aws

Hands-on tutorials

Remotely Run Commands on an EC2 Instance with AWS Systems Manager

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Remotely Run Commands on an EC2 Instance with AWS Systems Manager: Hands-on tutorials

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

Remotely Run Commands on an EC2 Instance with AWS Systems Manager

AWS experience	Beginner
Time to complete	10 minutes
Cost to complete	Free Tier eligible
Last updated	July 14, 2022

Overview

In this hands-on tutorial, you will learn how to use AWS Systems Manager to remotely run commands on your Amazon EC2 instances. Systems Manager is a management tool that enables you to gain operational insights and take action on AWS resources safely and at scale. Using the run command, one of the automation features of Systems Manager, you can simplify management tasks by eliminating the need to use bastion hosts, SSH, or remote PowerShell.

In our example scenario, as a System Administrator, you need to update the packages on your EC2 instances. To complicate this normally simple admin task, your security team does not allow you to direct access production servers via SSH or allow you to use bastion hosts. Fortunately, you can use Systems Manager to remotely run commands, like update packages, on your EC2 instances.

To solve this challenging scenario, you will create an Identity and Access Management (IAM) role, enable an agent on your instance that communicates with Systems Manager, then follow best practices by running the AWS-UpdateSSMAgent document to upgrade your Systems Manager Agent, and finally use Systems Manager to run a command on your instance.

AWS Systems Manager is an always free tier product. The EC2 instance you create in this tutorial is free tier eligible.

Open the AWS Management Console, so you can keep this step-by-step guide open. When the screen loads, enter your user name and password to get started.

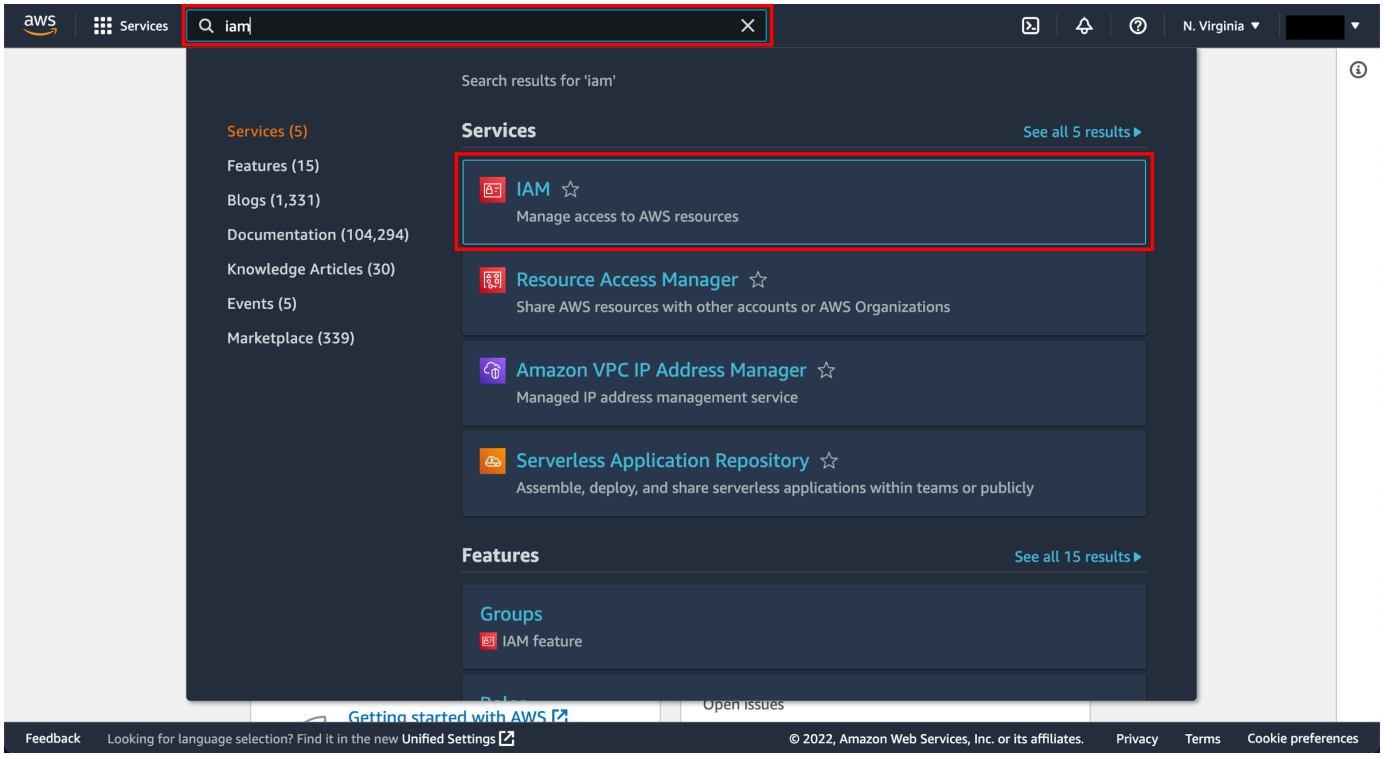
Implementation

Step 1: Create an Identity and Access Management (IAM) role

In this step, you will create an EC2 instance using the EnablesEC2ToAccessSystemsManagerRole role. This will allow the EC2 instance to be managed by Systems Manager.

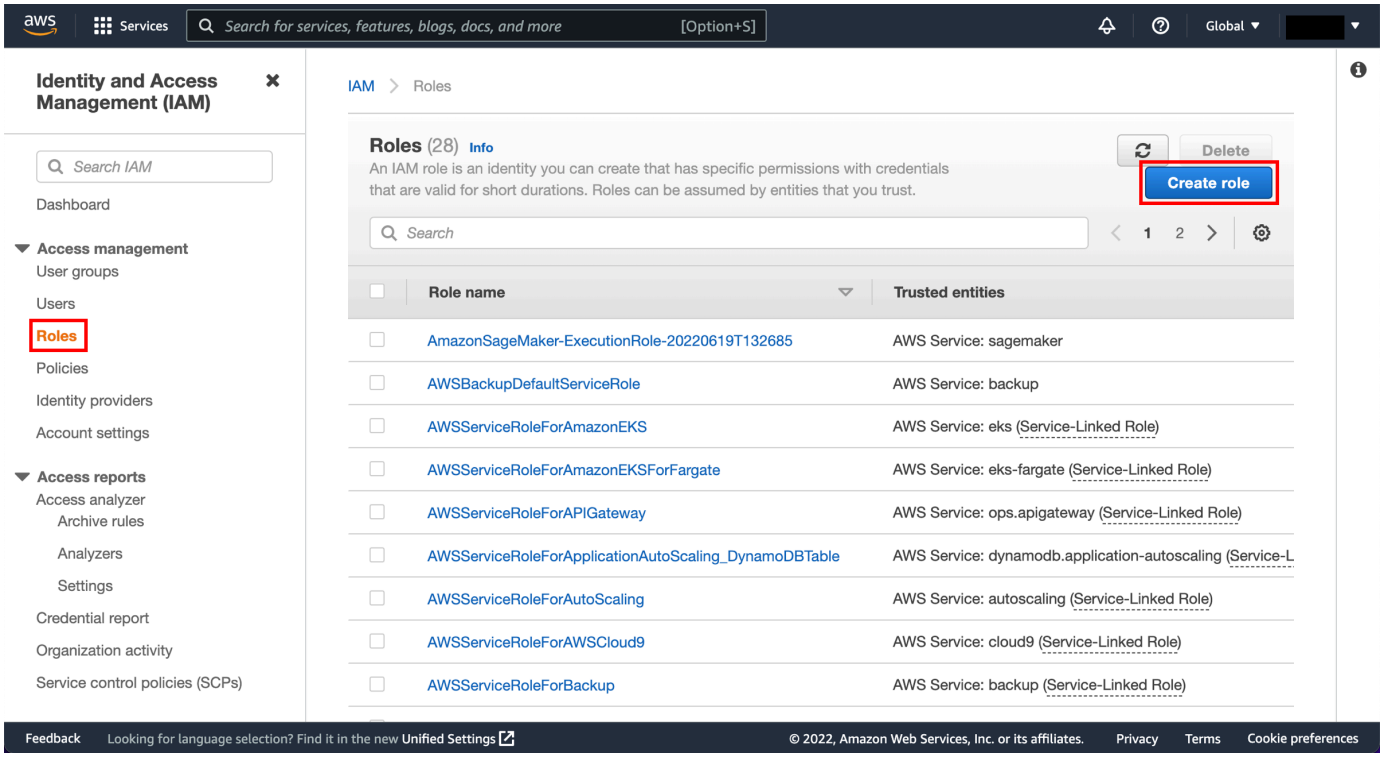
1. Open the IAM console

   Open the IAM console at https://console.aws.amazon.com/iam/.



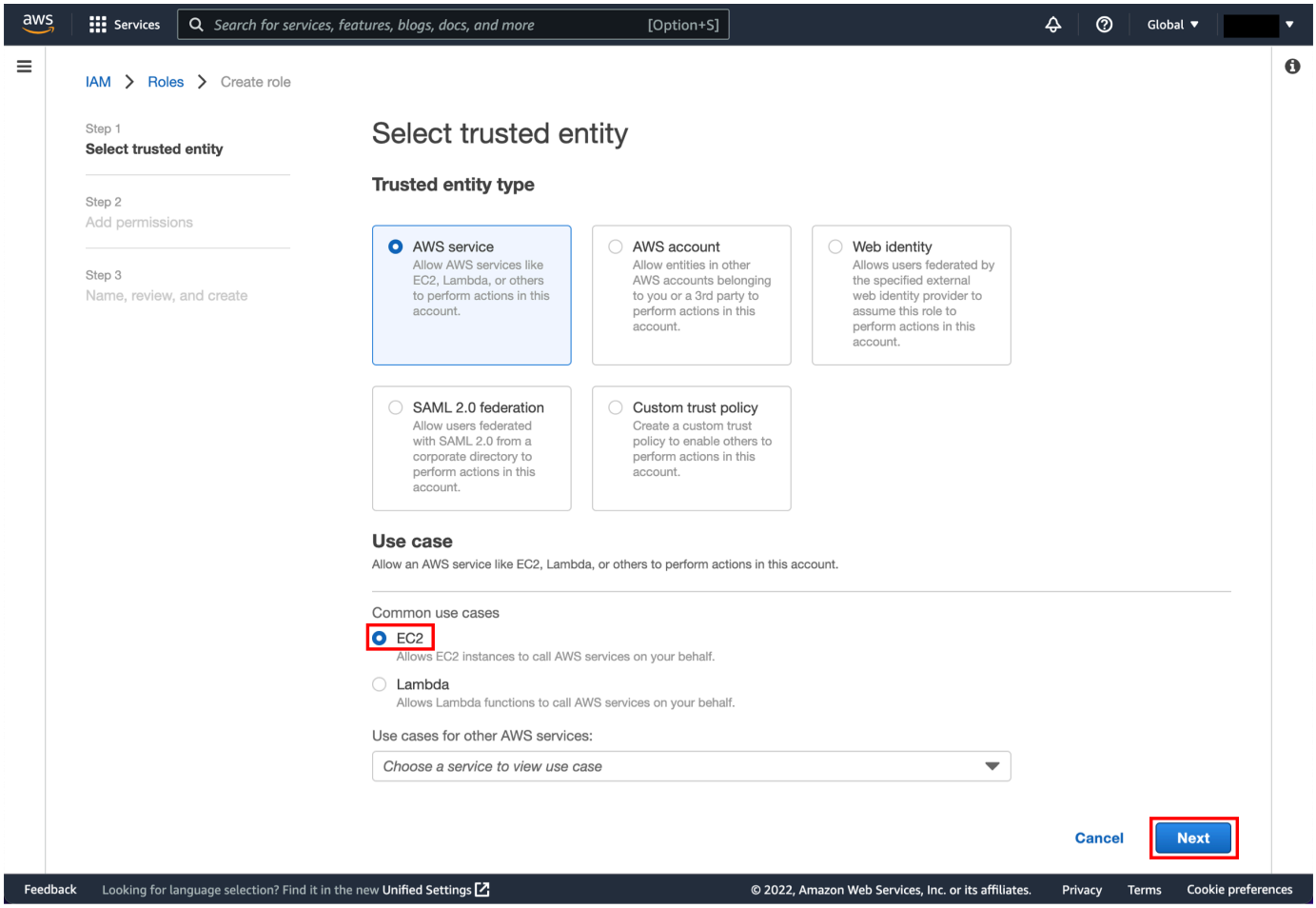
2. Create the role

   In the left navigation pane, choose **Roles**, and then choose **Create role**.



