AWS Systems Manager (SSM)

What is SSM?

SSM (AWS Systems Manager) allows you to manage EC2 instances securely without using SSH **(Putty)** or **Remote Desktop** (RDP)**.** It also enables command execution across one or more instances.

Purpose of Using SSM

- Connect to Linux/Windows EC2 instances **without SSH or RDP**
- **No need to open ports** (22 or 3389) in Security Group
- **No need for key pairs**
- Execute **commands across multiple EC2 instances** from the AWS console
- Centralized, auditable and secure access management

Prerequisites

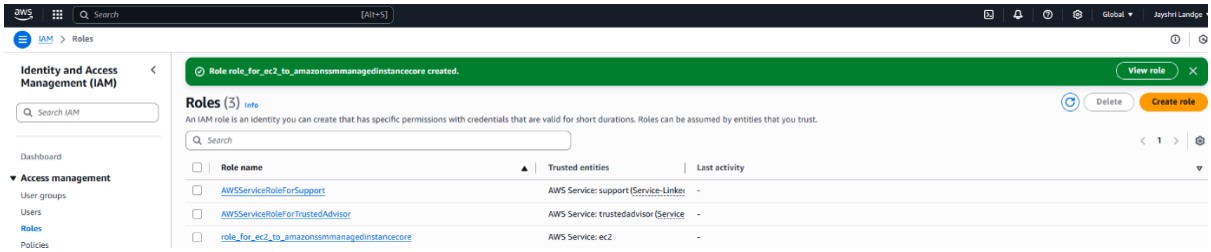
Before you connect EC2 instances using SSM, ensure the following:

- EC2 instance has **SSM Agent installed and running**
- EC2 instance has an **IAM Role** attached with proper permissions
- EC2 is in a **public or private subnet with internet access (via NAT Gateway or IGW)**
- The instance is using an **SSM-compatible AMI**

Step-by-Step Guide to Connect EC2 using SSM

Step 1: Create IAM Role for EC2

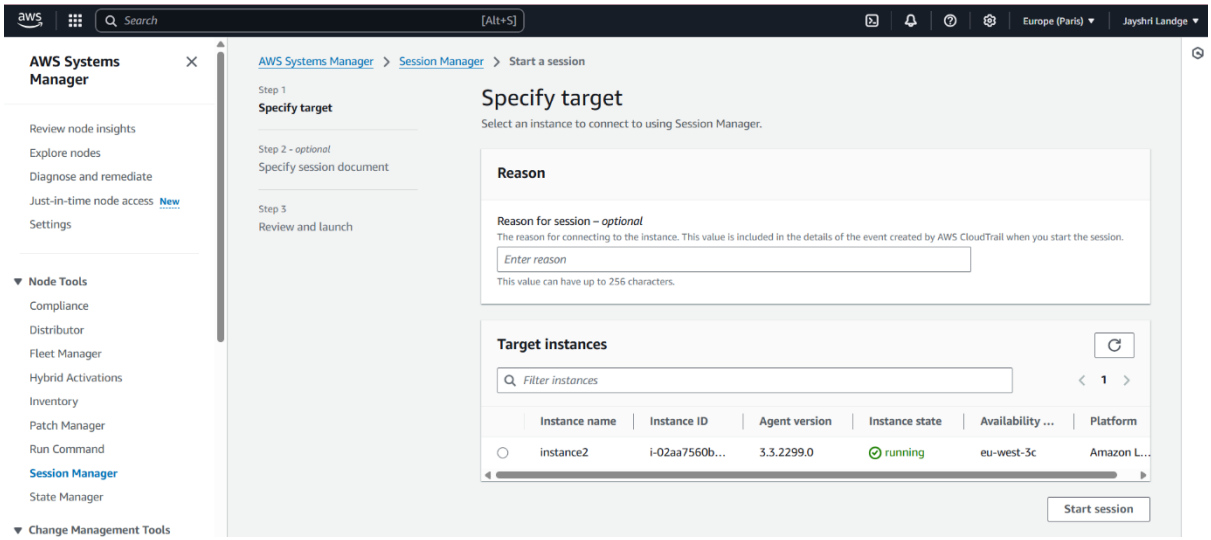
1. Go to **IAM Console**
2. Create a Role with:
    - **Trusted entity**: EC2
    - **Permissions policy**: AmazonSSMManagedInstanceCore



Screenshot 1: IAM role creation

Step 2: Launch EC2 Instance with IAM Role

1. Launch an EC2 instance
2. Attach the IAM Role created in Step 1
3. Skip key pair
4. No need to open port 22 or 3389 in Security Group



Screenshot 2: EC2 configuration with IAM role

Step 3: (If Required) Install & Start SSM Agent

Some AMIs may not come with the SSM Agent pre-installed.

