

## How to Set Up Passwordless Authentication on EC2 Instances

### Method 1: Using `ssh-copy-id` with Default Key

#### 1. Login with Private Key (if needed):

```
ssh -i ~/.ssh/rons-ec2-key.pem ubuntu@<INSTANCE-PUBLIC-IP>
```

#### 2. Ensure Password Authentication is Enabled (Optional):

- Edit the SSH config file on the EC2 instance:

```
sudo nano /etc/ssh/sshd_config
```

- Set:

```
PasswordAuthentication yes
```

- Restart SSH:

```
sudo systemctl restart ssh
```

#### 3. Set a Password for Ubuntu User (if not already set):

```
sudo passwd ubuntu
```

#### 4. Copy the Public Key to the Server:

```
ssh-copy-id ubuntu@<INSTANCE-PUBLIC-IP>
```

#### 5. Test Passwordless SSH Login:

```
ssh ubuntu@<INSTANCE-PUBLIC-IP>
```

---

### Method 2: Using `ssh-copy-id` with Explicit PEM File

#### 1. Run the command to copy the public key using the PEM file:

```
ssh-copy-id -f "-o IdentityFile <PATH TO PEM FILE>" ubuntu@<INSTANCE-PUBLIC-IP>
```

- `ssh-copy-id`: Command to copy public key.
- `-f`: Forces copying, even if a key already exists.
- `-o IdentityFile <PEM FILE>`: Uses a specific private key file for authentication.
- `ubuntu@<INSTANCE-PUBLIC-IP>`: The username and EC2 instance’s public IP.

2. Test Passwordless Login:

```
ssh ubuntu@<INSTANCE-PUBLIC-IP>
```

Key Differences Between the Methods

Feature	Method 1: Default Key Setup	Method 2: Explicit PEM File Setup
Key Usage	Uses the default SSH key ( <code>~/.ssh/id_rsa.pub</code> )	Uses a specific <code>.pem</code> key explicitly provided
Flexibility	Works if your default key is already loaded	Useful when using a custom private key
Command Simplicity	No extra flags needed, easier for quick setup	Requires specifying the private key manually
Use Case	Best for users who always use the same key	Best for multiple keys or external <code>.pem</code> files

Both methods achieve the same goal, but **Method 2 is better when managing multiple SSH keys**, while **Method 1 is simpler if using a single default key.** 🚀