Connecting to a Private EC2 Instance Using AWS Session Manager

Create an IAM Role with AmazonSSMManagedInstanceCore

Before connecting to a private EC2 instance via Session Manager, we need to ensure the instance has the necessary permissions to use the AWS Systems Manager (SSM) service. This is achieved by creating an IAM role with the AmazonSSMManagedInstanceCore policy.

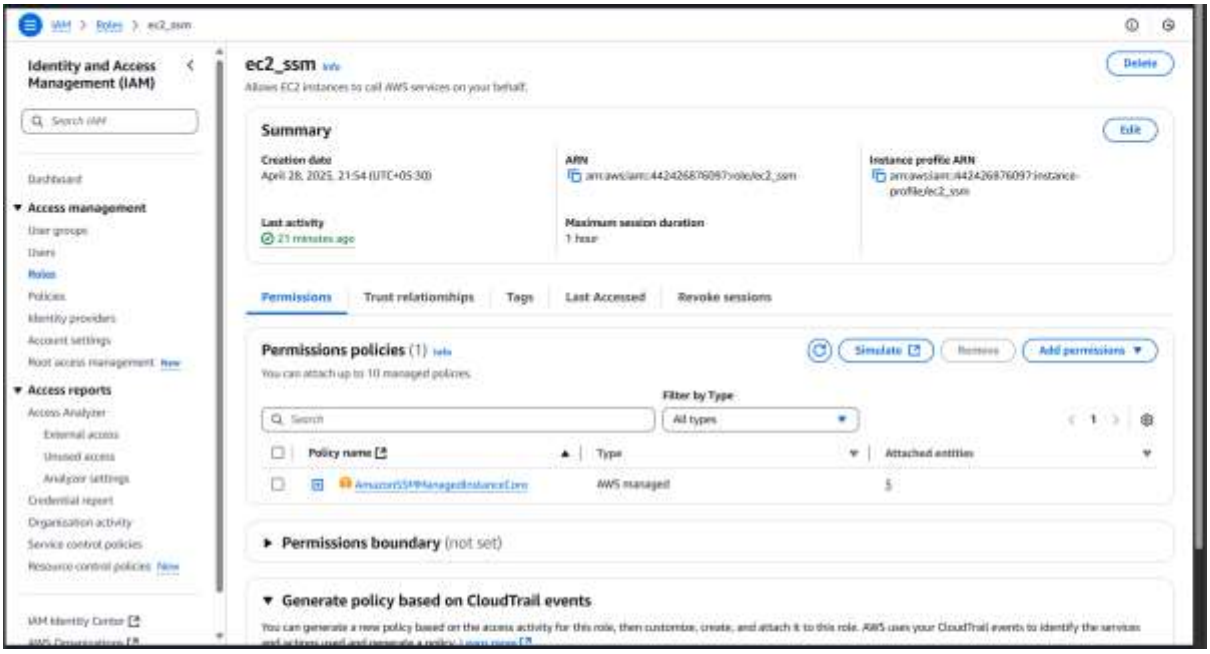
Steps:

Open the IAM Console:

- Navigate to the IAM Console

Create IAM Role:

- Click on **Roles** in the left navigation pane and then click **Create Role**.
- Select **EC2** as the trusted entity.
- Attach the **AmazonSSMManagedInstanceCore** policy to the role.
- Name the role (e.g., ec2_ssm) for easy identification.
- Click **Create Role** to finalize the process.



ec2_ssm Info

Allows EC2 instances to call AWS services on your behalf.

Summary

Creation date: April 28, 2025, 21:54 (UTC+05:30)

ARN: arn:aws:iam::442426876097:role/ec2_ssm

Instance profile ARN: arn:aws:iam::442426876097:instance-profile/ec2_ssm

Last activity: 21 minutes ago

Maximum session duration: 1 hour

Permissions policies (1) Info

You can attach up to 10 managed policies.

Filter by Type: All types

Policy name	Type	Attached entities
AmazonSSMManagedInstanceCore	AWS managed	5

Permissions boundary (not set)

Generate policy based on CloudTrail events

You can generate a new policy based on the access activity for this role, then customize, create, and attach it to this role. AWS uses your CloudTrail events to identify the services and actions used and generate a policy. Learn more

Launch a Public EC2 Instance

You will launch a public EC2 instance, configure the necessary security settings, and associate the previously created IAM role to this instance. This EC2 instance will act as the "jump box" or intermediary, from which you can initiate connections to the private instance.

Steps:

Navigate to EC2 Console:

- Open the EC2 Console.

Launch a New Instance:

- Click **Launch Instance**.
- Choose the **t2.micro** instance type for a low-cost, basic instance suitable for this purpose.

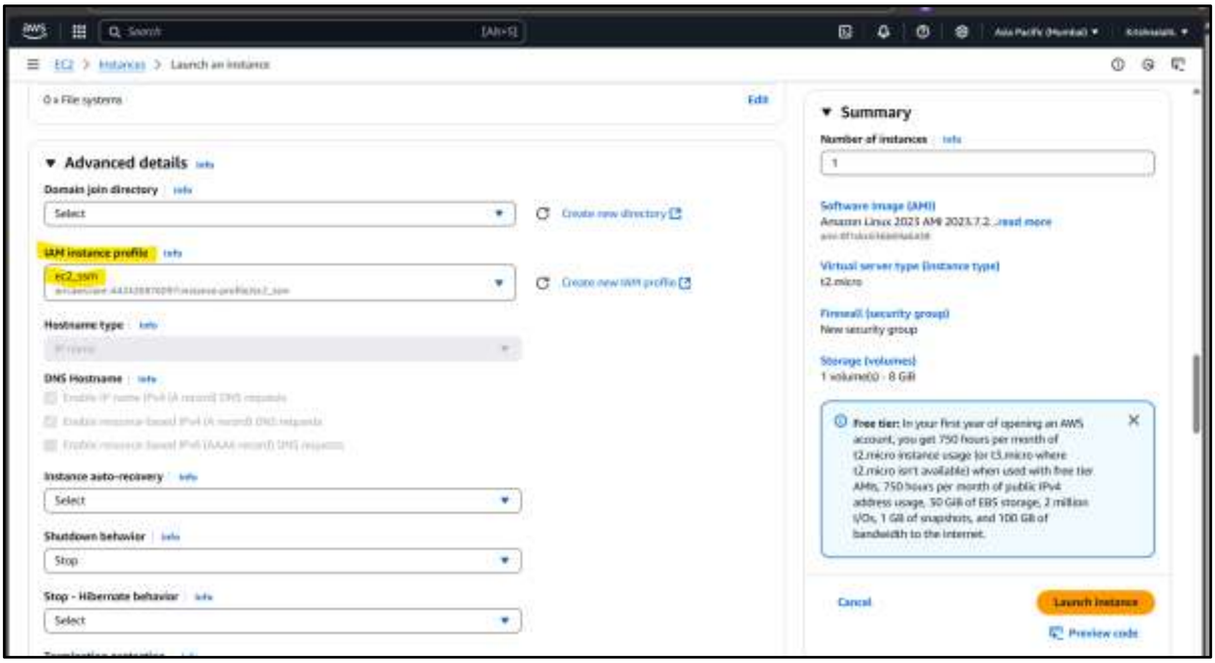
Configure Security Group:

