



Hands-on tutorials

Remotely Run Commands on an EC2 Instance with AWS Systems Manager



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	i
Overview	1
Implementation	2
Congratulations	24

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.

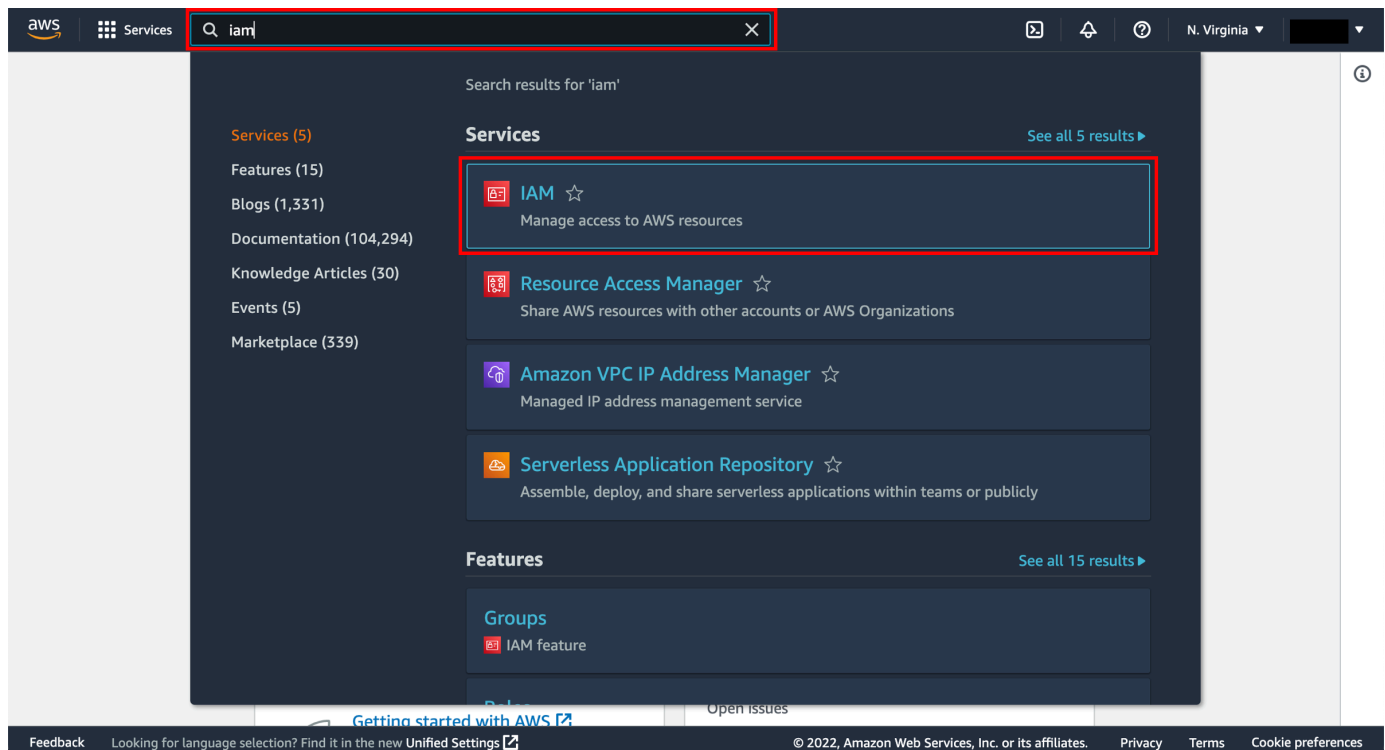
Implementation

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.

