCOMES

- Limiting Root Access
- Configuring Alternative Ports,
- Limiting User Access
- Configuring Key-Based Authentication with Passphrases.

SSH Server Configuration in Ubuntu

- To secure your SSH server on Ubuntu, you'll modify the /etc/ssh/sshd_config file.
- Must restart the SSH service for the modifications to take effect after each change.
- Example:
  - sudo service restart sshd
    or
  - sudo systemctl restart sshd

Limiting Root Access

- Disable direct root login to prevent brute-force attacks on the most powerful account.

  vim /etc/ssh/sshd_config

- Find the line *#PermitRootLogin prohibit-password* and change it to:

  PermitRootLogin no

Configuring Alternative Ports

- Changing the default SSH port (22) helps to avoid constant scans from bots.

vim /etc/ssh/sshd_config

find the line #Port 22.

- Uncomment it and change the number to a port of your choice, for example, 2222 and restart the ssh service.

Configuring Alternative Ports

- Changing the default SSH port (22) helps to avoid constant scans from bots.

vim /etc/ssh/sshd_config

find the line #Port 22.

- Uncomment it and change the number to a port of your choice, for example, 2222 and restart the sshd service.

Limiting User Access

- Restrict SSH access to a specific group of users to enhance security.
  - First, create a new group for SSH users.

    sudo groupadd ssh-users

  - Add the user you want to allow to this group.

    sudo usermod -aG ssh-users your_username

  - In the **sshd_config** file, add the AllowGroups directive.

    AllowGroups ssh-users

Configuring Key-Based Authentication with Passphrases

- Most secure way to authenticate.
- The passphrase acts as an extra layer of protection for your private key.
- Generate the Key Pair: On your local machine, create a key pair. You will be prompted to enter a strong passphrase.

```
ssh-keygen -t ed25519
```

- Pushing the public key to the host *authorized_keys* file.