- Create a security group that allows inbound traffic on the following ports:
  - **SSH** (Port 22) for command-line access.
  - **HTTP** (Port 80) for web access.
  - **HTTPS** (Port 443) for secure web access.

Assign IAM Role:

- In the **Advanced Details** section, select the **ec2_ssm** IAM role you created earlier.

s


EC2 > Instances > Launch an instance

0 x File systems

Advanced details

Domain join directory: Select

IAM instance profile: ec2_ssm

Hostname type: IP name

DNS Hostname

- ☐ Enable IP name IPv4 (A record) DNS requests
- ☑ Enable resource-based IPv4 (A record) DNS requests
- ☐ Enable resource-based IPv6 (AAAA record) DNS requests

Instance auto-recovery: Select

Shutdown behavior: Stop

Stop - Hibernate behavior: Select

Summary

Number of instances: 1

Software Image (AMI): Amazon Linux 2023 AMI 2023.7.2...read more

Virtual server type (instance type): t2.micro

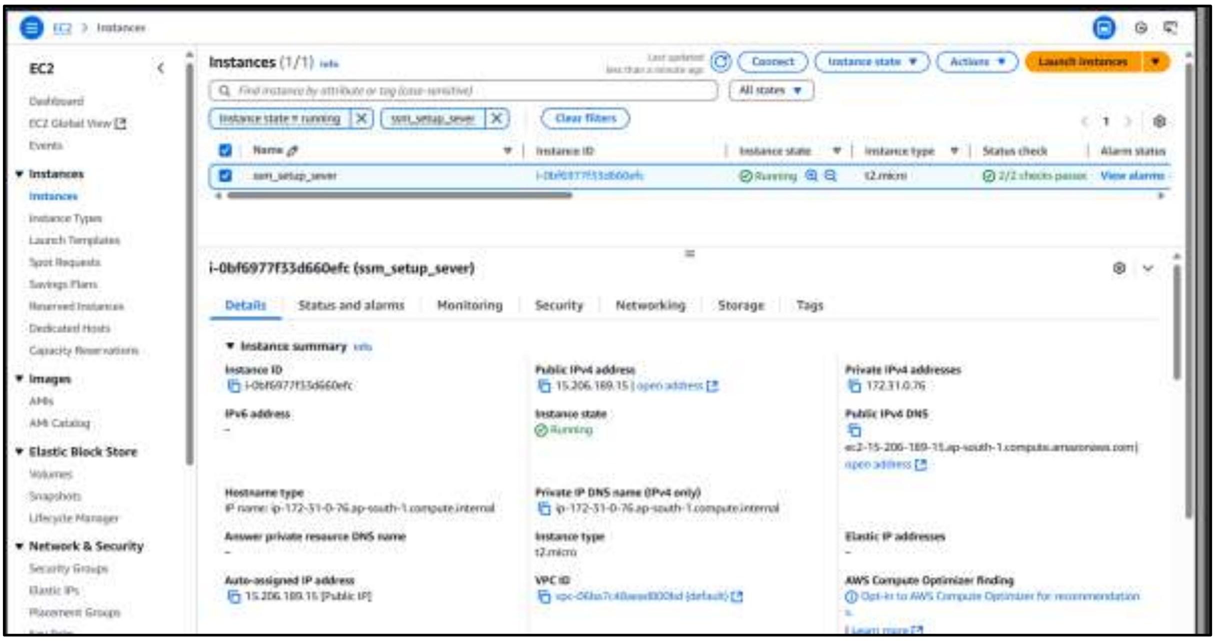
Firewall (security group): New security group

Storage (volumes): 1 volume(s) - 8 GiB

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Launch the Instance:

- After completing the configuration, click **Launch Instance** to initiate the creation of the public EC2 instance.


EC2 > Instances

Instances (1/1)

Filters: Instance state = running; ssm_setup_sever

Name	Instance ID	Instance state	Instance type	Status check	Alarm status
ssm_setup_sever	i-0bf6977f33d660efc	Running	t2.micro	2/2 checks passed	View alarms

i-0bf6977f33d660efc (ssm_setup_sever)

Instance summary

Instance ID: i-0bf6977f33d660efc

Public IPv4 address: 15.206.189.15 | open address

Private IPv4 addresses: 172.31.0.76

IPv6 address: –

Instance state: Running

Public IPv4 DNS: ec2-15-206-189-15.ap-south-1.compute.amazonaws.com | open address

Hostname type: IP name: ip-172-31-0-76.ap-south-1.compute.internal

Private IP DNS name (IPv4 only): ip-172-31-0-76.ap-south-1.compute.internal

Answer private resource DNS name: –

Instance type: t2.micro

Elastic IP addresses: –

Auto-assigned IP address: 15.206.189.15 [Public IP]

VPC ID: vpc-06ba7c40aeed8006d (default)