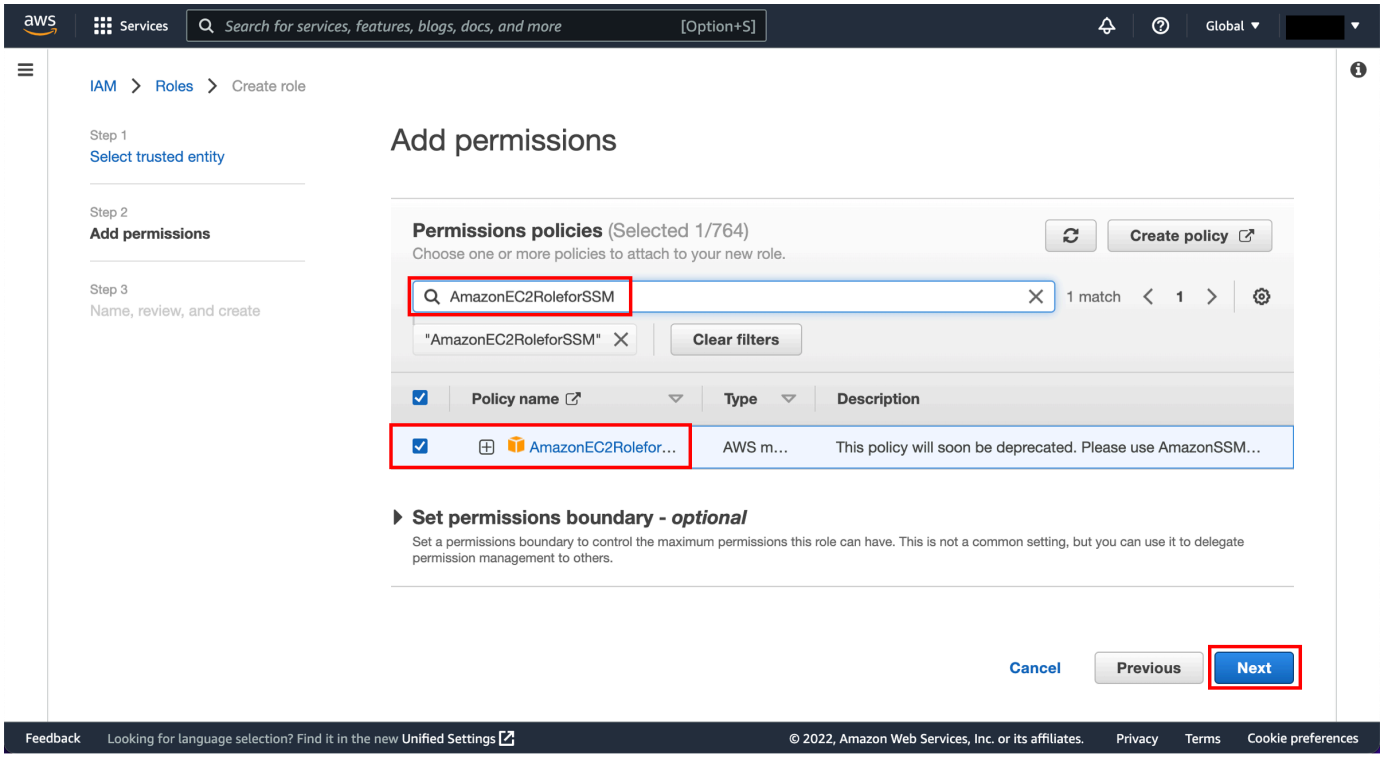
3. Select trusted entity

   On the **Select trusted entity** page, under **AWS Service**, choose **EC2**, and then choose **Next**.



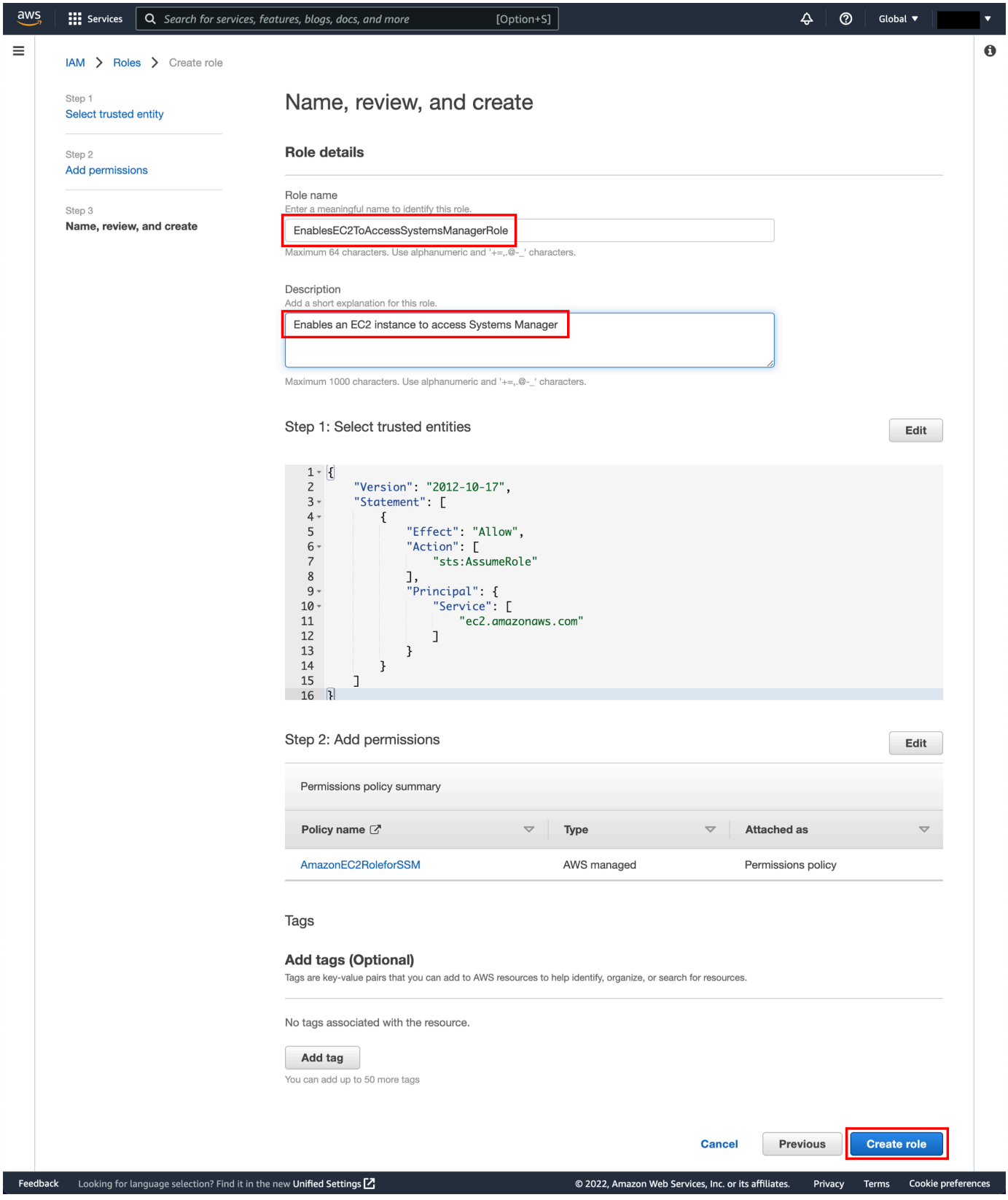
4. Add permissions

   On the **Add permissions** page, in the search bar type **AmazonEC2RoleforSSM**. From the policy list select **AmazonEC2RoleforSSM** and then choose **Next**.



5. Enter a role name and description

   On the **Name, review, and create** page, in the **Role name** box, type in **EnablesEC2ToAccessSystemsManagerRole**. In the **Description** box, type in **Enables an EC2 instance to access Systems Manager**. Choose **Create role**.

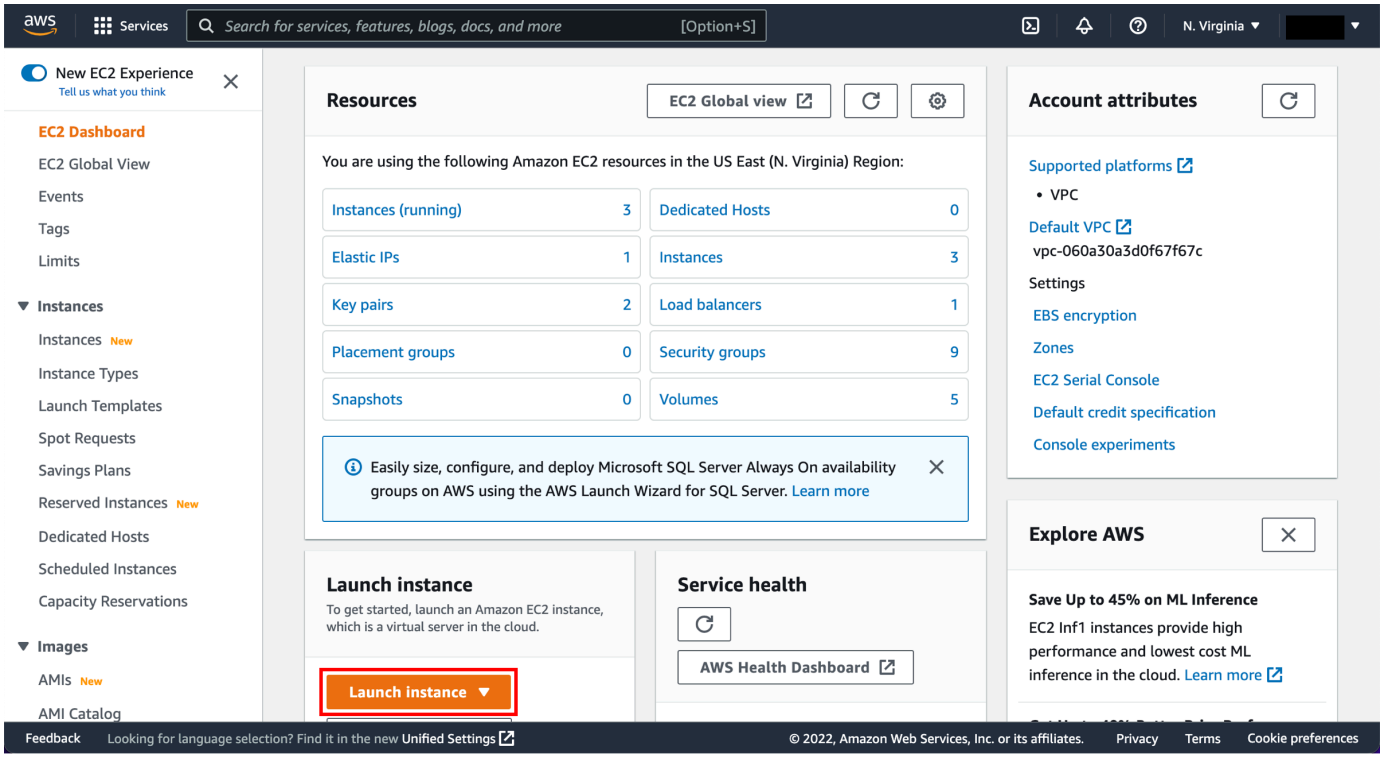

aws
Services
Search for services, features, blogs, docs, and more [Option+S]
Global
IAM > Roles > Create role
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Name, review, and create
Role details
Role name
Enter a meaningful name to identify this role.
EnablesEC2ToAccessSystemsManagerRole
Maximum 64 characters. Use alphanumeric and '+=,.@-_' characters.
Description
Add a short explanation for this role.
Enables an EC2 instance to access Systems Manager
Maximum 1000 characters. Use alphanumeric and '+=,.@-_' characters.
Step 1: Select trusted entities
Edit
{
"Version": "2012-10-17",
"Statement": [
{
"Effect": "Allow",
"Action": [
"sts:AssumeRole"
],
"Principal": {
"Service": [
"ec2.amazonaws.com"
]
}
}
]
}
Step 2: Add permissions
Edit
Permissions policy summary
Policy name
Type
Attached as
AmazonEC2RoleforSSM
AWS managed
Permissions policy
Tags
Add tags (Optional)
Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.
No tags associated with the resource.
Add tag
You can add up to 50 more tags
Cancel
Previous
Create role
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Step 2: Create an EC2 instance

Now that you have an EC2 instance running the Systems Manager agent, you can automate administration tasks and manage the instance. In this step, you run a pre-packaged command, called a document, that will upgrade the agent. It is best practice to update the Systems Manager Agent when you create a new instance.

