How to SSH from One AWS EC2 Instance to Another

If you need to connect from one EC2 instance to another within the same AWS environment, follow these steps. This is useful for managing multiple servers, setting up internal communication, or running distributed applications.

---

1 Ensure Both EC2 Instances Exist

Before connecting, make sure:

- Both EC2 instances are **running**.
- You have the **private key** (`.pem` **file**) for the instance you are connecting to.
- Both instances are in the **same VPC** (if using private IPs).

---

2 Update Security Groups for SSH Access

- Go to **AWS EC2 Dashboard → Security Groups**.
- Select the **security group** of the target EC2 instance.
- Under **Inbound Rules**, add a new rule:
    - **Type:** SSH
    - **Protocol:** TCP
    - **Port Range:** 22
    - **Source:**
        - **For Private Connection:** Use the **private IP of the source instance** or its security group ID.
        - **For Public Connection:** Use **My IP** (`0.0.0.0/0`) (not recommended for security reasons).
- Save changes.

---

3 Find the Target Instance's Private IP

On the AWS EC2 Dashboard:

- Click on the target instance.
- Copy its **Private IPv4 Address** (if connecting within the same VPC) or **Public IPv4 Address** (for external SSH).

4 Transfer the Private Key to the Source EC2 Instance

If the private key (`.pem` file) is on your local machine, copy it to the source instance using **SCP (Secure Copy Protocol)**:

```bash
CopyEdit
scp -i my-key.pem my-key.pem ubuntu@source-instance-ip:~/
```

Move the key to a secure location and set proper permissions:

```bash
CopyEdit
mv my-key.pem ~/.ssh/
chmod 400 ~/.ssh/my-key.pem
```

5 SSH into the Target EC2 Instance from the Source EC2 Instance

Using the Private IP (Recommended for Internal Network)

If both instances are in the same **VPC**, connect using the private IP:

```bash
CopyEdit
ssh -i ~/.ssh/my-key.pem ubuntu@private-ip-of-target-instance
```

Using the Public IP (If Not in Same VPC)

If the target instance is in a different VPC or external network, use the public IP:

```bash
CopyEdit
ssh -i ~/.ssh/my-key.pem ubuntu@public-ip-of-target-instance
```

Note:

- **Amazon Linux**: Use `ec2-user` instead of `ubuntu`.
- **RHEL/CentOS**: Use `centos` or `ec2-user`.

6 Verify Connection & Perform Actions

Once connected, you can run commands on the remote instance:

```bash
CopyEdit
uname -a  # Check system info
ls        # List files
whoami    # Verify user
```

7 (Optional) SSH Without Using the Key Every Time

To simplify connections, add the target instance to your SSH config file:

```bash
CopyEdit
nano ~/.ssh/config
```

Add the following:

```bash
CopyEdit
Host target-ec2
  HostName private-ip-of-target-instance
  User ubuntu
  IdentityFile ~/.ssh/my-key.pem
```

Save and exit, then connect using:

```bash
CopyEdit
ssh target-ec2
```

Conclusion

By following these steps, you can successfully **SSH from one EC2 instance to another**, ensuring secure and efficient communication between servers