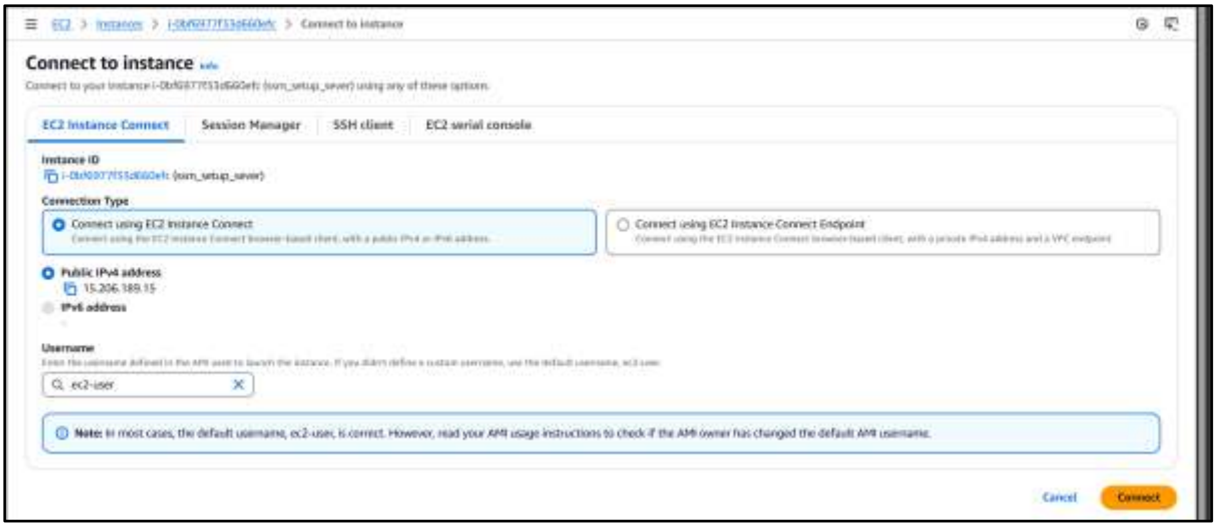
AWS Compute Optimizer finding: Opt-in to AWS Compute Optimizer for recommendations. | Learn more

Connect to the Public EC2 Instance and Install the SSM Agent

The public EC2 instance now needs to be configured with the AWS Systems Manager (SSM) agent to allow it to communicate with the AWS SSM service.

Steps:

Connect to EC2 Instance: using Ec2 Instance Connect.



EC2 > Instances > i-0bf0877f53d660efc > Connect to instance

Connect to instance Info

Connect to your instance i-0bf0877f53d660efc (ssm_setup_sever) using any of these options.

EC2 Instance Connect | Session Manager | SSH client | EC2 serial console

Instance ID
i-0bf0877f53d660efc (ssm_setup_sever)

Connection Type

● Connect using EC2 Instance Connect
Connect using the EC2 Instance Connect browser-based client, with a public IPv4 or IPv6 address.

○ Connect using EC2 Instance Connect Endpoint
Connect using the EC2 Instance Connect browser-based client, with a private IPv4 address and a VPC endpoint.

● Public IPv4 address
15.206.189.15

○ IPv6 address

Username
Enter the username defined in the AMI used to launch the instance. If you didn't define a custom username, use the default username, ec2-user.

ec2-user

Note: In most cases, the default username, ec2-user, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

Switch to the Root User:

- Once logged in, switch to the root user by running:

```
sudo su -
```

Install the SSM Agent:

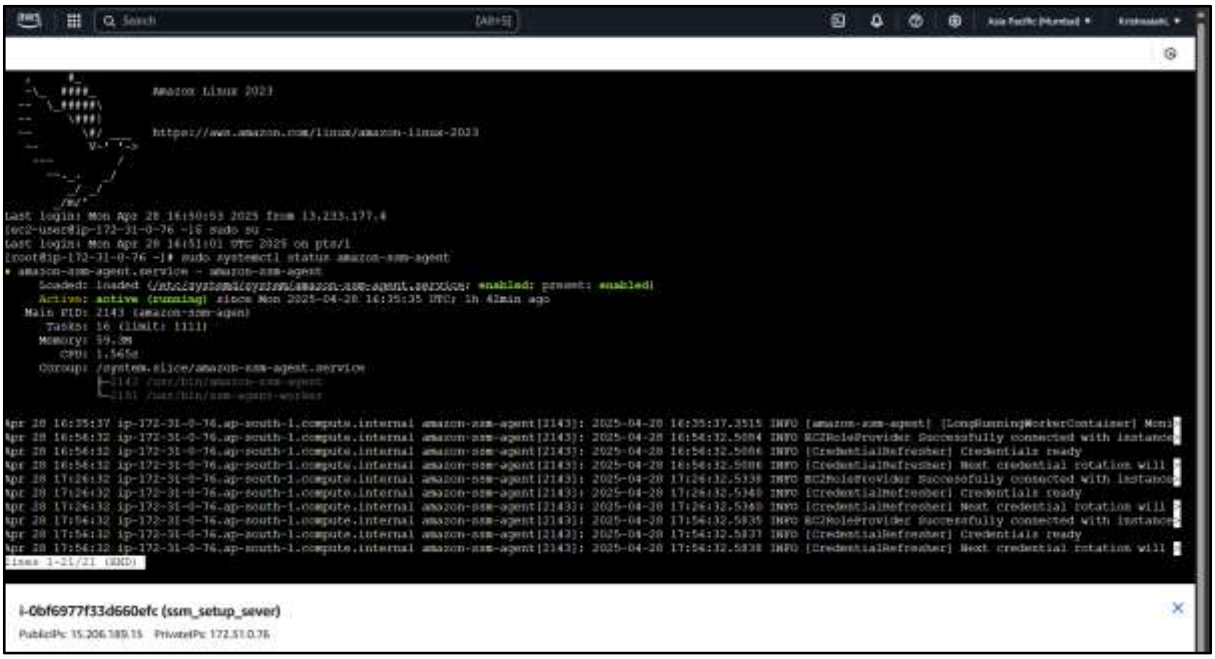
Run the following commands to update the instance and install the SSM agent:

```
sudo dnf update -y
sudo dnf install amazon-ssm-agent -y
sudo systemctl start amazon-ssm-agent
sudo systemctl enable amazon-ssm-agent
```

Verify the SSM Agent is Running:

