

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

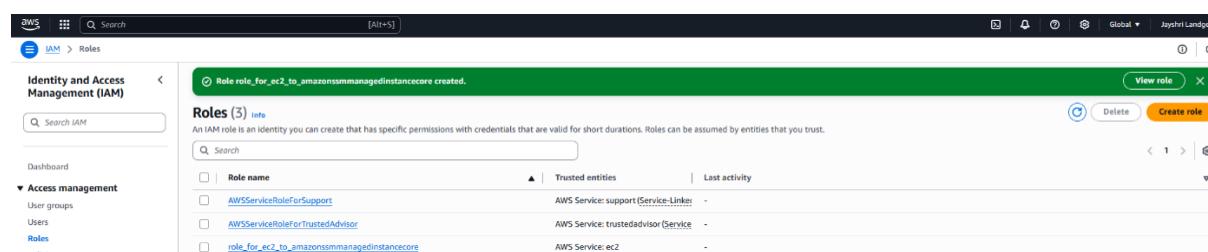
Before you connect EC2 instances using SSM, ensure the following:

- 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

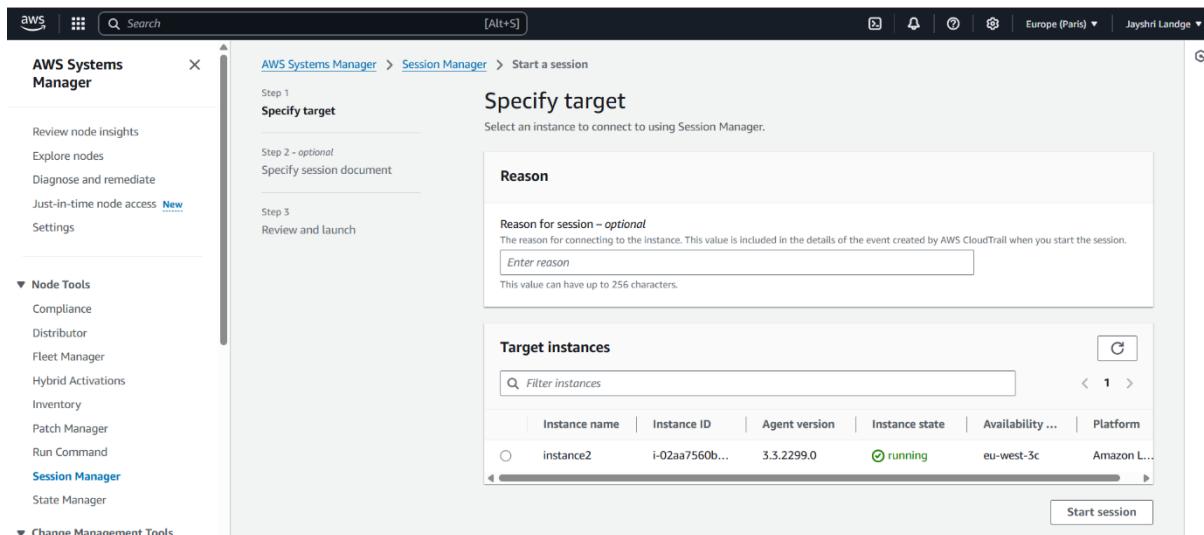
1. Go to **IAM Console**
2. Create a Role with:
 - **Trusted entity:** EC2
 - **Permissions policy:** AmazonSSMManagedInstanceCore



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
```

```
[root@ip-172-31-34-172 ~]# yum list amazon-ssm-agent
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Installed Packages
amazon-ssm-agent.x86_64          3.3.2299.0-1.amzn2
amazon-ssm-agent.service - amazon-ssm-agent
  Loaded: loaded (/usr/lib/systemd/system/amazon-ssm-agent.service; enabled; vendor preset: enabled)
  Active: active (running) since Tue 2025-07-08 09:41:28 UTC; 6min ago
    Main PID: 3140 (amazon-ssm-agent)
   Group: /system.slice/amazon-ssm-agent.service
           ├─3140 /usr/bin/amazon-ssm-agent
           ├─3252 /usr/bin/ssm-agent-worker
           ├─3305 /usr/bin/ssm-session-worker root-h7kgq38t8nop4sqjpu4ykcq82u
           └─3324 sh