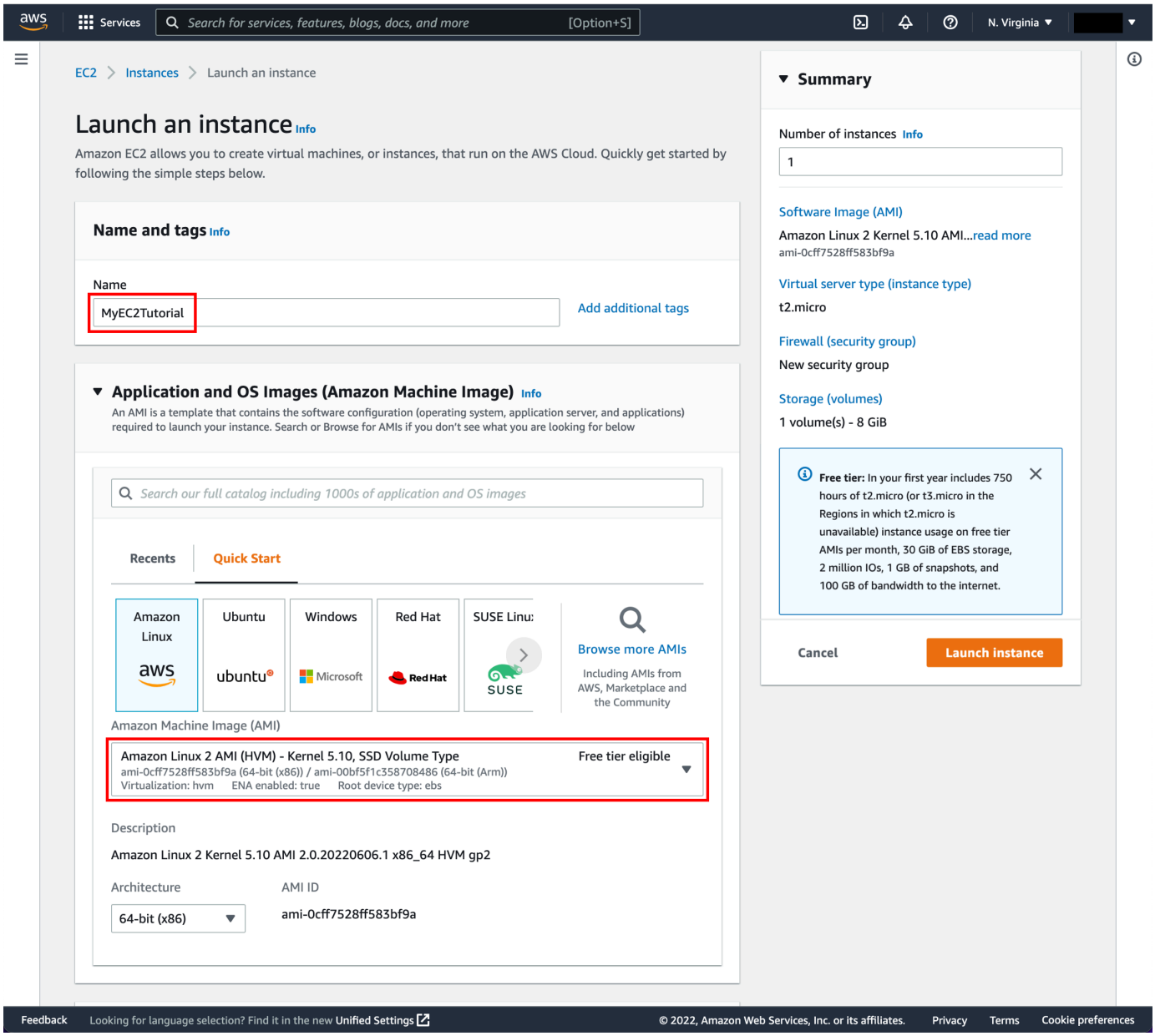
1. Launch an EC2 instance

   Open the Amazon EC2 console. From the EC2 console, select your preferred Region. Systems Manager is supported in all AWS Regions. Now choose **Launch instance**.



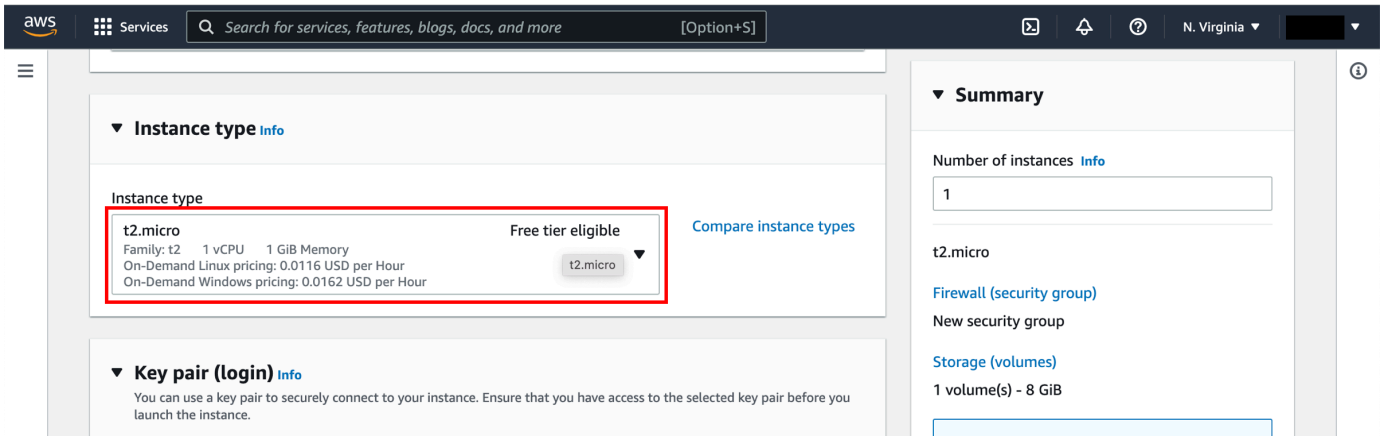
2. Enter an instance name and choose an AMI

   In the **Name** field, enter **MyEC2Tutorial**. Select the **Amazon Linux AMI**. Retain the default selection that appears in the dropdown. You can also install the Systems Manager Agent on your own Windows or Linux system.



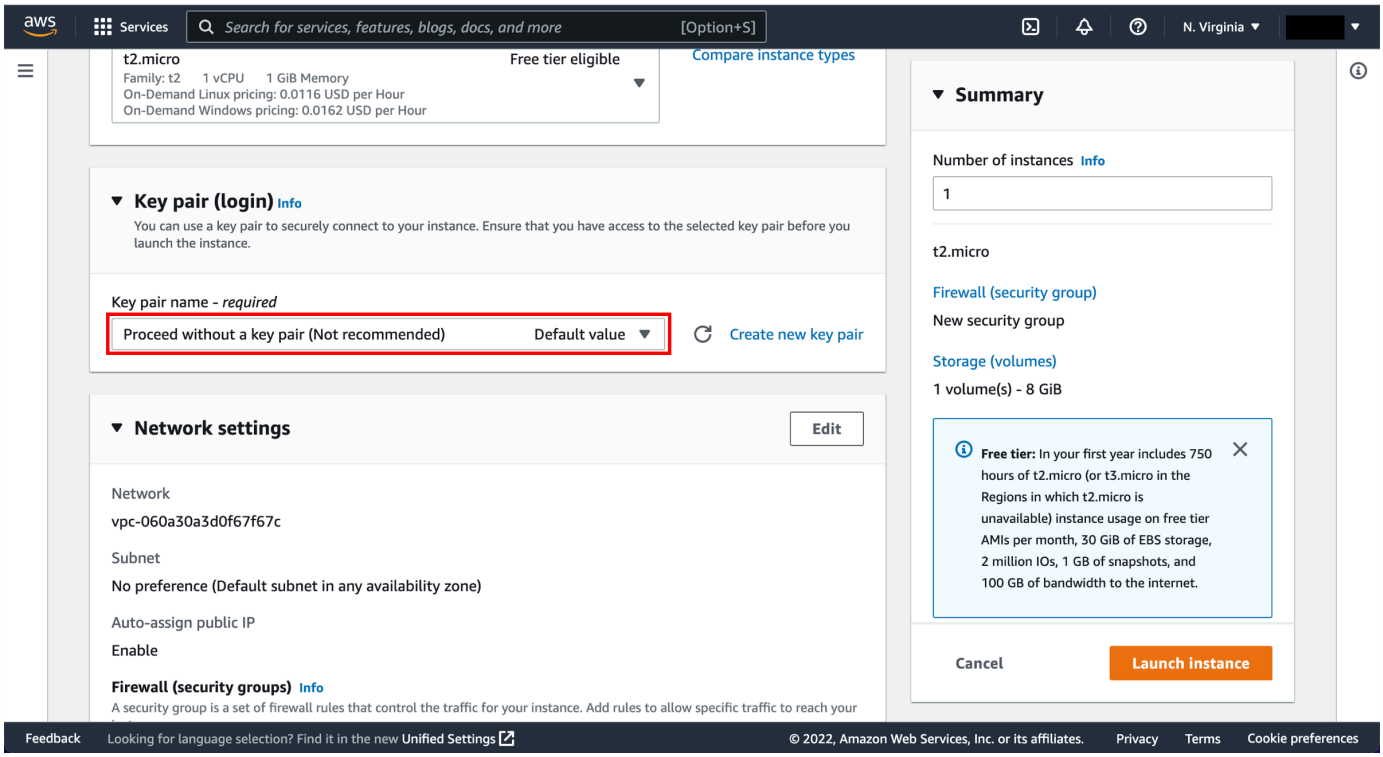
3. Choose an instance type

   Choose the t2.micro instance type.



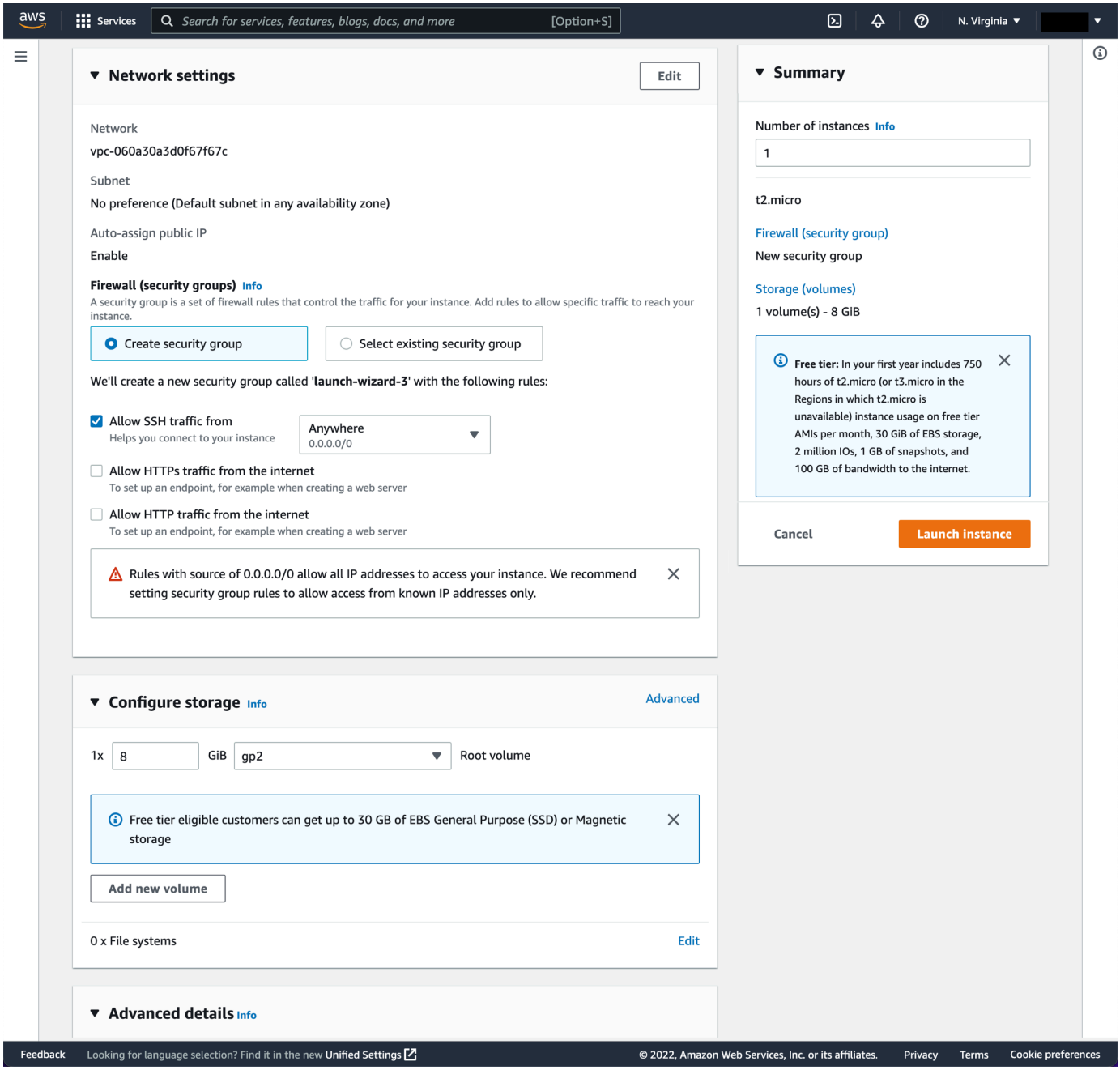
4. Choose to proceed without a key pair

   You will not need a keypair to use Systems Manager to remotely run commands. Scroll down to **Key pair** and under the **Key pair name** dropdown, choose **Proceed without a key pair**.