Ensure the agent is running correctly by executing:

```
sudo systemctl status amazon-ssm-agent
```



```
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Mon Apr 28 16:50:53 2025 from 13.233.177.4
[ec2-user@ip-172-31-0-76 ~]$ sudo su -
Last login: Mon Apr 28 16:51:01 UTC 2025 on pts/1
[root@ip-172-31-0-76 ~]# sudo systemctl status amazon-ssm-agent
● amazon-ssm-agent.service - amazon-ssm-agent
     Loaded: loaded (/etc/systemd/system/amazon-ssm-agent.service; enabled; preset: enabled)
     Active: active (running) since Mon 2025-04-28 16:35:35 UTC; 1h 42min ago
   Main PID: 2143 (amazon-ssm-agen)
      Tasks: 16 (limit: 1111)
     Memory: 59.3M
        CPU: 1.565s
     CGroup: /system.slice/amazon-ssm-agent.service
             ├─2143 /usr/bin/amazon-ssm-agent
             └─2181 /usr/bin/ssm-agent-worker

Apr 28 16:35:37 ip-172-31-0-76.ap-south-1.compute.internal amazon-ssm-agent[2143]: 2025-04-28 16:35:37.3515 INFO [amazon-ssm-agent] [LongRunningWorkerContainer] Moni>
Apr 28 16:56:32 ip-172-31-0-76.ap-south-1.compute.internal amazon-ssm-agent[2143]: 2025-04-28 16:56:32.5084 INFO EC2RoleProvider Successfully connected with instance>
Apr 28 16:56:32 ip-172-31-0-76.ap-south-1.compute.internal amazon-ssm-agent[2143]: 2025-04-28 16:56:32.5086 INFO [CredentialRefresher] Credentials ready
Apr 28 16:56:32 ip-172-31-0-76.ap-south-1.compute.internal amazon-ssm-agent[2143]: 2025-04-28 16:56:32.5086 INFO [CredentialRefresher] Next credential rotation will >
Apr 28 17:26:32 ip-172-31-0-76.ap-south-1.compute.internal amazon-ssm-agent[2143]: 2025-04-28 17:26:32.5338 INFO EC2RoleProvider Successfully connected with instance>
Apr 28 17:26:32 ip-172-31-0-76.ap-south-1.compute.internal amazon-ssm-agent[2143]: 2025-04-28 17:26:32.5340 INFO [CredentialRefresher] Credentials ready
Apr 28 17:26:32 ip-172-31-0-76.ap-south-1.compute.internal amazon-ssm-agent[2143]: 2025-04-28 17:26:32.5340 INFO [CredentialRefresher] Next credential rotation will >
Apr 28 17:56:32 ip-172-31-0-76.ap-south-1.compute.internal amazon-ssm-agent[2143]: 2025-04-28 17:56:32.5835 INFO EC2RoleProvider Successfully connected with instance>
Apr 28 17:56:32 ip-172-31-0-76.ap-south-1.compute.internal amazon-ssm-agent[2143]: 2025-04-28 17:56:32.5837 INFO [CredentialRefresher] Credentials ready
Apr 28 17:56:32 ip-172-31-0-76.ap-south-1.compute.internal amazon-ssm-agent[2143]: 2025-04-28 17:56:32.5838 INFO [CredentialRefresher] Next credential rotation will >
lines 1-21/21 (END)
```

i-0bf6977f33d660efc (ssm_setup_sever)

PublicIPs: 15.206.189.15 PrivateIPs: 172.31.0.76

This installation ensures the instance can be managed and connected to using AWS Systems Manager.

Create an Amazon Machine Image (AMI)

Next, we will create an AMI from the public EC2 instance that includes the SSM agent, which can then be used to launch the private EC2 instance.

Steps:

Go to the EC2 Console:

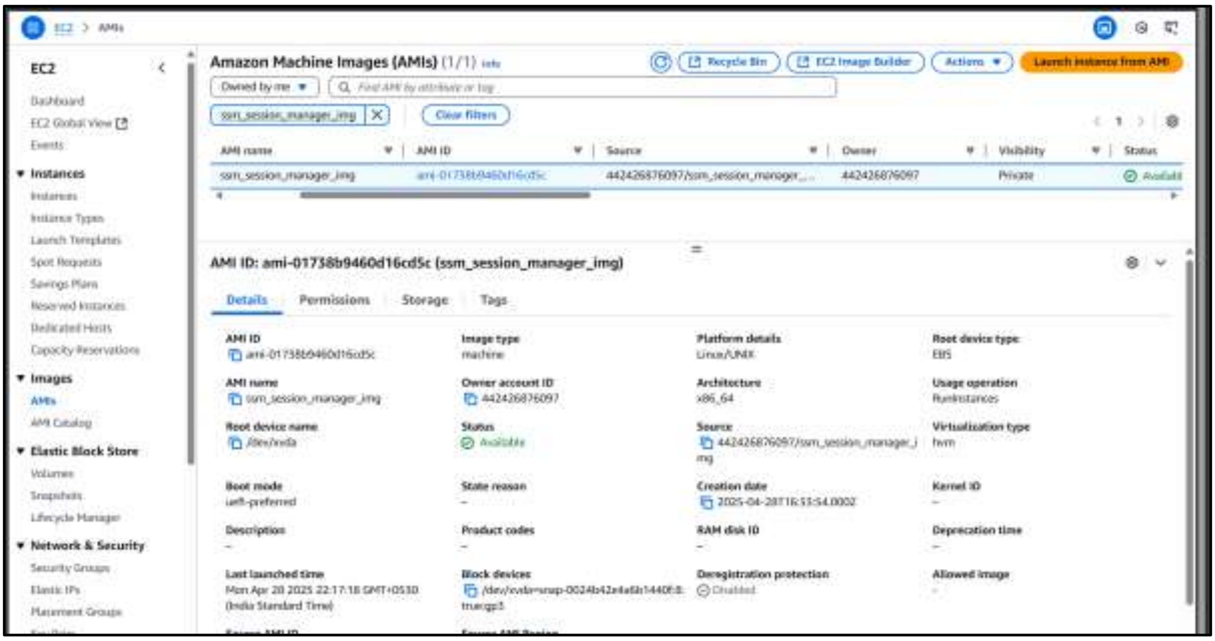
- Navigate to the EC2 Console.

Create an Image:

- Select the EC2 instance you just launched.
- From the **Actions** dropdown, choose **Image and templates** > **Create Image**.
- Provide a descriptive name for the image (e.g., ssm_session_manager_img).
- Click **Create Image**.

Verify AMI Status:

- Ensure that the AMI status is marked as **Available**. It may take a few minutes for the AMI to become available.



Amazon Machine Images (AMIs) (1/1) Info

ssm_session_manager_img

AMI name	AMI ID	Source	Owner	Visibility	Status
ssm_session_manager_img	ami-01738b9460d16cd5c	442426876097/ssm_session_manager_...	442426876097	Private	Availabl

AMI ID: ami-01738b9460d16cd5c (ssm_session_manager_img)

AMI ID ami-01738b9460d16cd5c	Image type machine	Platform details Linux/UNIX	Root device type EBS
AMI name ssm_session_manager_img	Owner account ID 442426876097	Architecture x86_64	Usage operation RunInstances
Root device name /dev/xvda	Status Available	Source 442426876097/ssm_session_manager_img	Virtualization type hvm
Boot mode uefi-preferred	State reason –	Creation date 2025-04-20T16:53:54.000Z	Kernel ID –
Description –	Product codes –	RAM disk ID –	Deprecation time –
Last launched time Mon Apr 20 2025 22:17:18 GMT+0530 (India Standard Time)	Block devices /dev/xvda=snap-0024b42e4a6b1440f:8:true:gp3	Deregistration protection Disabled	Allowed image –
Source AMI ID	Source AMI Region		

Launch the Private EC2 Instance

Now, you will use the custom AMI to launch a private EC2 instance in a private subnet, ensuring it is not directly accessible from the internet.