Jul 08 09:41:30 ip-172-31-34-172.eu-west-3.compute.internal amazon-ssm-agent[3140]: 2025-07-08 09:41:30.5340 INFO [CredentialRefresher] Starting credentials refresher loop
Jul 08 09:41:30 ip-172-31-34-172.eu-west-3.compute.internal amazon-ssm-agent[3140]: 2025-07-08 09:41:30.5653 INFO EC2RoleProvider Successfully connected with insta...ntials
Jul 08 09:41:30 ip-172-31-34-172.eu-west-3.compute.internal amazon-ssm-agent[3140]: 2025-07-08 09:41:30.5670 INFO [CredentialRefresher] Credentials ready
Jul 08 09:41:30 ip-172-31-34-172.eu-west-3.compute.internal amazon-ssm-agent[3140]: 2025-07-08 09:41:30.5671 INFO [CredentialRefresher] Next credential rotation wi...nutes
Jul 08 09:41:31 ip-172-31-34-172.eu-west-3.compute.internal amazon-ssm-agent[3140]: 2025-07-08 09:41:31.6229 INFO [amazon-ssm-agent] [LongRunningWorkerContainer] (...rocess
Jul 08 09:41:31 ip-172-31-34-172.eu-west-3.compute.internal amazon-ssm-agent[3140]: 2025-07-08 09:41:31.6265 INFO [amazon-ssm-agent] [LongRunningWorkerContainer] (...rted
Jul 08 09:42:18 ip-172-31-34-172.eu-west-3.compute.internal useradd[3315]: new group: name=ssm-user, GID=1001
Jul 08 09:42:18 ip-172-31-34-172.eu-west-3.compute.internal useradd[3315]: new user: name=ssm-user, UID=1001, home=/home/ssm-user, shell=/bin/bash
Jul 08 09:42:24 ip-172-31-34-172.eu-west-3.compute.internal sudo[3325]: ssm-user : TTY=/pts/0 ; PWD=/usr/bin ; USER=root ; COMMAND=/bin/su#040-
Hint: Some lines were ellipsized, use -l to show in full.
[root@ip-172-31-34-172 ~]#
```

Screenshot 3: SSM agent installation

Step 4: Tag Your EC2 Instances

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

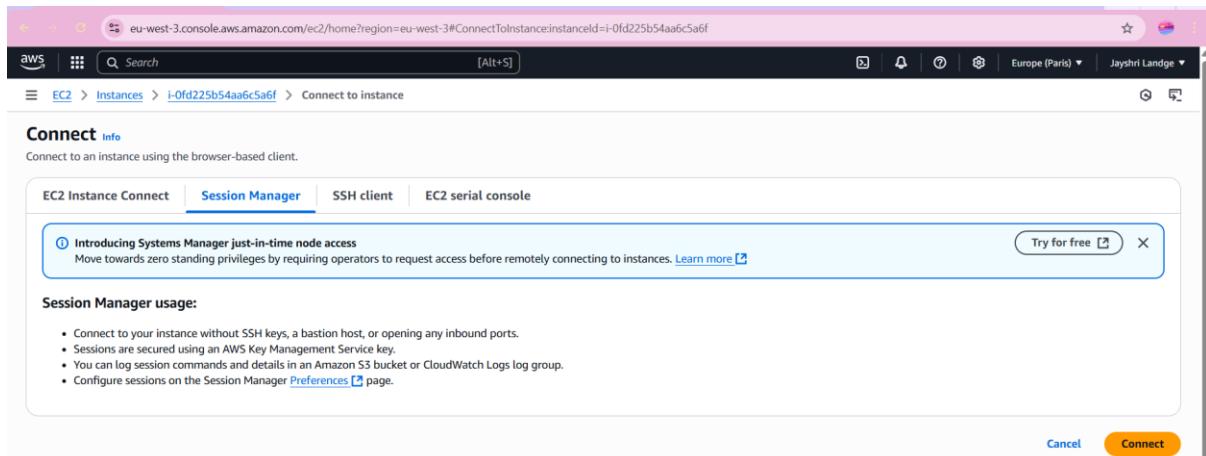
Tags help in **grouping and selecting instances** while executing commands via SSM.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public
instance1	i-081645094d45e18a1	Running	t2.micro	2/2 checks passed	View alarms +	eu-west-3c	ec2-13-39-155-40.eu-w...	13.39.1
instance5	i-0fd225b54aa6c5a6f	Running	t2.micro	2/2 checks passed	View alarms +	eu-west-3c	ec2-52-47-90-112.eu-w...	52.47.5
instance2	i-02aa7560b8733675f	Running	t2.micro	2/2 checks passed	View alarms +	eu-west-3c	ec2-13-39-51-166.eu-w...	13.39.5

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**

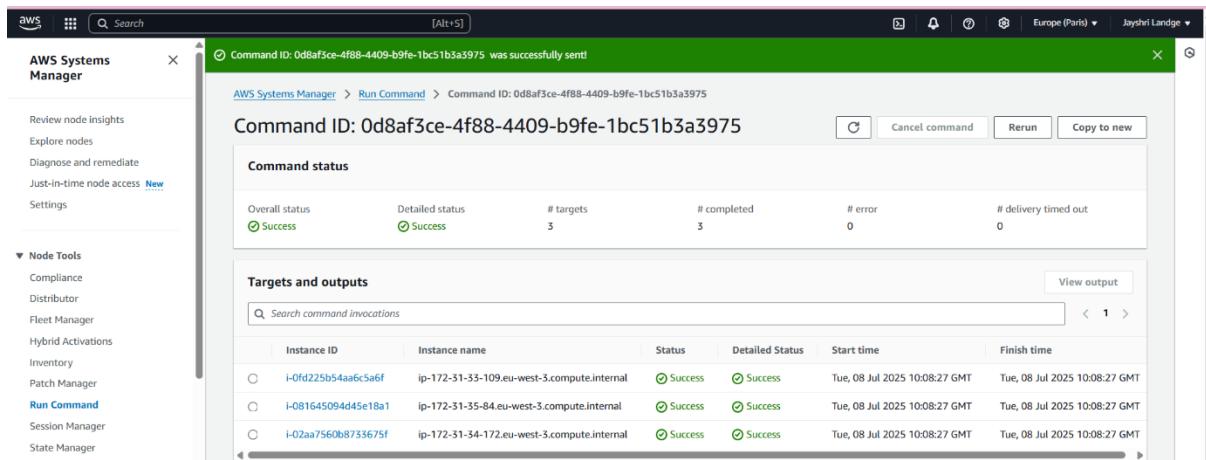


Screenshot of Session Manager connection

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.