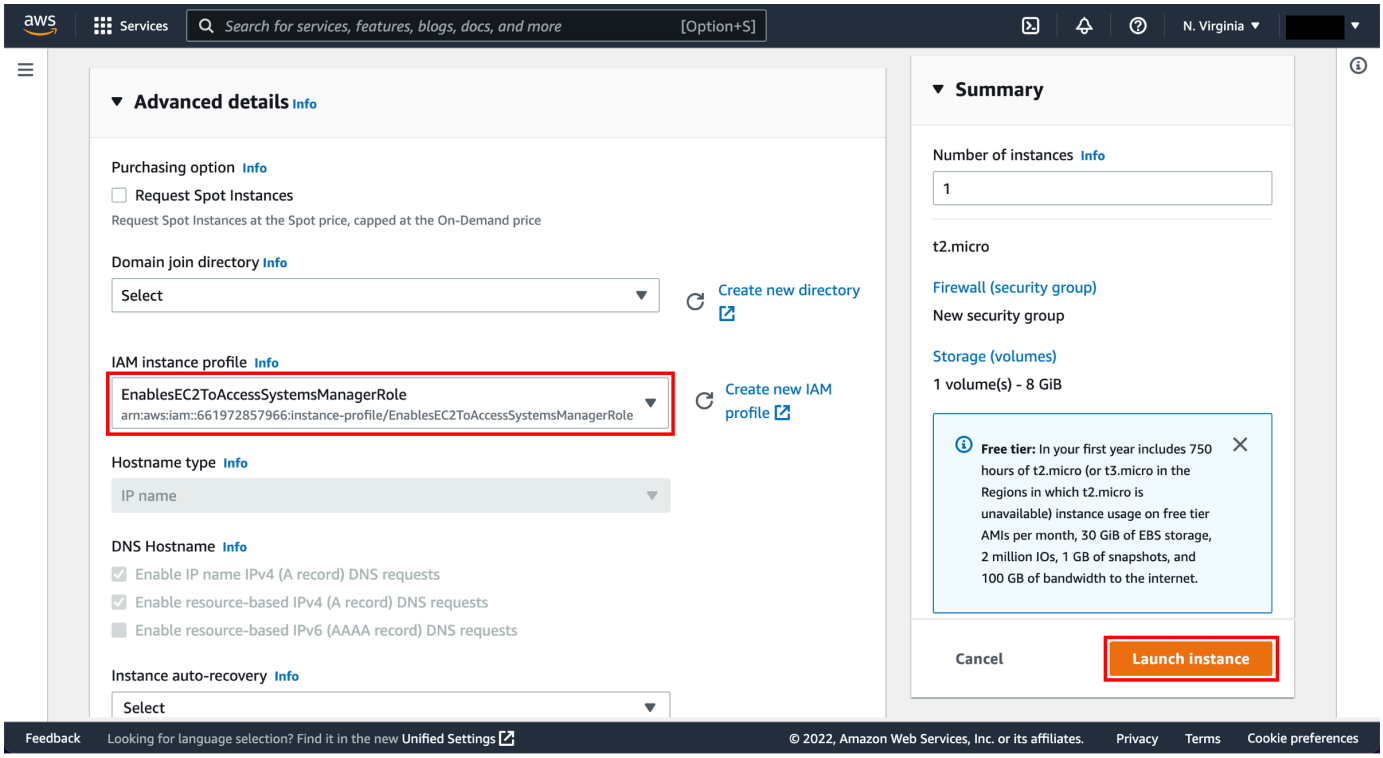
5. Keep default network and storage

   Retain default settings under **Network settings** and **Configure storage**.



6. Attach the IAM role to the EC2 instance

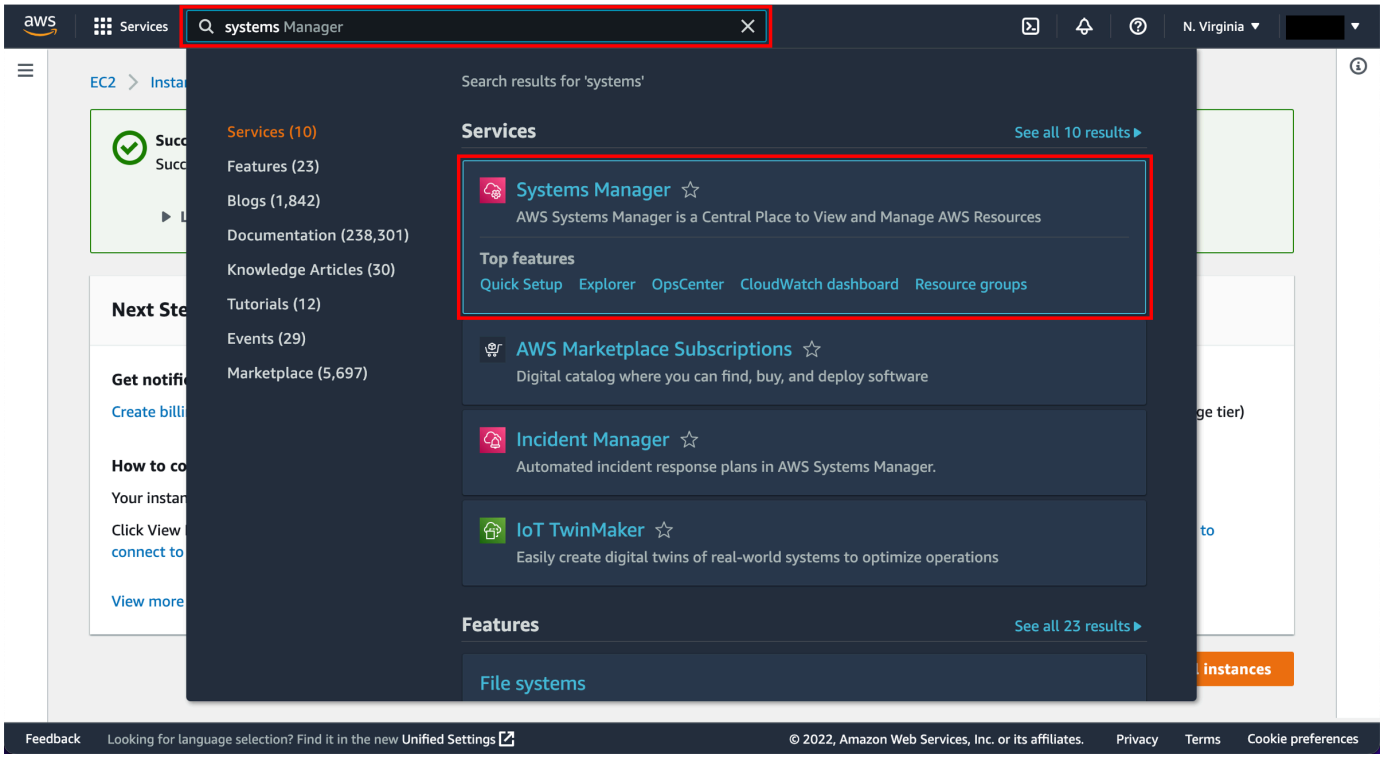
   Under **Advanced details**, in the IAM instance profile dropdown choose the EnablesEC2ToAccessSystemsManagerRole role you created earlier. Leave everything else as default. Choose **Launch instance**.



Step 3: Run a remote shell script

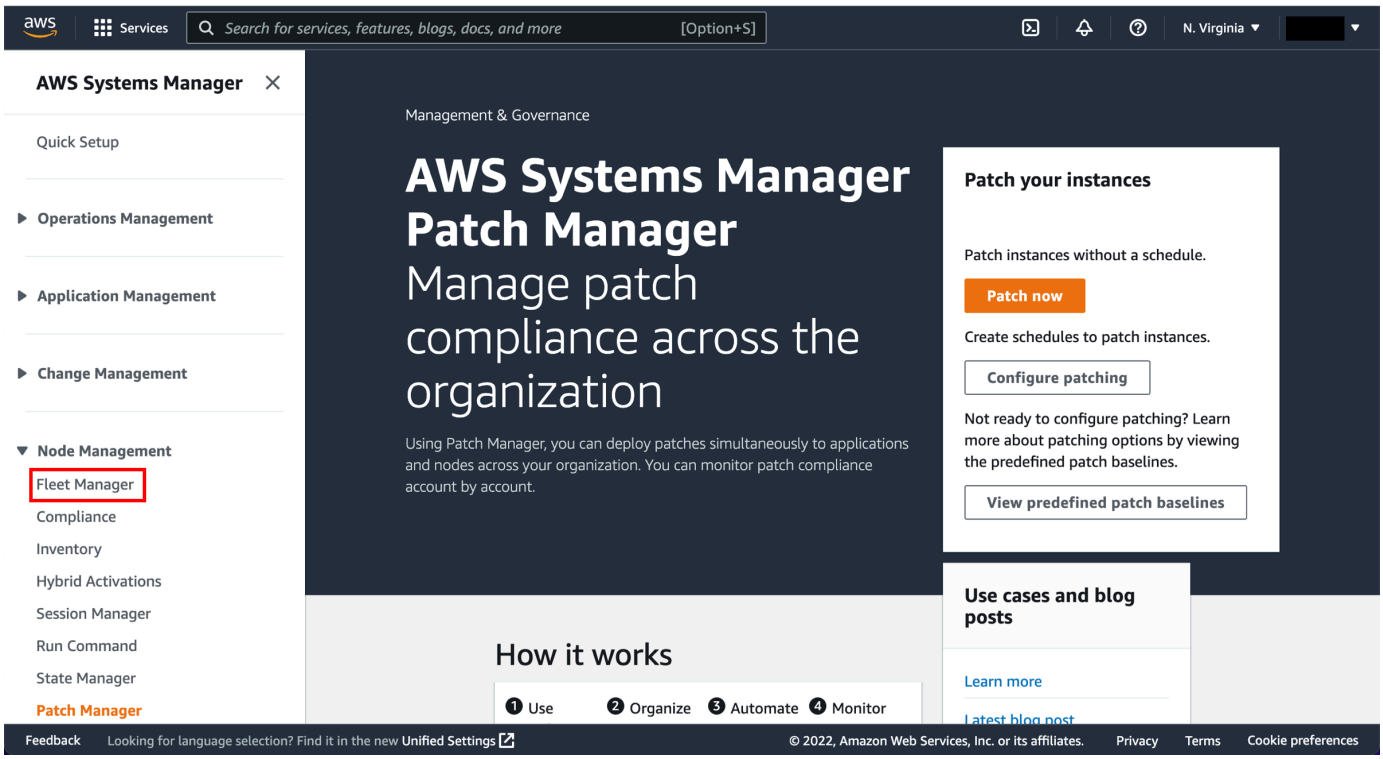
1. Open Systems Manager

   In the top navigation bar, search for **Systems Manager** and open the Systems Manager console.



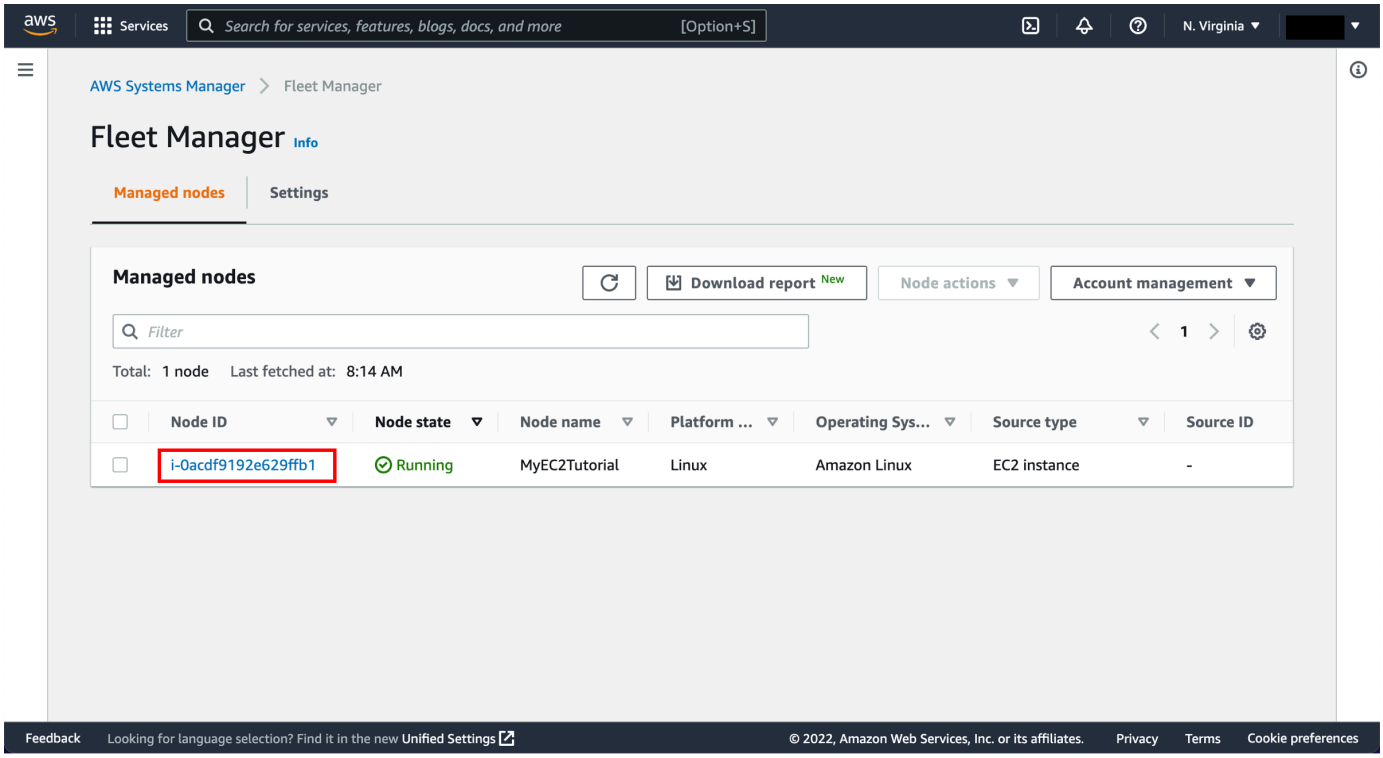
2. Choose Fleet Manager

Under the **Node Management** section on the left navigation bar, choose **Fleet Manager**.