Steps:

Navigate to AMIs:

- Open the EC2 Console and go to the **AMIs** section.

Launch Instance from AMI:

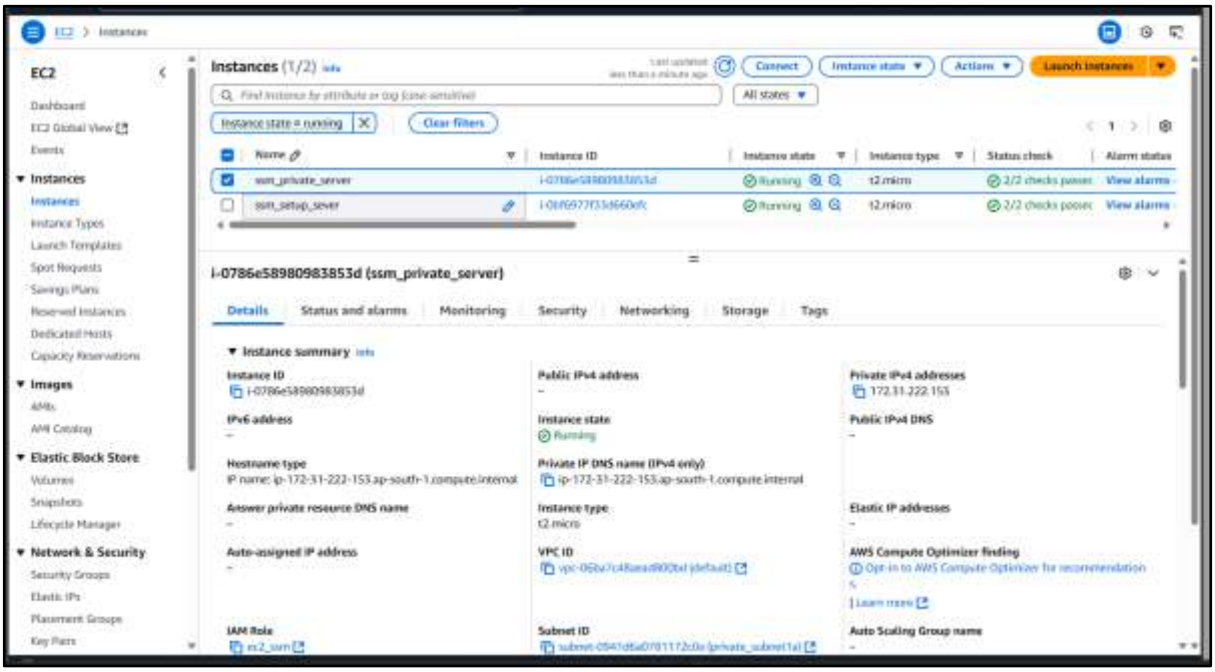
- Select the custom AMI you created (e.g., ssm_session_manager_img) and click **Launch instance from AMI**.

Configure the Instance:

- Choose the **t2.micro** instance type.
- In the **IAM instance profile** section, select the **ec2_ssm** role.
- Place the instance in a **private subnet** (ensure that the instance does not have a public IP).

Complete the Launch:

- Complete the rest of the configuration and click **Launch Instance**.



Instances (1/2) Info

Last updated less than a minute ago

Find Instance by attribute or tag (case-sensitive)

Instance state = running

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status
☑	ssm_private_server	i-0786e58980983853d	Running	t2.micro	2/2 checks passed	View alarms
□	ssm_setup_sever	i-0bf6977f33d660dfc	Running	t2.micro	2/2 checks passed	View alarms

i-0786e58980983853d (ssm_private_server)

Instance summary Info

Instance ID i-0786e58980983853d	Public IPv4 address –	Private IPv4 addresses 172.31.222.153
IPv6 address –	Instance state Running	Public IPv4 DNS –
Hostname type IP name: ip-172-31-222-153.ap-south-1.compute.internal	Private IP DNS name (IPv4 only) ip-172-31-222-153.ap-south-1.compute.internal	
Answer private resource DNS name –	Instance type t2.micro	Elastic IP addresses –
Auto-assigned IP address –	VPC ID vpc-06ba7c48aead800bd (default)	AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. Learn more
IAM Role ec2_ssm	Subnet ID subnet-0541d6a0781172c0a (private_subnet1a)	Auto Scaling Group name –

Create VPC Endpoints

For the private instance to communicate with AWS Systems Manager, we need to create VPC endpoints for the SSM service. These endpoints will enable private connectivity to SSM without using the internet.

Steps:

Go to the VPC Console:

- Open the VPC Console.

Create Interface Endpoints:

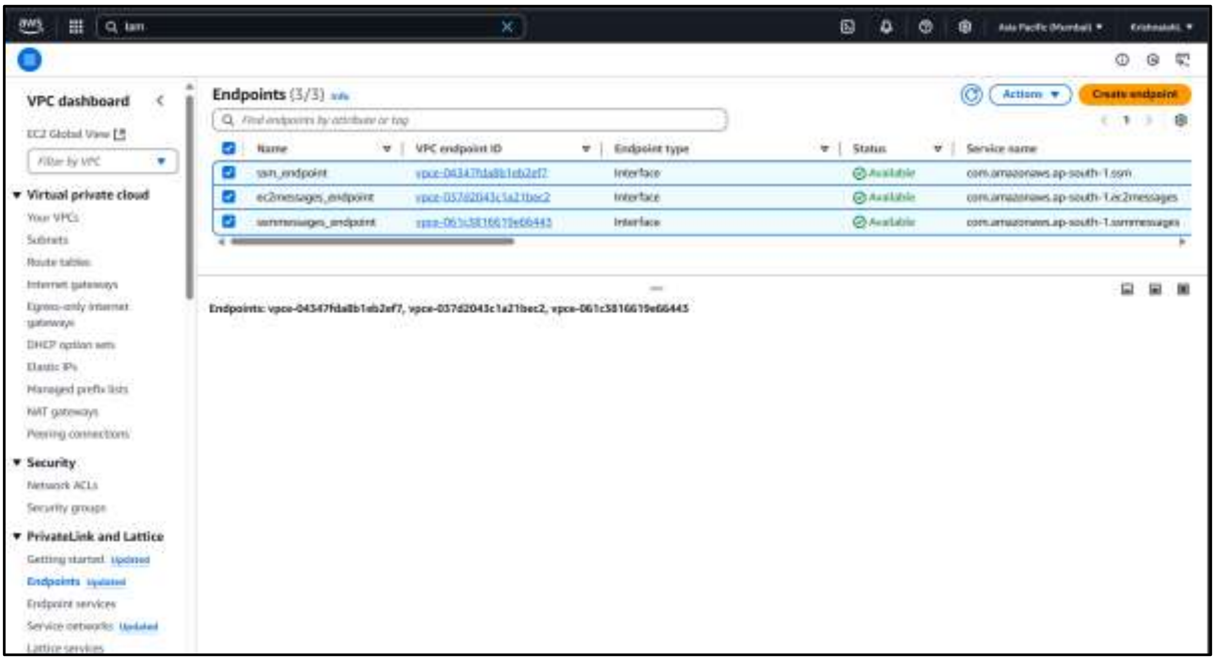
- Click **Endpoints** > **Create Endpoint**.
- Create the following three Interface Endpoints:
  - com.amazonaws.<region>.ssm
  - com.amazonaws.<region>.ssmmessages
  - com.amazonaws.<region>.ec2messages

Configure Each Endpoint:

- For each endpoint, select the **VPC** where the EC2 instance resides.
- Select the private subnets where the EC2 instance is running.
- Attach a **security group** to the endpoint.

Create Endpoints:

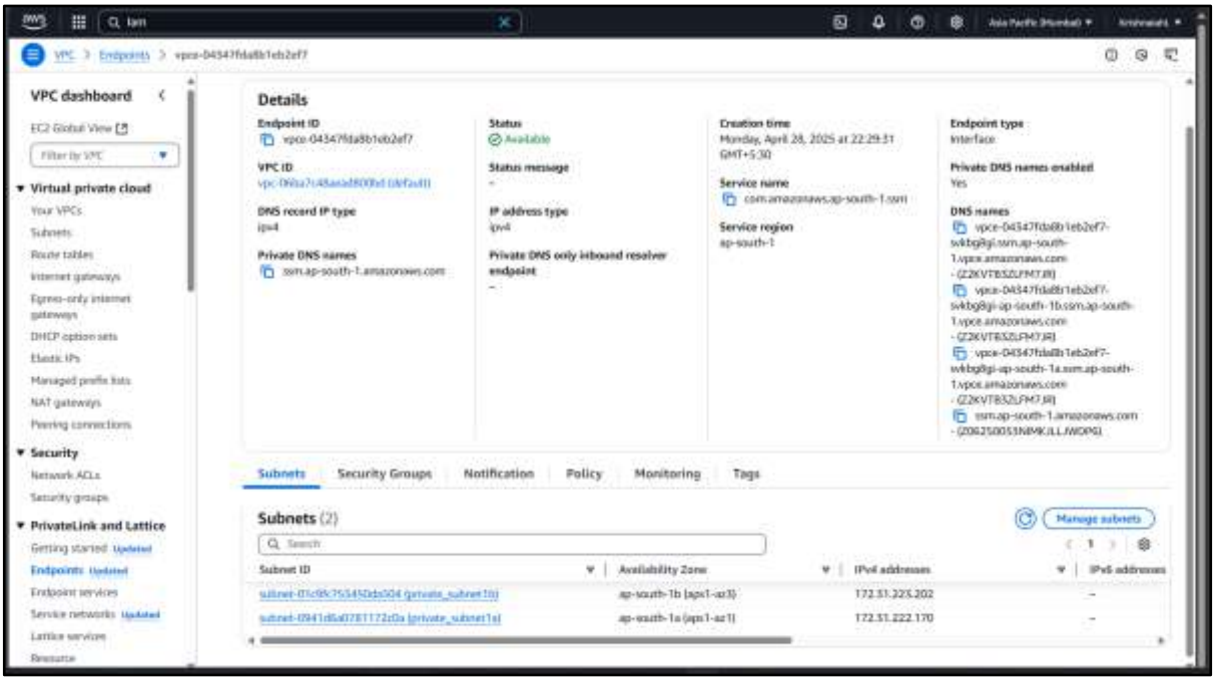
- Click **Create** to create each endpoint.



Endpoints (3/3)

☑	Name	VPC endpoint ID	Endpoint type	Status	Service name
☑	ssm_endpoint	vpce-04347fda8b1eb2ef7	Interface	Available	com.amazonaws.ap-south-1.ssm
☑	ec2messages_endpoint	vpce-037d2043c1a21bec2	Interface	Available	com.amazonaws.ap-south-1.ec2messages
☑	ssmmessages_endpoint	vpce-061c3816619e66443	Interface	Available	com.amazonaws.ap-south-1.ssmmessages

Endpoints: vpce-04347fda8b1eb2ef7, vpce-037d2043c1a21bec2, vpce-061c3816619e66443



VPC > Endpoints > vpce-04347fda8b1eb2ef7

Details

Endpoint ID
vpce-04347fda8b1eb2ef7

VPC ID
vpc-0f6a7c48aad800bd (default)

DNS record IP type
ipv4

Private DNS names
ssm.ap-south-1.amazonaws.com

Status
Available

Status message
–

IP address type
ipv4

Private DNS only inbound resolver endpoint
–

Creation time
Monday, April 28, 2025 at 22:29:31 GMT+5:30

Service name
com.amazonaws.ap-south-1.ssm

Service region
ap-south-1

Endpoint type
Interface

Private DNS names enabled
Yes

DNS names
vpce-04347fda8b1eb2ef7-wkbg8gi.ssm.ap-south-1.vpce.amazonaws.com - (Z2KVTB32LFM7JR)
vpce-04347fda8b1eb2ef7-wkbg8gi-ap-south-1b.ssm.ap-south-1.vpce.amazonaws.com - (Z2KVTB32LFM7JR)
vpce-04347fda8b1eb2ef7-wkbg8gi-ap-south-1a.ssm.ap-south-1.vpce.amazonaws.com - (Z2KVTB32LFM7JR)
ssm.ap-south-1.amazonaws.com - (Z06250053NMKJLLWOPG)

Subnets (2)

Subnet ID	Availability Zone	IPv4 addresses	IPv6 addresses
subnet-01c9fc755450ds504 (private_subnet1b)	ap-south-1b (aps1-az3)	172.31.223.202	–
subnet-0941d6a0781172d0a (private_subnet1a)	ap-south-1a (aps1-az1)	172.31.222.170	–

Connect to the Private EC2 Instance

After setting up everything, you can connect to your private EC2 instance using **AWS Session Manager**.

