# Placement Empowerment Program

***Cloud Computing and DevOps Centre***

Up SSH Key-Based Authentication Locally: Generate an SSH key pair and configure it for passwordless login between two local machines or VMs

Name: Shahana.M.S Department: ADS



**Introduction**

SSH key-based authentication enhances security by allowing passwordless, encrypted access between two machines or virtual machines (VMs). Instead of relying on passwords, SSH keys use a cryptographic key pair to authenticate users securely. This method is widely used for remote administration, automation, and secure communication between systems.

**Objective**

* Generate an SSH key pair (public and private keys).
* Configure key-based authentication for secure, passwordless SSH login.
* Improve security by eliminating password-based authentication.
* Enable seamless remote access between two local machines or VMs.

**Overview**

Setting up SSH key-based authentication involves creating an SSH key pair on the client machine and adding the public key to the authorized keys on the remote machine. This process allows secure login without requiring a password. The private key remains on the client machine, while the public key is stored on the server or another machine.

**Importance of SSH Key-Based Authentication**

1. **Enhanced Security** 🔒
   * Eliminates the need for passwords, reducing the risk of brute-force attacks.
   * Uses strong cryptographic algorithms to authenticate users securely.
   * Prevents password leaks and credential theft.
2. **Convenience & Efficiency** ⚡
   * Allows seamless, passwordless login, eliminating the need to remember or enter passwords repeatedly.
   * Speeds up access to remote machines for system administrators and developers.
3. **Automation & DevOps Integration** 🤖
   * Enables secure, automated SSH access for scripts, CI/CD pipelines, and remote jobs.
   * Essential for cloud infrastructure management, where servers frequently communicate with each other.
4. **Reduces Attack Surface** 🚨
   * Disables password authentication, preventing unauthorized users from attempting login attempts using weak or stolen credentials.
   * Helps comply with security best practices for IT and cloud environments.
5. **Data Encryption** 🔐
   * SSH keys use encryption to protect data during transmission, preventing eavesdropping or man-in-the-middle attacks.
   * Ensures that remote access is both secure and confidential.
6. **Scalability & Centralized Access Control** 🌍
   * Simplifies access management by distributing public keys across multiple servers without sharing passwords.
   * Ideal for managing a large number of remote servers in enterprise environments.

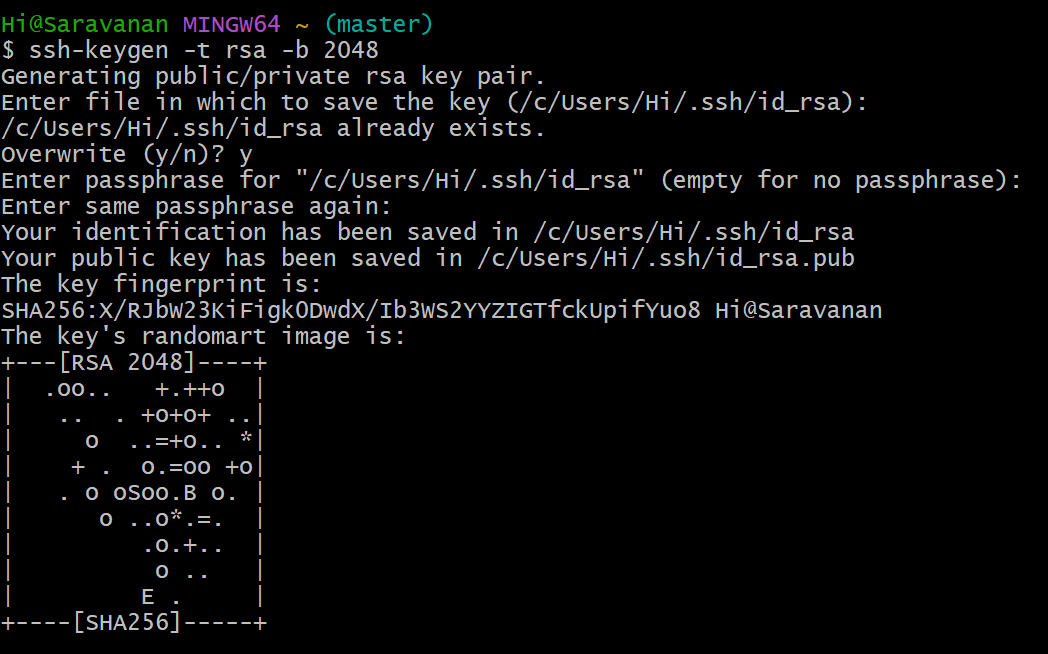
# Step-by-Step Overview

## Step 1:

### Generate SSH Key Pair

1. Open Git Bash or Terminal on the local machine or your system(Git Bash is a command-line tool for Windows that provides a Unix-like environment to run Git commands and other Bash utilities.)

Enter the following commads and enter Y for confirmity

Optionally, you can set a passphrase for extra security, but if you want passwordless login, leave it empty and press **Enter**.



**Step 2: Copy Public Key to Target Machine**

* Copy the public key to the target machine using the ssh-copy-id command.
* If ssh-copy-id is unavailable, manually copy the key by displaying it on the source machine and pasting it into the ~/.ssh/authorized\_keys file on the target machine.

**Step 3: Set Correct File Permissions**

* Ensure that the .ssh directory and authorized\_keys file on the target machine have the correct permissions to allow secure SSH authentication.

**Step 4: Test Passwordless SSH Login**

* Attempt to SSH into the target machine from the source machine.
* If configured correctly, the login should proceed without asking for a pa

**Outcome**

* Successfully configured SSH key-based authentication between two machines.
* Secure, passwordless login, reducing security risks associated with passwords.
* Improved automation capabilities for remote tasks and scripts.
* Enhanced system security with encrypted SSH access.

.