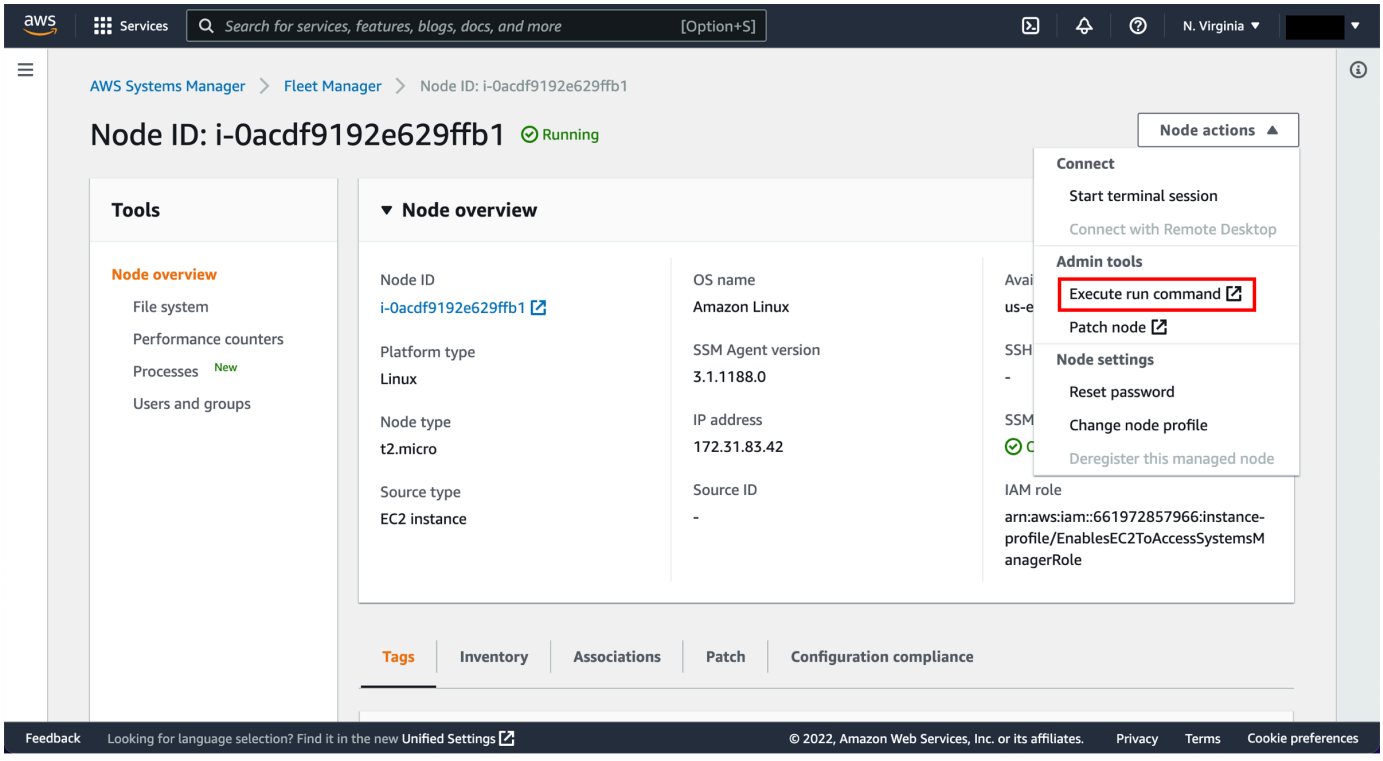
3. Choose an instance

Select the node ID created in step 2, MyEC2Tutorial, to open the node detail page.



4. Choose Run Command

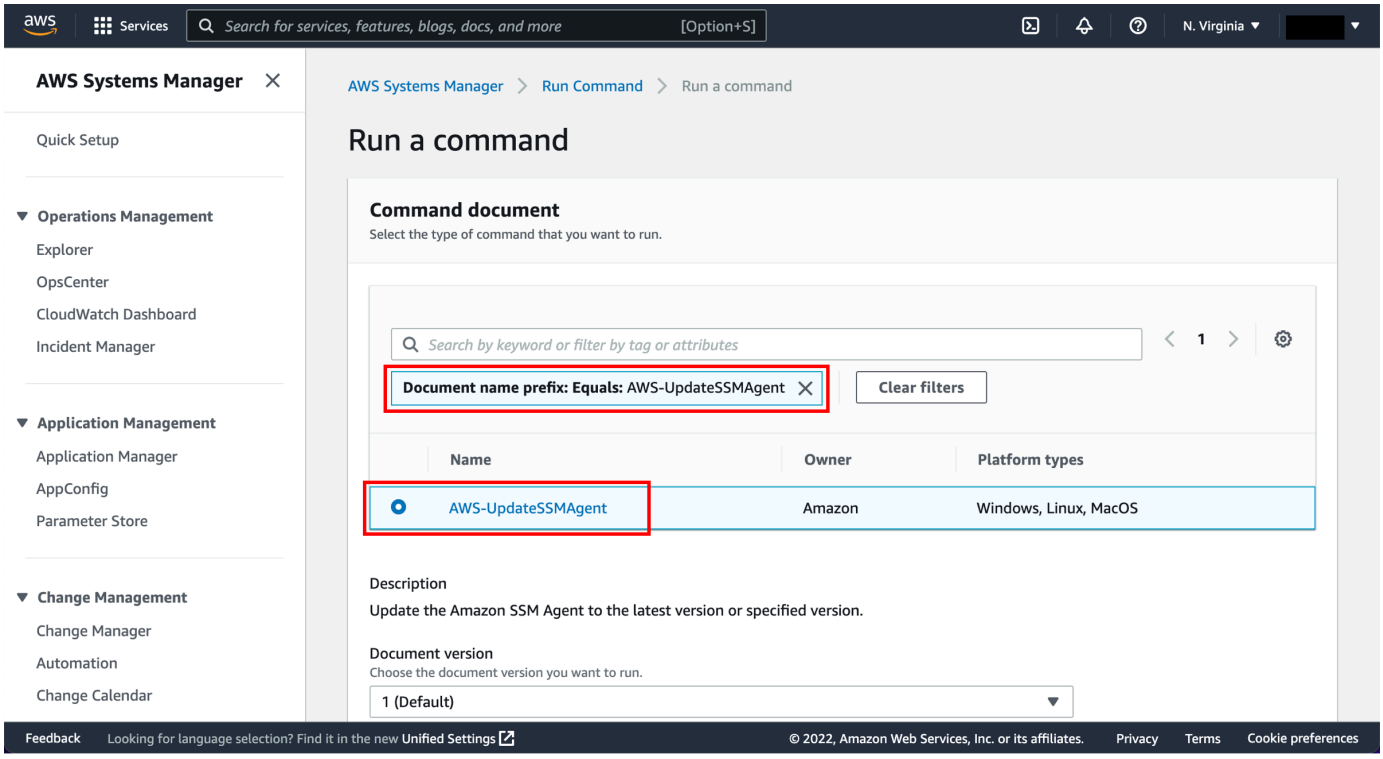
On the node detail page, in the **Node actions** dropdown, select **Execute run command**.



5. Choose AWS-UpdateSSMAgent

On the **Run a command** page, click in the search bar and select, **Document name prefix**, then click on **Equals**, then type in **AWS-UpdateSSMAgent**.

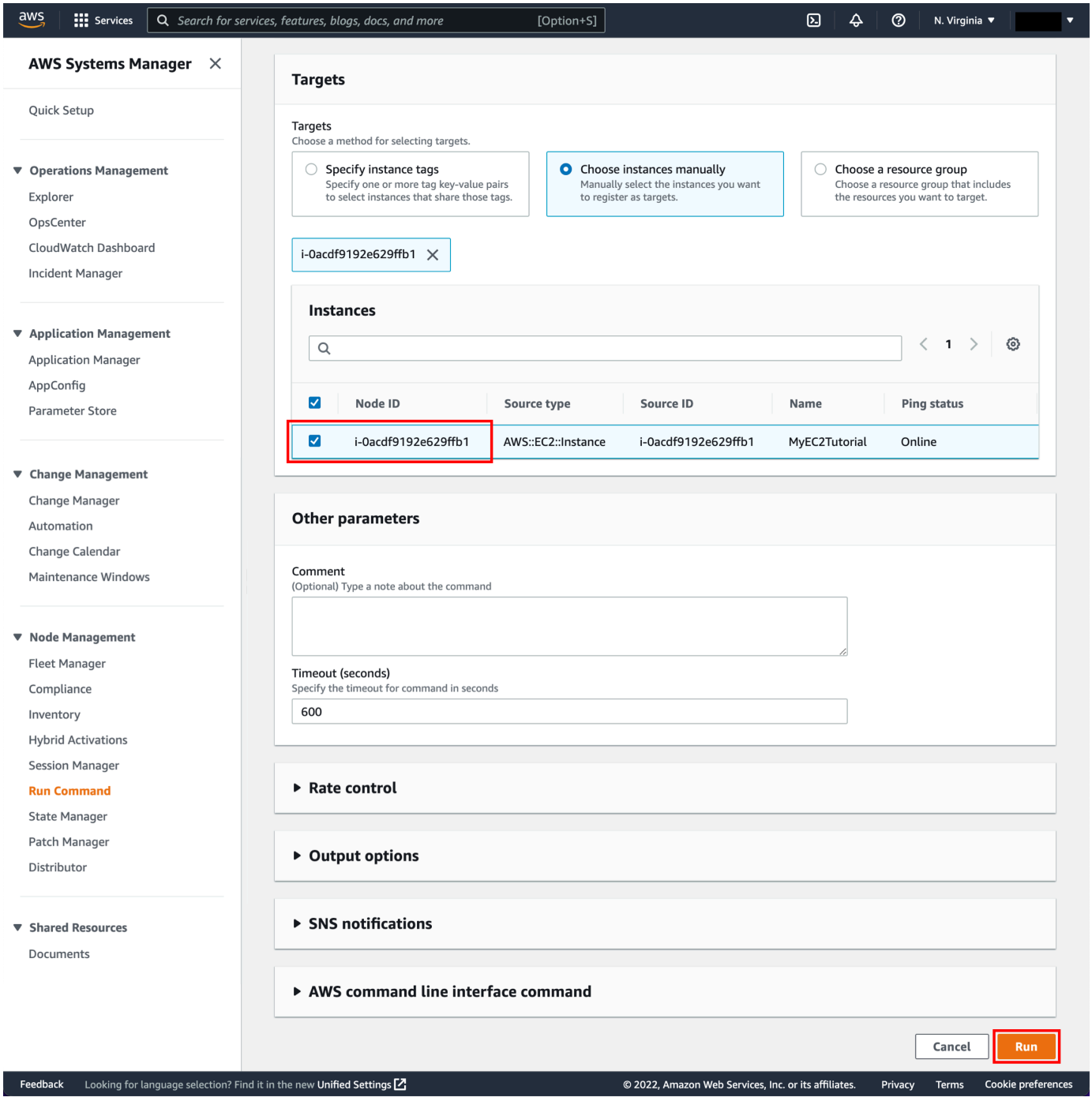
Now select the radio button on the left of **AWS-UpdateSSMAgent**. This document will upgrade the Systems Management agent on the instance.