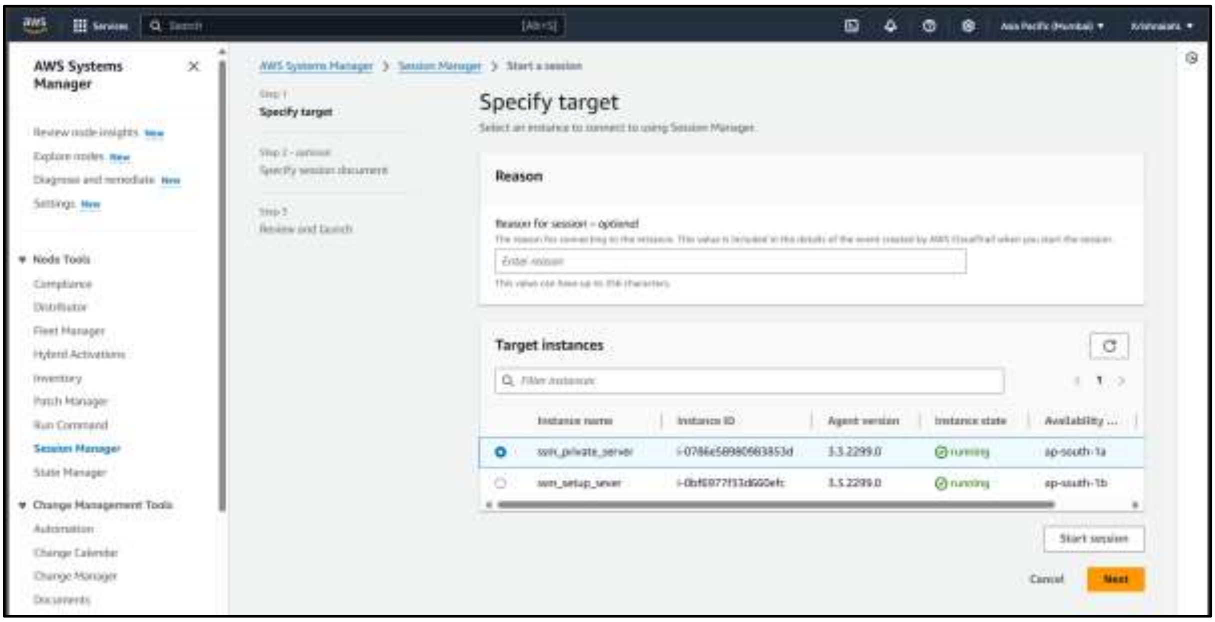
Via Session Manager:

Open Systems Manager Console:

- Go to the Systems Manager Console.

Start a Session:

- Click **Session Manager** > **Start Session**.
- Select your private EC2 instance and click **Start Session**.



Specify target

Select an instance to connect to using Session Manager.

Reason

Reason for session - optional

The reason for connecting to the instance. This value is included in the details of the event created by AWS CloudTrail when you start the session.

Enter reason

This value can have up to 256 characters.

Target instances

Filter instances

	Instance name	Instance ID	Agent version	Instance state	Availability ...
●	ssm_private_server	i-0786e58980983853d	3.3.2299.0	running	ap-south-1a
○	ssm_setup_server	i-0bf6977f53d660efc	3.3.2299.0	running	ap-south-1b

Via EC2 Instance Connect:

Go to EC2 Console:

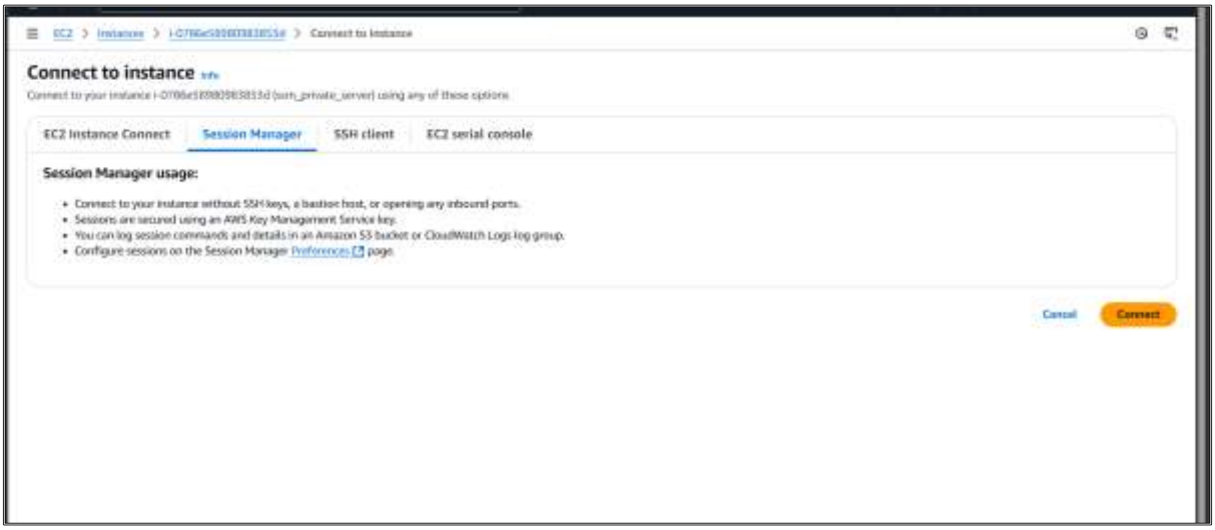
- Open the EC2 Console and go to **Instances**.

Select Your Instance:

- Choose the private EC2 instance you want to connect to.

Connect via Session Manager:

- Click **Connect** and choose **Session Manager**.
- Click **Connect** again to initiate the connection.