Check availability

yum list amazon-ssm-agent

Install it

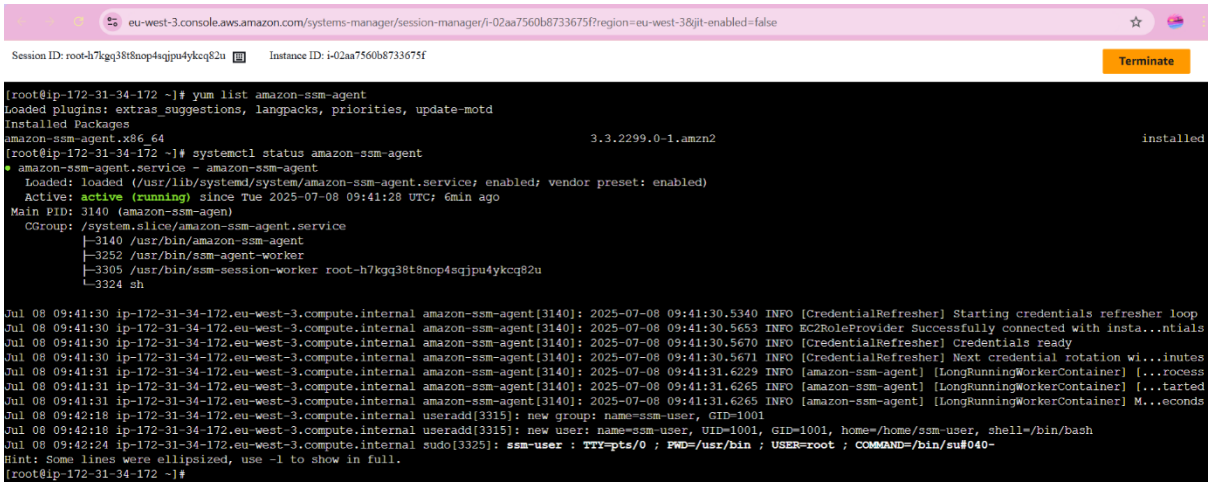
sudo yum install amazon-ssm-agent -y

Check service status

systemctl status amazon-ssm-agent

Start the agent

sudo systemctl start amazon-ssm-agent

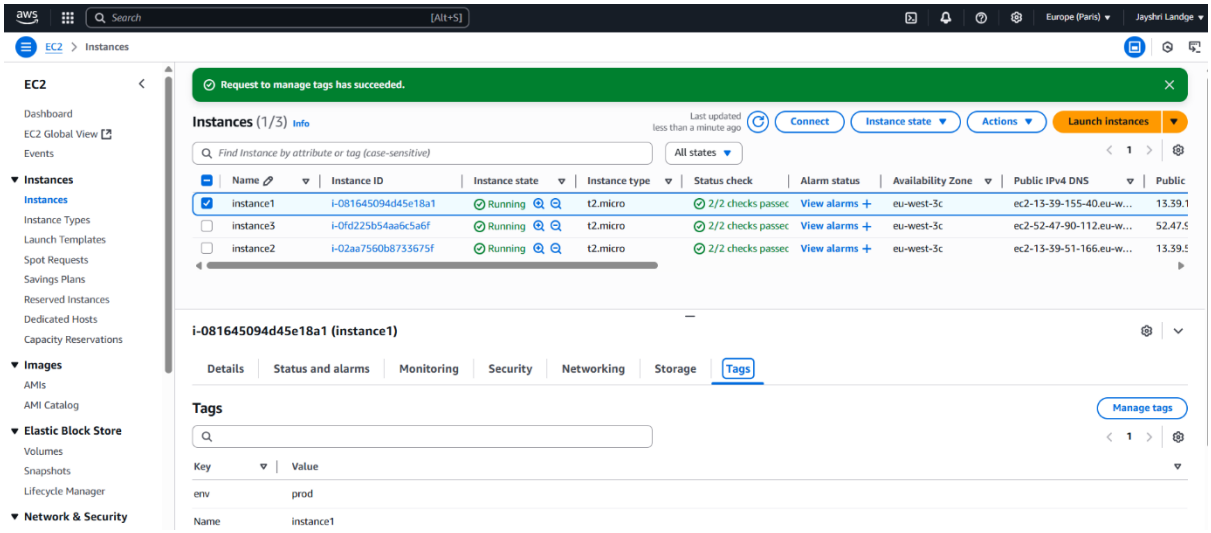


Screenshot 3: SSM agent installation

Step 4: Tag Your EC2 Instances

1. Go to EC2 → **Tags** section
2. Add tags like:
    - Key: dev | Value: prod

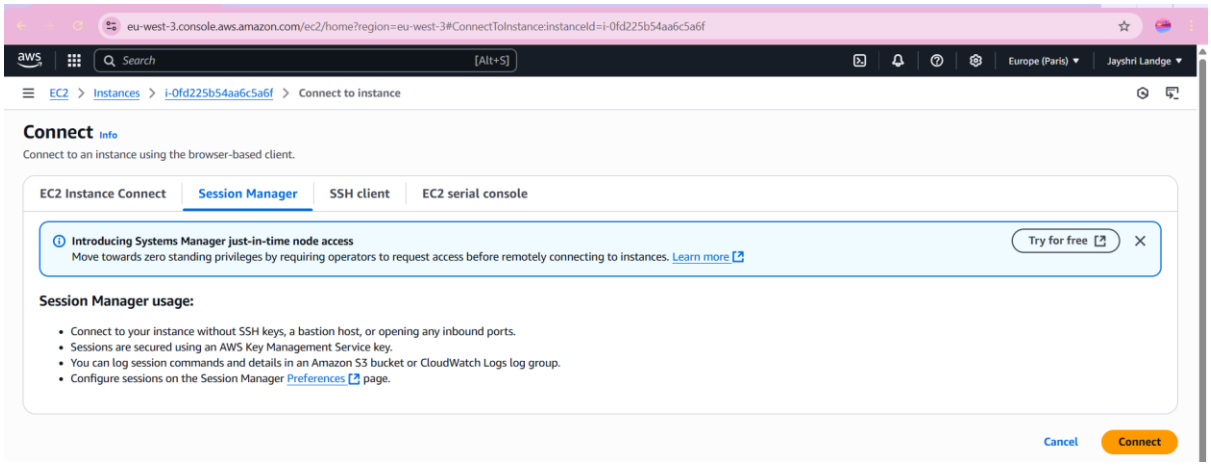
Tags help in **grouping and selecting instances** while executing commands via SSM.



Screenshot 4: instance tag creation

Step 5: Connect to EC2 Instance using SSM

1. Go to EC2 Console
2. Select your instance
3. Click **Connect → Session Manager tab → Connect**

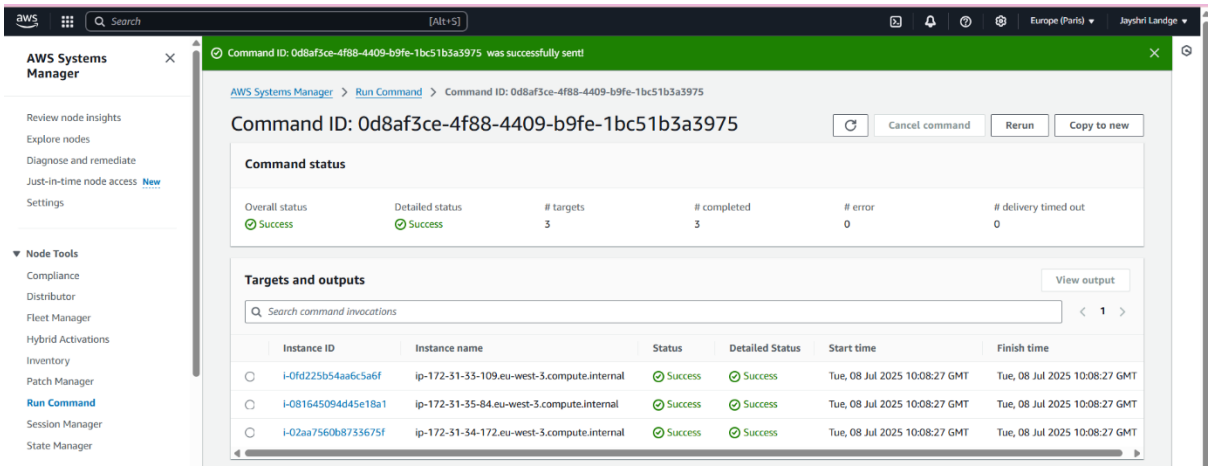


Screenshot of Session Manager connection

Step 6: Run Commands on Multiple EC2 Instances

1. Go to **Systems Manager Console → Run Command**
2. Choose document: AWS-RunShellScript
3. Target:
    - Choose EC2 instances manually
    - Or use **tags** to target a group of instances
4. Add your commands (e.g., update packages)

sudo yum update -y



Screenshot of Run Command setup

Benefits of Using SSM

- **No open inbound ports** required
- **No SSH keys or RDP credentials** needed
- **Run commands across multiple instances** at once
- Centralized and **auditable session logs**
- Supports **hybrid environments** with on-premises servers
-

Conclusion

AWS SSM is a powerful and secure solution for EC2 instance management. It eliminates the need for key pairs and open ports, enhances security, and offers scalable features like tagging and command execution on multiple machines. It's a **must-have tool** for cloud administrators and DevOps engineers.