6. Select targets

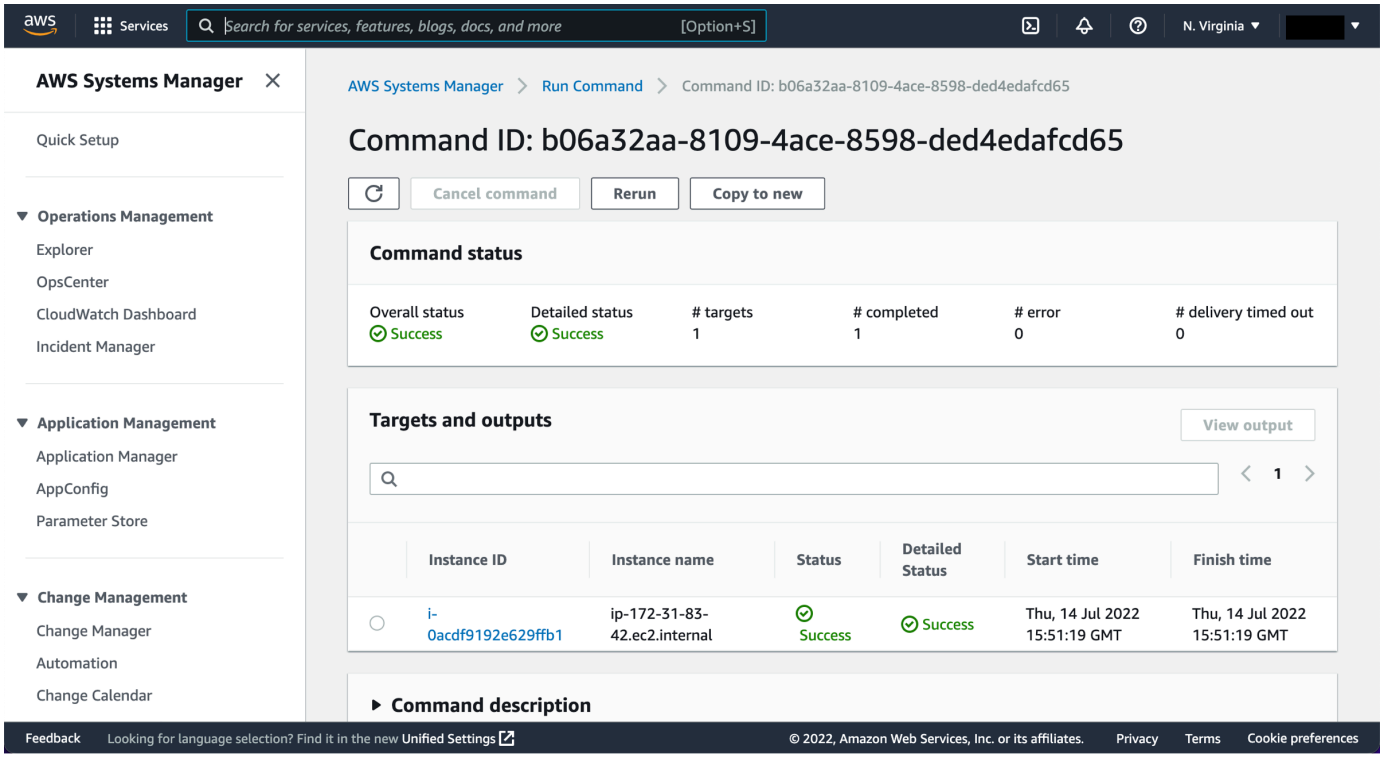
Scroll down to the **Targets** panel and select the check box next to your managed EC2 instance.

Finally, scroll down and select **Run**.



7. Select targets

Next you will see a page documenting your running command, and then overall success in green. Congrats, you have just run your first remote command using Systems Manager.



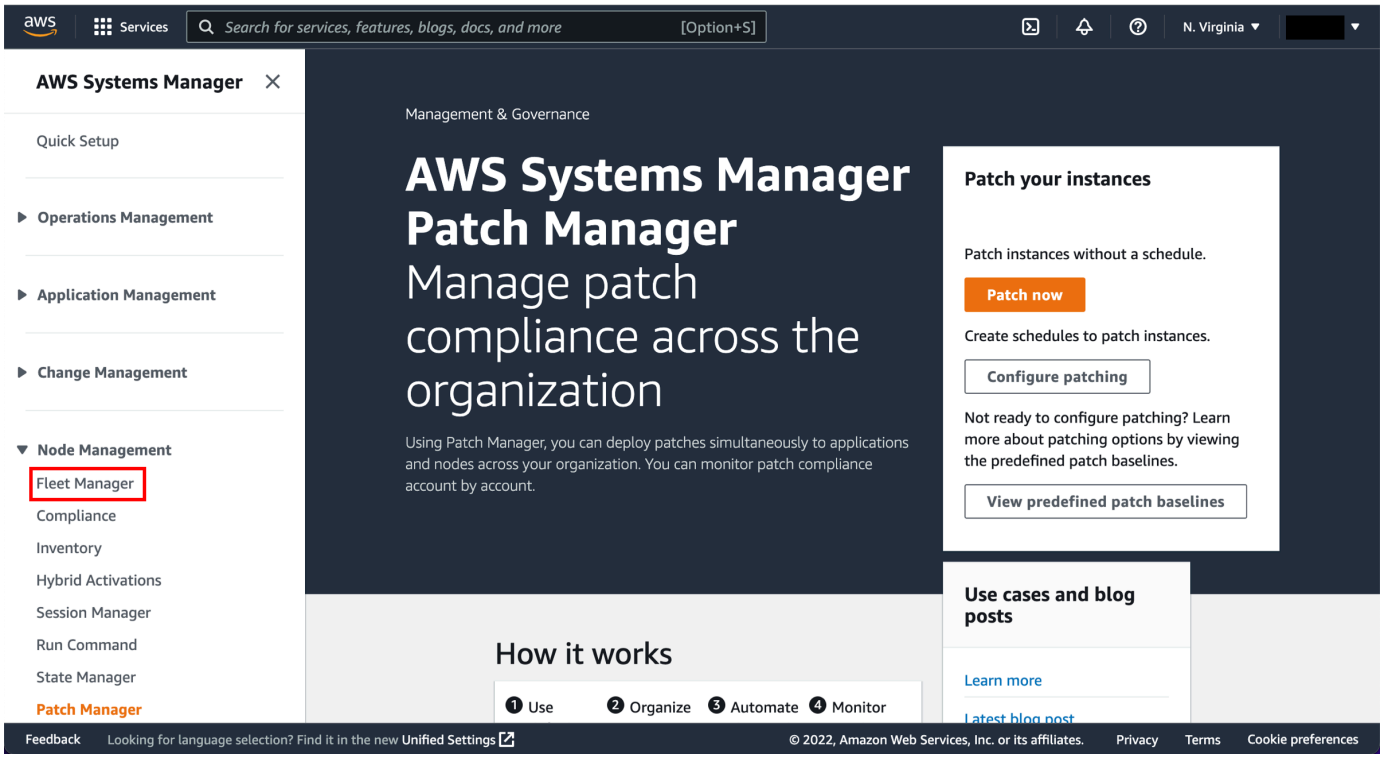
Step 4: Terminate your resources

In this step, you will terminate your Systems Manager and EC2 related resources.

> **Important**
>
> Terminating resources that are not actively being used reduces costs and is a best practice. Not terminating your resources can result in a charge.

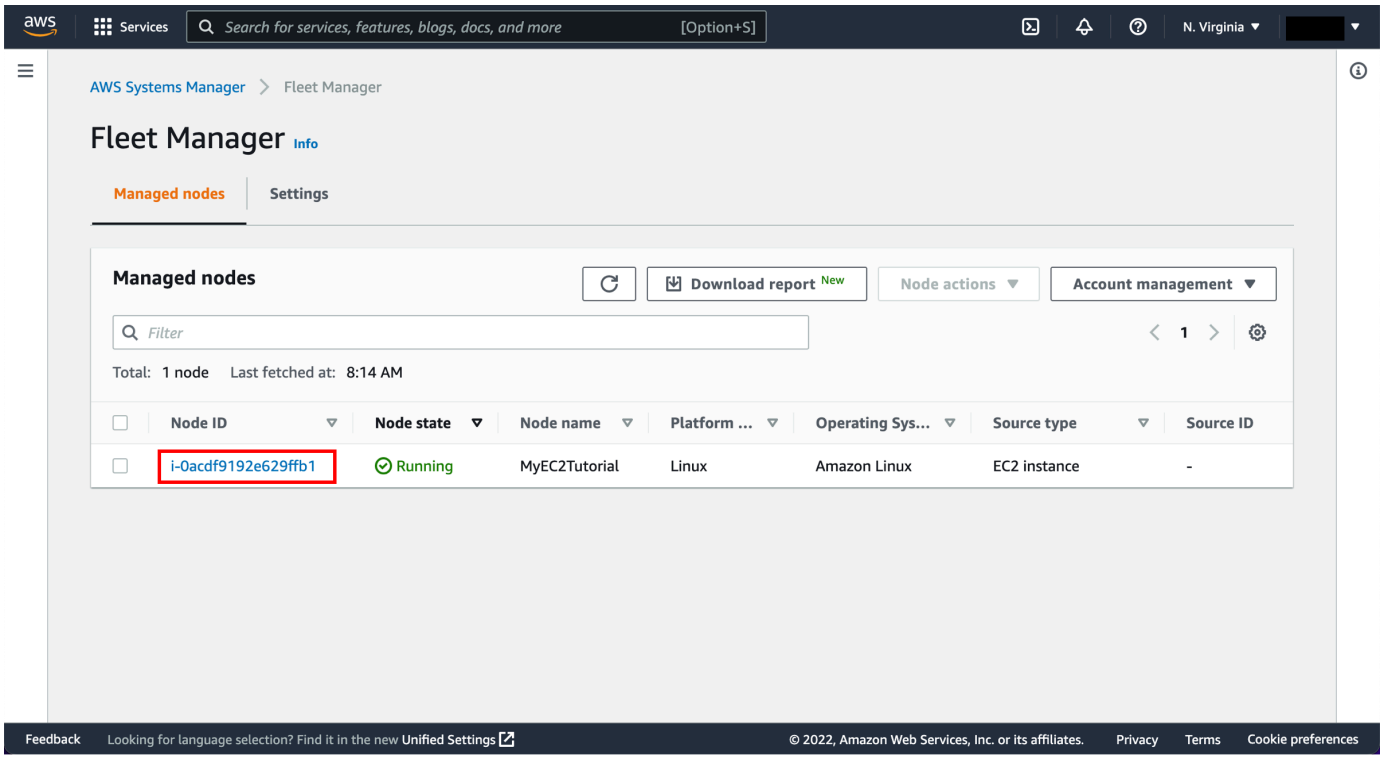
1. Choose Fleet Manager

   Under the **Node Management** section on the left navigation bar, choose **Fleet Manager**.



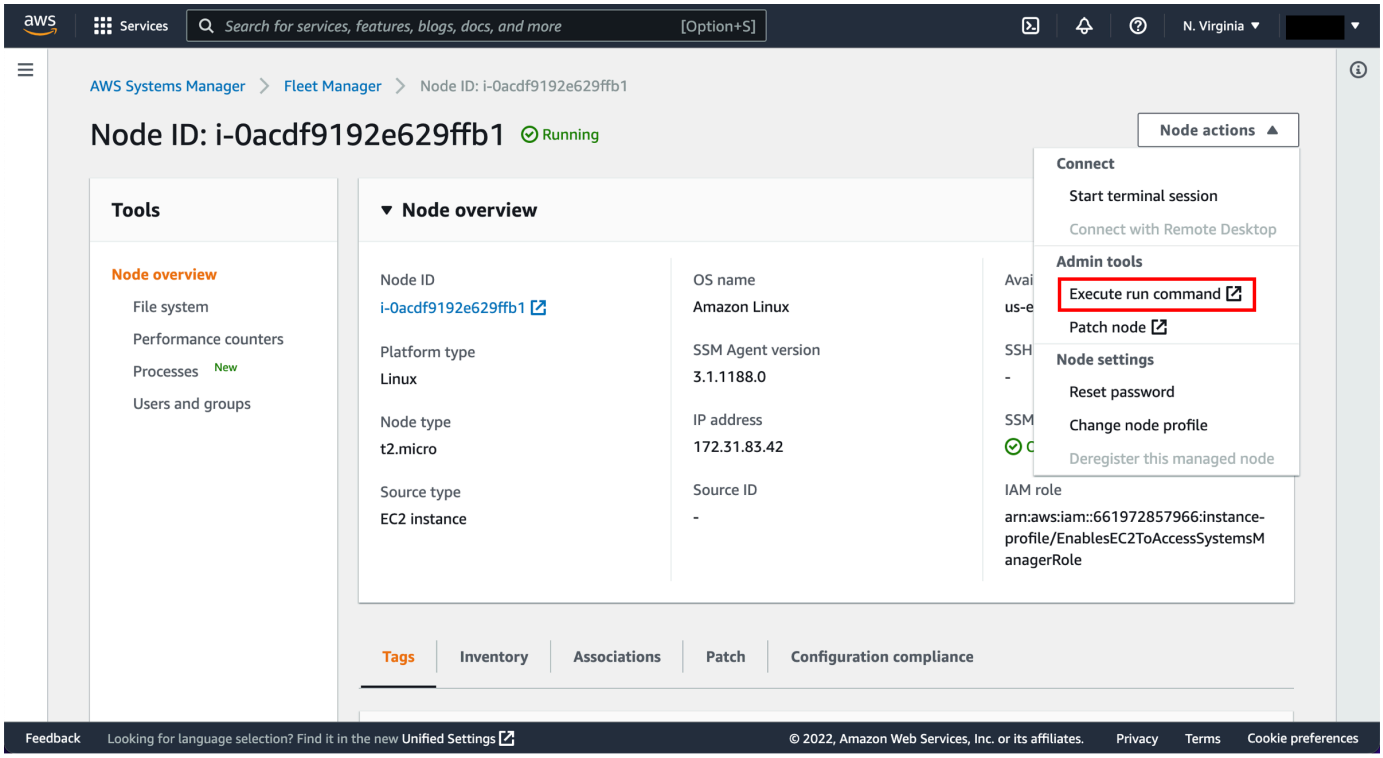
2. Choose an instance

Select the node ID created in step 2, MyEC2Tutorial, to open the node detail page.