Connect to instance

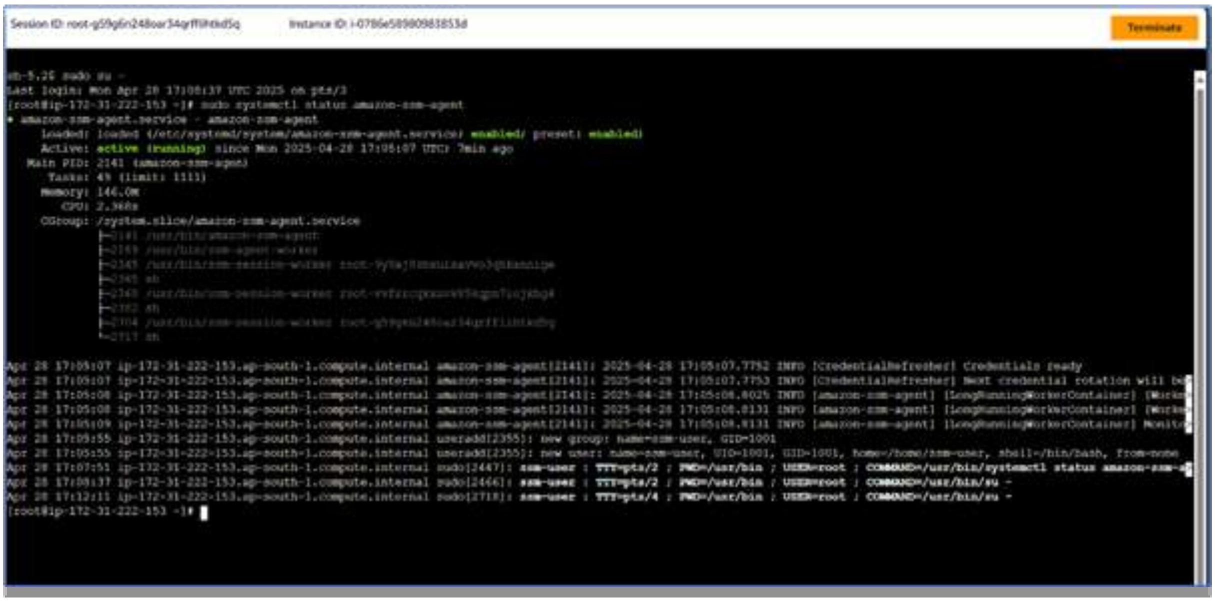
Connect to your instance i-0786e58980983853d (ssm_private_server) using any of these options

Session Manager usage:

- Connect to your instance without SSH keys, a bastion host, or opening any inbound ports.
- Sessions are secured using an AWS Key Management Service key.
- You can log session commands and details in an Amazon S3 bucket or CloudWatch Logs log group.
- Configure sessions on the Session Manager Preferences page.

After connecting verify the SSM Agent service using:

sudo systemctl status amazon-ssm-agent



Session ID: root-g59g6n248oar34qrff9htkd5q
Instance ID: i-0786e58980983853d
Terminate

```
sh-5.2$ sudo su -
Last login: Mon Apr 28 17:08:37 UTC 2025 on pts/3
[root@ip-172-31-222-153 ~]# sudo systemctl status amazon-ssm-agent
● amazon-ssm-agent.service - amazon-ssm-agent
     Loaded: loaded (/etc/systemd/system/amazon-ssm-agent.service; enabled; preset: enabled)
     Active: active (running) since Mon 2025-04-28 17:05:07 UTC; 7min ago
   Main PID: 2141 (amazon-ssm-agent)
      Tasks: 49 (limit: 1111)
     Memory: 146.0M
        CPU: 2.368s
     CGroup: /system.slice/amazon-ssm-agent.service
             ├─2141 /usr/bin/amazon-ssm-agent
             ├─2169 /usr/bin/ssm-agent-worker
             ├─2345 /usr/bin/ssm-session-worker root-9yfaj8mmuisavvo3q8lannige
             ├─2365 sh
             ├─2368 /usr/bin/ssm-session-worker root-vvfxrcppxxxv95xgpn7iojxbg4
             ├─2382 sh
             ├─2704 /usr/bin/ssm-session-worker root-g59g6n248oar34qrff9htkd5q
             └─2717 sh

Apr 28 17:05:07 ip-172-31-222-153.ap-south-1.compute.internal amazon-ssm-agent[2141]: 2025-04-28 17:05:07.7752 INFO [CredentialRefresher] Credentials ready
Apr 28 17:05:07 ip-172-31-222-153.ap-south-1.compute.internal amazon-ssm-agent[2141]: 2025-04-28 17:05:07.7753 INFO [CredentialRefresher] Next credential rotation will be
Apr 28 17:05:08 ip-172-31-222-153.ap-south-1.compute.internal amazon-ssm-agent[2141]: 2025-04-28 17:05:08.8025 INFO [amazon-ssm-agent] [LongRunningWorkerContainer] [Worke
Apr 28 17:05:08 ip-172-31-222-153.ap-south-1.compute.internal amazon-ssm-agent[2141]: 2025-04-28 17:05:08.8131 INFO [amazon-ssm-agent] [LongRunningWorkerContainer] [Worke
Apr 28 17:05:09 ip-172-31-222-153.ap-south-1.compute.internal amazon-ssm-agent[2141]: 2025-04-28 17:05:08.8131 INFO [amazon-ssm-agent] [LongRunningWorkerContainer] Monito
Apr 28 17:05:55 ip-172-31-222-153.ap-south-1.compute.internal useradd[2355]: new group: name=ssm-user, GID=1001
Apr 28 17:05:55 ip-172-31-222-153.ap-south-1.compute.internal useradd[2355]: new user: name=ssm-user, UID=1001, GID=1001, home=/home/ssm-user, shell=/bin/bash, from=none
Apr 28 17:07:51 ip-172-31-222-153.ap-south-1.compute.internal sudo[2447]: ssm-user : TTY=pts/2 ; PWD=/usr/bin ; USER=root ; COMMAND=/usr/bin/systemctl status amazon-ssm-a
Apr 28 17:08:37 ip-172-31-222-153.ap-south-1.compute.internal sudo[2466]: ssm-user : TTY=pts/2 ; PWD=/usr/bin ; USER=root ; COMMAND=/usr/bin/su -
Apr 28 17:12:11 ip-172-31-222-153.ap-south-1.compute.internal sudo[2718]: ssm-user : TTY=pts/4 ; PWD=/usr/bin ; USER=root ; COMMAND=/usr/bin/su -
[root@ip-172-31-222-153 ~]# 
```

Conclusion

By following these steps, you can successfully connect to a private EC2 instance using AWS Session Manager, even when the instance has no public IP. This method ensures that your private instances remain secure while still being accessible for administrative tasks.