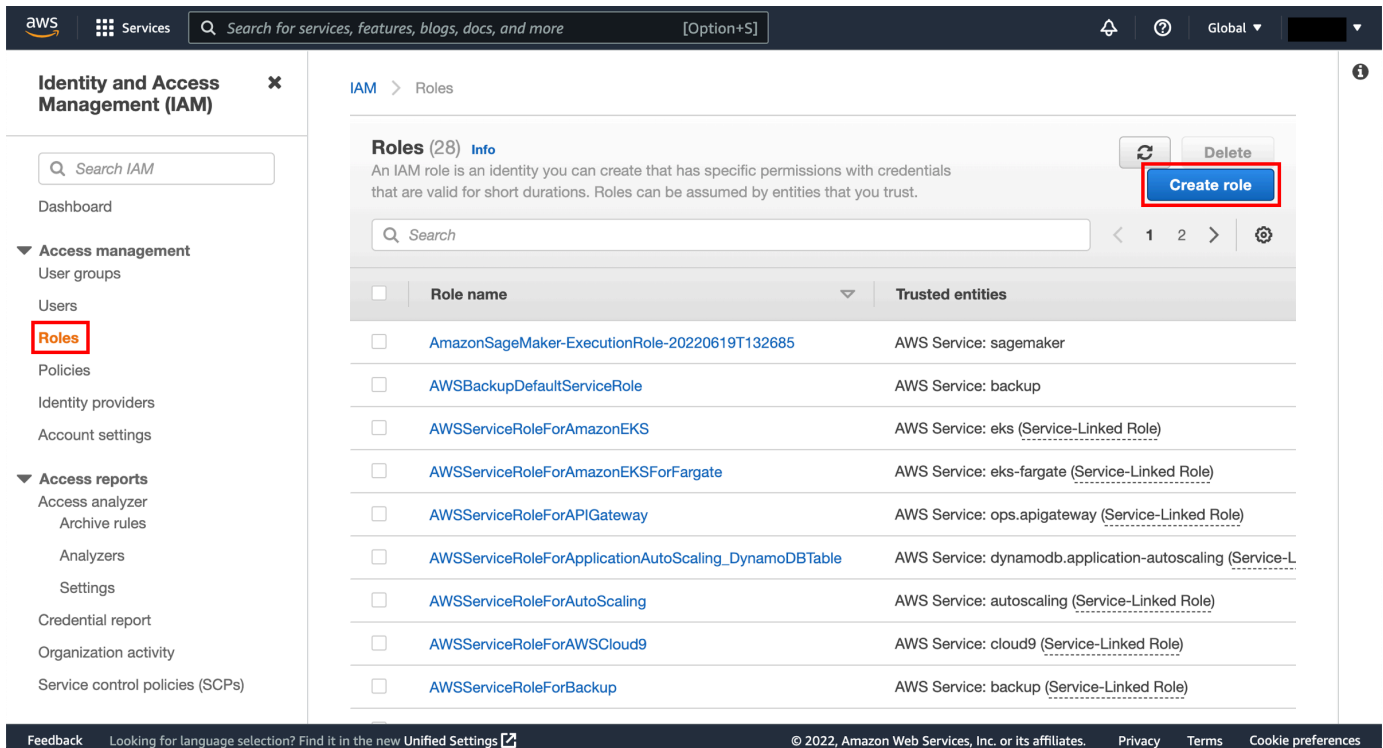
1. Open the IAM console

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**.



The screenshot shows the AWS IAM console interface. On the left, the navigation menu is visible, with 'Roles' highlighted under 'Access management'. The main content area displays the 'Roles (28)' page, which includes a search bar, a 'Create role' button (circled in red), and a table of existing roles. The table has columns for 'Role name' and 'Trusted entities'.

<input type="checkbox"/>	Role name	Trusted entities
<input type="checkbox"/>	AmazonSageMaker-ExecutionRole-20220619T132685	AWS Service: sagemaker
<input type="checkbox"/>	AWSBackupDefaultServiceRole	AWS Service: backup
<input type="checkbox"/>	AWSServiceRoleForAmazonEKS	AWS Service: eks (Service-Linked Role)
<input type="checkbox"/>	AWSServiceRoleForAmazonEKSFargate	AWS Service: eks-fargate (Service-Linked Role)
<input type="checkbox"/>	AWSServiceRoleForAPIGateway	AWS Service: ops.apigateway (Service-Linked Role)
<input type="checkbox"/>	AWSServiceRoleForApplicationAutoScaling_DynamoDBTable	AWS Service: dynamodb.application-autoscaling (Service-L
<input type="checkbox"/>	AWSServiceRoleForAutoScaling	AWS Service: autoscaling (Service-Linked Role)
<input type="checkbox"/>	AWSServiceRoleForAWSCloud9	AWS Service: cloud9 (Service-Linked Role)
<input type="checkbox"/>	AWSServiceRoleForBackup	AWS Service: backup (Service-Linked Role)

3. Select trusted entity

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

The screenshot shows the AWS IAM console interface for creating a role. The breadcrumb trail is IAM > Roles > Create role. The page is in Step 1: Select trusted entity. The 'Trusted entity type' section has five options: 'AWS service' (selected), 'AWS account', 'Web identity', 'SAML 2.0 federation', and 'Custom trust policy'. The 'Use case' section has 'EC2' selected under 'Common use cases'. A 'Next' button is highlighted with a red box.