3. Choose Run Command

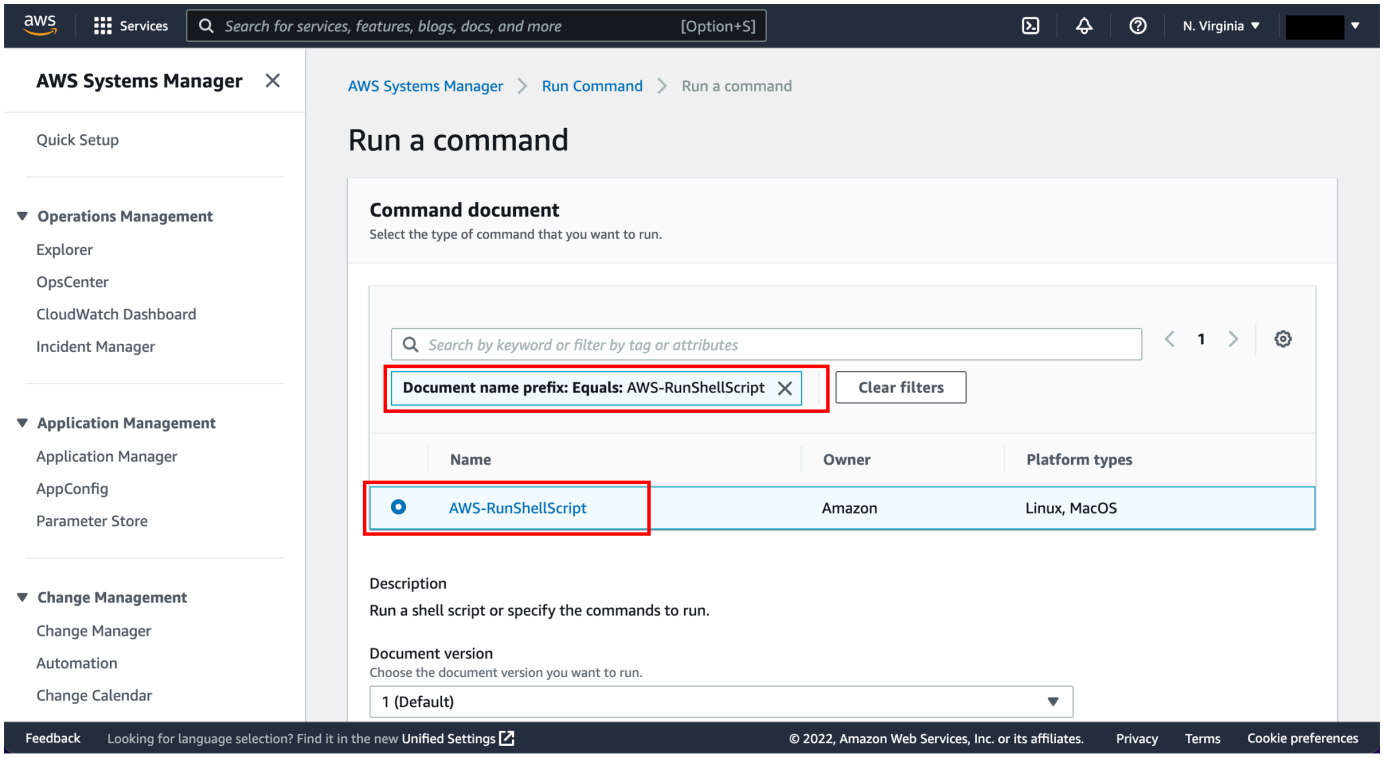
On the node detail page, in the **Node actions** dropdown, select **Execute run command**.



4. Choose AWS-RunShellScript

   On the **Run a command** page, click in the search bar and select, **Document name prefix**, then click on **Equals**, then type in **AWS-RunShellScript**.

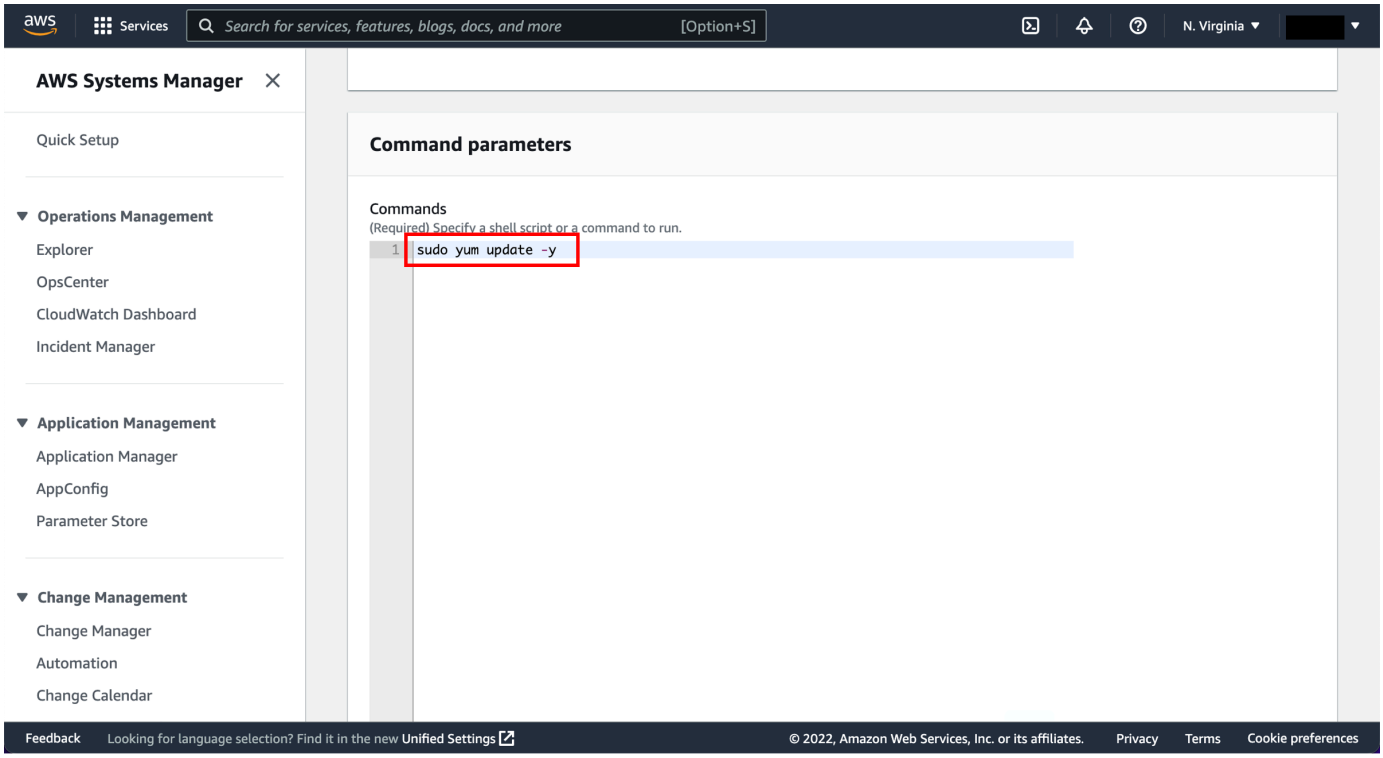
   Now select the radio button on the left of AWS-RunShellScript.



5. Enter update command

Scroll down to the **Command Parameters** panel and insert the following command in the **Commands** text box:

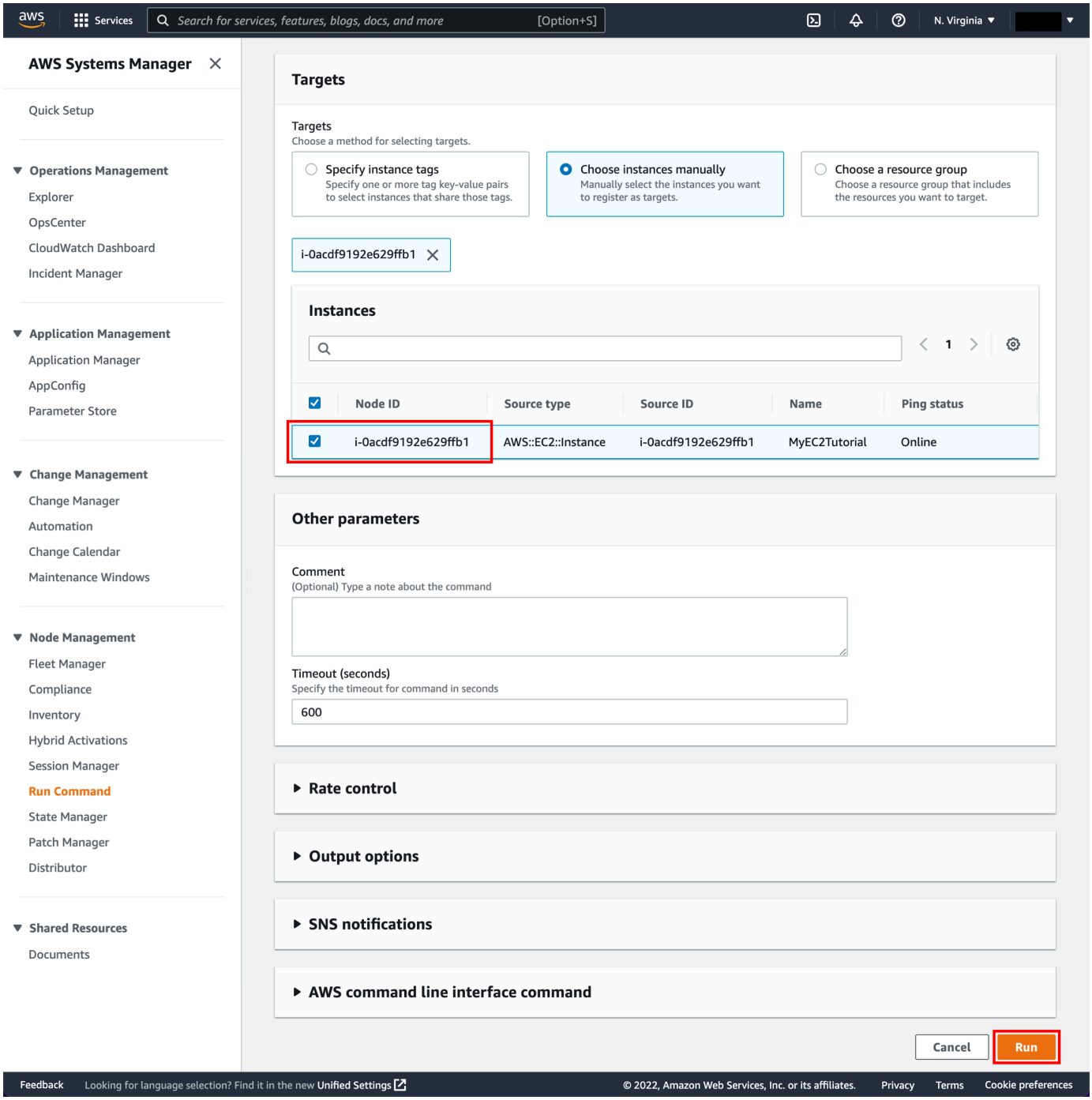
```
sudo yum update -y
```



6. Select targets

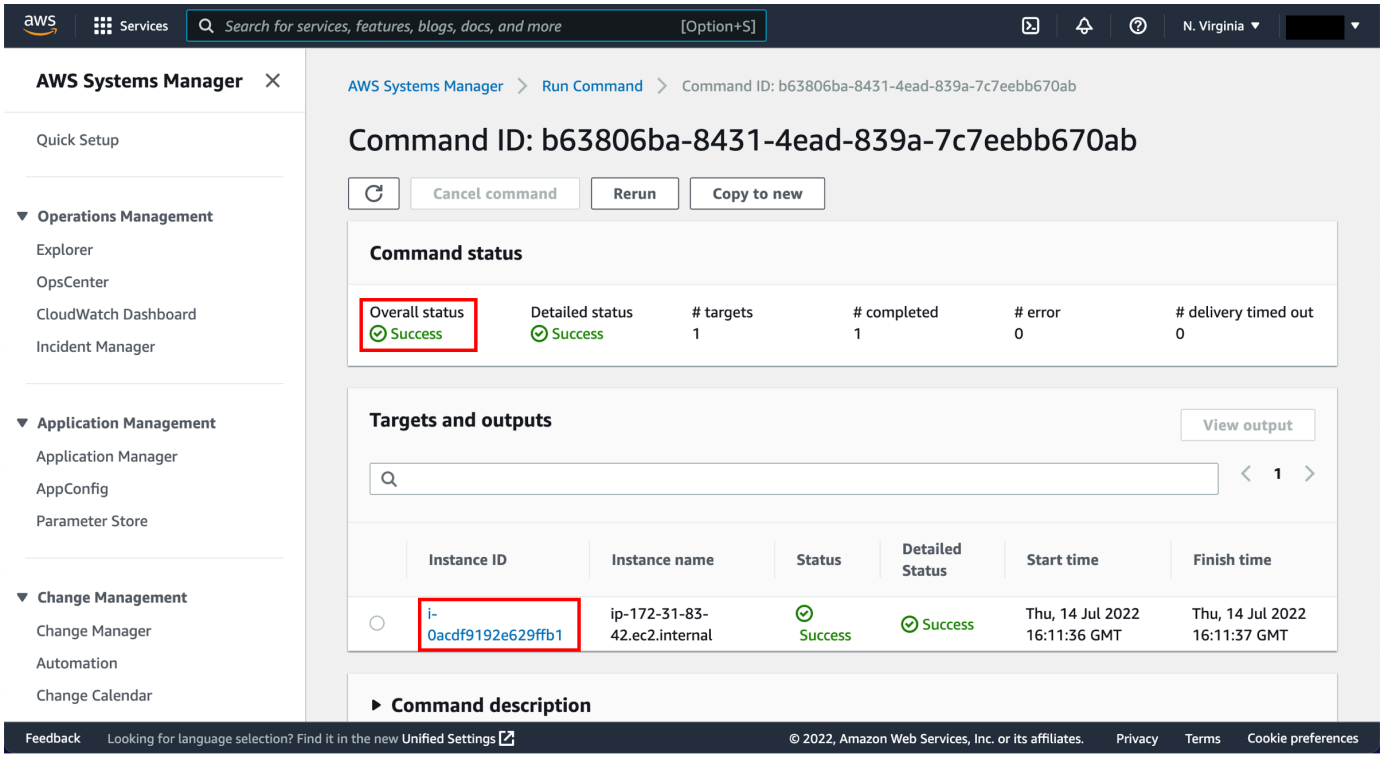
   Scroll down to the **Targets** panel and select the check box next to your managed EC2 instance.