4. Add permissions

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

The screenshot shows the AWS IAM console interface for creating a role. The breadcrumb navigation is IAM > Roles > Create role. The left sidebar shows three steps: Step 1: Select trusted entity, Step 2: Add permissions (current), and Step 3: Name, review, and create. The main content area is titled 'Add permissions' and shows 'Permissions policies (Selected 1/764)'. A search bar contains 'AmazonEC2RoleforSSM' and shows 1 match. A table lists the selected policy: 'AmazonEC2Rolefor...' with type 'AWS m...' and description 'This policy will soon be deprecated. Please use AmazonSSM...'. Below the table is a section for 'Set permissions boundary - optional'. At the bottom right, there are 'Cancel', 'Previous', and 'Next' buttons, with 'Next' highlighted in red.

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

```
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action": [
7         "sts:AssumeRole"
8       ],
9       "Principal": {
10        "Service": [
11          "ec2.amazonaws.com"
12        ]
13      }
14    }
15  ]
16 }
```

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

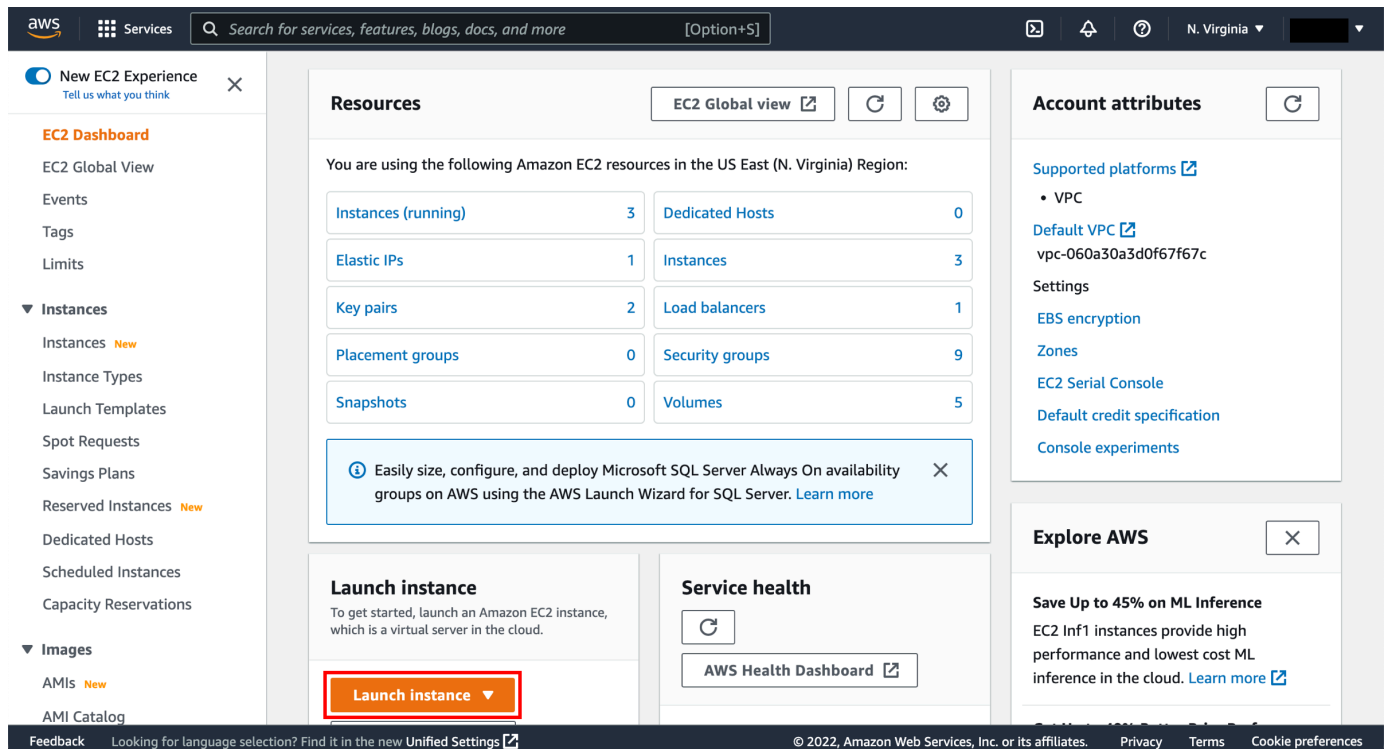
Cancel Previous **Create role**

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**.