   Finally, scroll down and select **Run**.



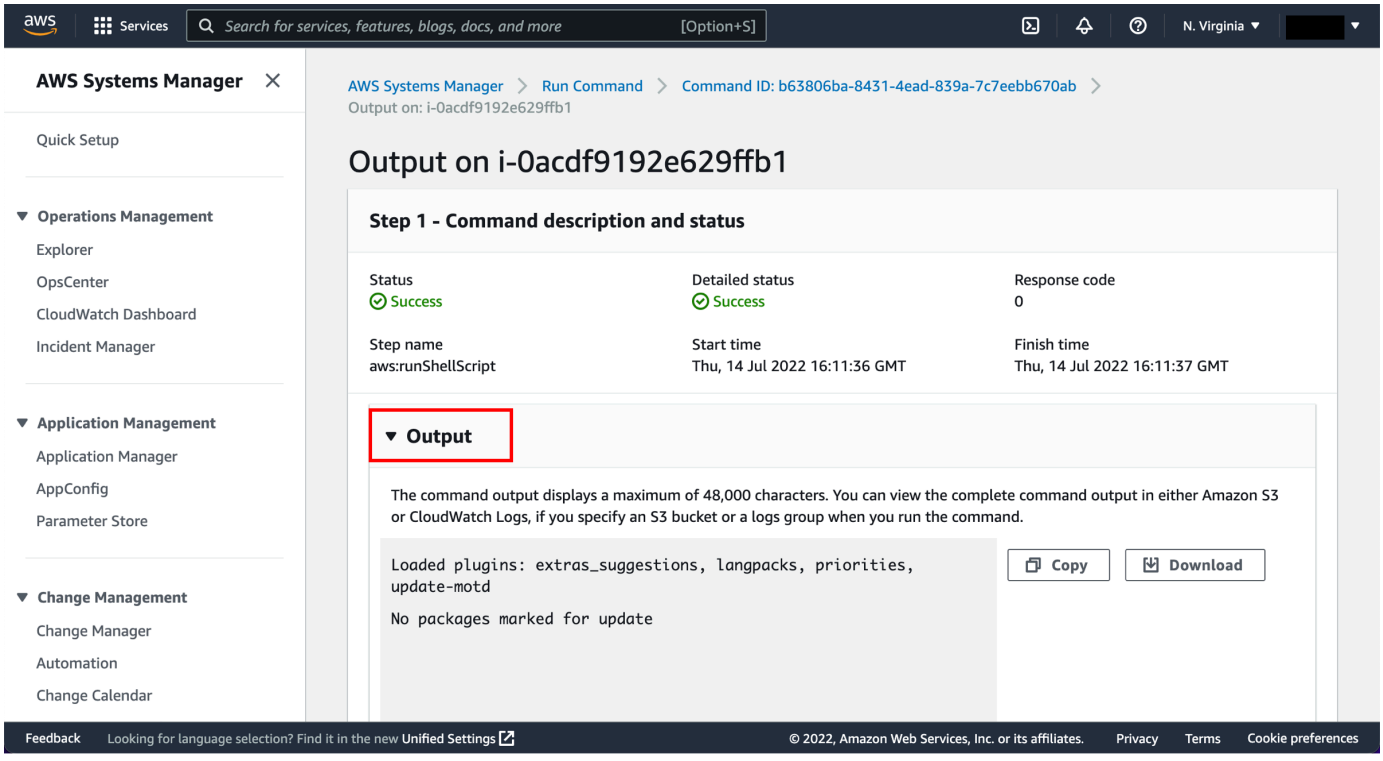
7. View command status

While your script is running remotely on the managed EC2 instance, the **Overall status** will be **In Progress**. Soon the **Overall status** will turn to **Success**. When it does, scroll down to the **Targets and outputs** panel and select the Instance ID of your instance. Your Instance ID will be different than the one pictured.



8. View command output

From the **Output on: i-XX** page, select the header of the **Output** panel to view the output of the update command from the instance.



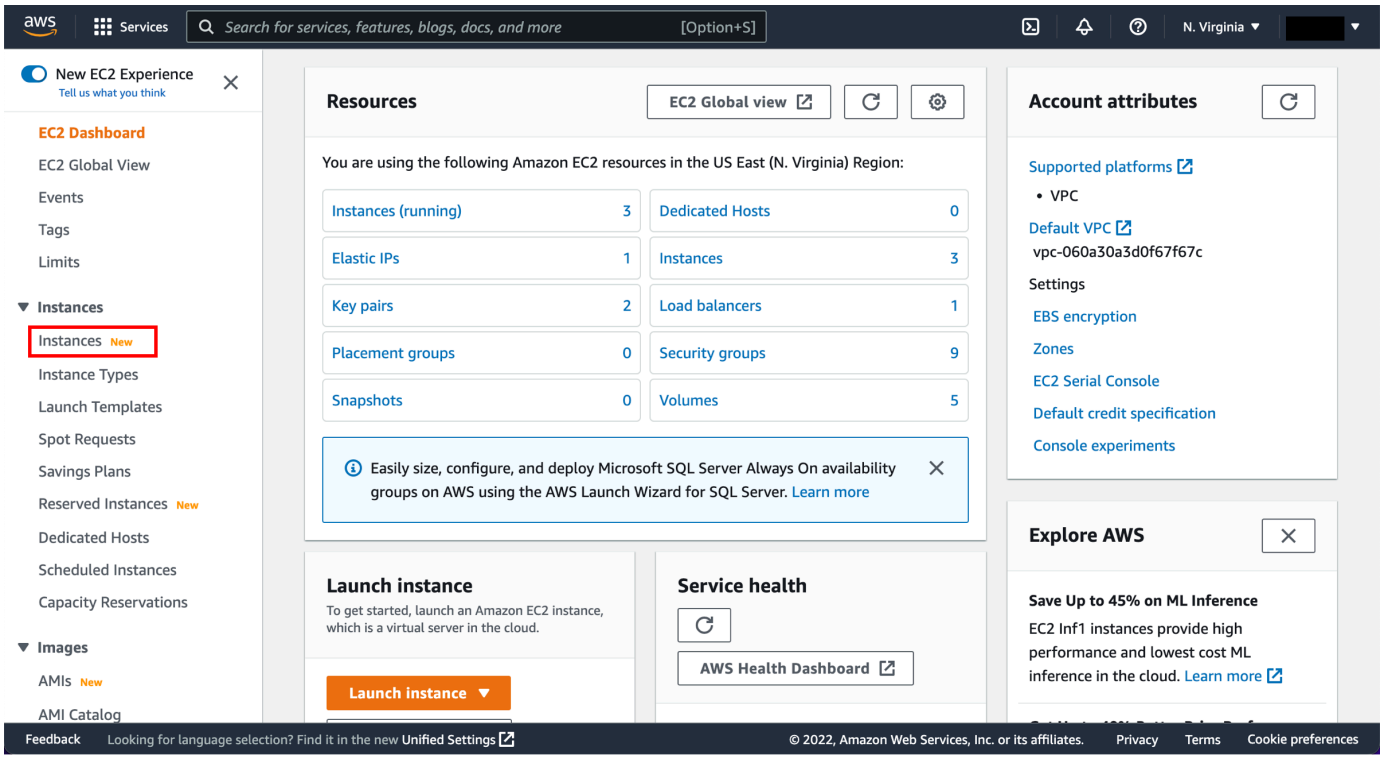
Step 5: Update the Systems Manager Agent

In this step, you will terminate your Systems Manager and EC2 related resources.

> **Important**
>
> Terminating resources that are not actively being used reduces costs and is a best practice. Not terminating your resources can result in a charge.

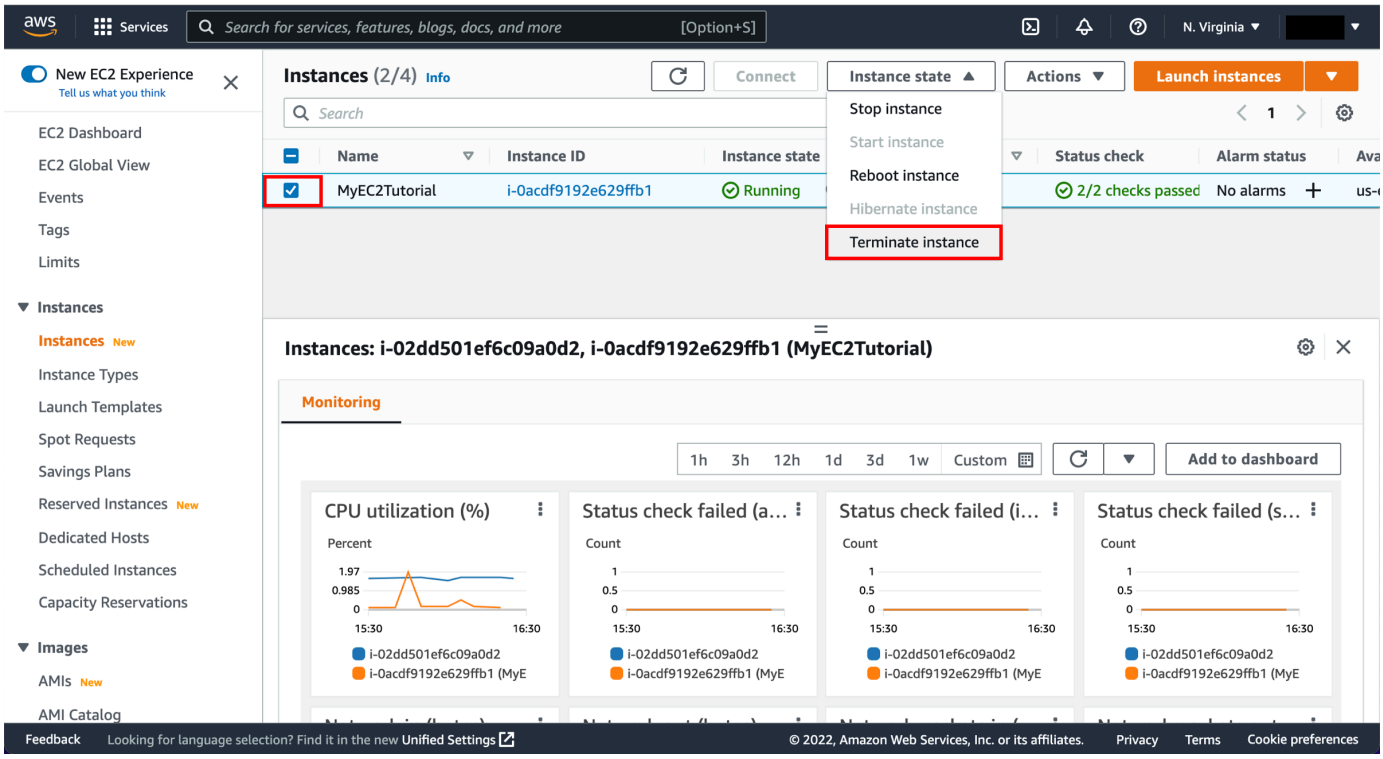
1. Open the EC2 console and choose Instances

   Open the Amazon EC2 console and from the left navigation under the **Instances** heading, select **Instances**.



2. Terminate your instance

   Select your instance's checkbox and choose **Instance state**, then select **Terminate instance**. This will terminate your instance completely.



Congratulations

Congratulations, you have successfully created a managed instance and remotely run a command using AWS Systems Manager. You first set up the correct permissions through IAM. Next you launched an Amazon Linux instance that was preinstalled with the Systems Manager agent. Finally, you used Run Command to update the agent and remotely perform a yum update.

Systems Manager is a good choice when you need to view operation data for groups of resources, automate operational actions, understand and control the current state of your resources, manage hybrid environments, and maintain security and compliance.