The screenshot shows the Amazon EC2 console interface. The top navigation bar includes the AWS logo, a search bar, and the region 'N. Virginia'. The left sidebar contains navigation options like 'EC2 Dashboard', 'Instances', and 'Images'. The main content area is divided into several sections:

- Resources:** A table showing the usage of various Amazon EC2 resources in the US East (N. Virginia) Region. The resources and their counts are: Instances (running) - 3, Elastic IPs - 1, Key pairs - 2, Placement groups - 0, Snapshots - 0, Dedicated Hosts - 0, Instances - 3, Load balancers - 1, Security groups - 9, and Volumes - 5.
- Account attributes:** A section showing account details such as 'Supported platforms', 'Default VPC', and 'Settings'.
- Launch instance:** A section with a 'Launch instance' button highlighted in orange. Below it, there is a 'Service health' section with an 'AWS Health Dashboard' link.

At the bottom of the console, there is a footer with 'Feedback', 'Looking for language selection? Find it in the new Unified Settings', and copyright information for Amazon Web Services, Inc. or its affiliates.

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.

The screenshot shows the AWS Management Console interface for launching an EC2 instance. The 'Name and tags' section has the name 'MyEC2Tutorial' entered. The 'Application and OS Images' section shows a search for AMIs, with 'Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type' selected. The 'Summary' panel on the right indicates 1 instance, t2.micro virtual server type, and 1 volume of 8 GiB. A 'Free tier' notification is visible, stating that the first year includes 750 hours of t2.micro (or t3.micro) usage.

3. Choose an instance type

Choose the t2.micro instance type.

The screenshot shows the AWS Management Console interface for configuring an EC2 instance. The top navigation bar includes the AWS logo, 'Services', a search bar, and the region 'N. Virginia'. The main content area is divided into several sections:

- Instance type info:** This section is highlighted with a red box. It shows the selected instance type as 't2.micro', which is 'Free tier eligible'. Below this, it lists specifications: 'Family: t2', '1 vCPU', and '1 GiB Memory'. Pricing information is also provided: 'On-Demand Linux pricing: 0.0116 USD per Hour' and 'On-Demand Windows pricing: 0.0162 USD per Hour'. A 'Compare instance types' link is visible to the right.
- Key pair (login) info:** This section is located below the instance type section. It contains a message: 'You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.'
- Summary:** This section is on the right side of the console. It shows 'Number of instances' set to '1', the instance type 't2.micro', 'Firewall (security group)' set to 'New security group', and 'Storage (volumes)' set to '1 volume(s) - 8 GiB'.

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**.

This screenshot shows the same AWS Management Console interface, but with the 'Key pair (login) info' section expanded. The 'Key pair name - required' dropdown menu is highlighted with a red box. The selected option is 'Proceed without a key pair (Not recommended)', with 'Default value' shown next to it. A 'Create new key pair' link is visible to the right of the dropdown.

Below the key pair section, the 'Network settings' section is visible, showing 'Network' set to 'vpc-060a30a3d0f67f67c', 'Subnet' set to 'No preference (Default subnet in any availability zone)', and 'Auto-assign public IP' set to 'Enable'. The 'Firewall (security groups)' section is also visible at the bottom of the main content area.

On the right side, the 'Summary' section is visible, and a 'Free tier' notification box is displayed at the bottom right. The notification box contains the following text: 'Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.' The notification box has a close button (X) in the top right corner. Below the notification box, there are 'Cancel' and 'Launch instance' buttons.

5. Keep default network and storage

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

The screenshot displays the AWS Management Console interface for configuring an EC2 instance. The top navigation bar includes the AWS logo, 'Services' menu, a search bar, and the region 'N. Virginia'. The main content area is divided into several sections:

- Network settings:** Includes options for Network (vpc-060a30a3d0f67f67c), Subnet (No preference), Auto-assign public IP (Enable), and Firewall (security groups). A warning message states: "Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only."
- Configure storage:** Shows 1x 8 GiB gp2 Root volume. A warning message states: "Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage."
- Advanced details:** A section for configuring advanced instance details.
- Summary:** Shows 1 instance, t2.micro instance type, and 1 volume(s) - 8 GiB storage. A warning message states: "Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet."

The 'Launch instance' button is highlighted in orange, indicating the next step in the process.

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**.

The screenshot displays the AWS Management Console interface for configuring an EC2 instance. The top navigation bar includes the AWS logo, 'Services', a search bar, and the region 'N. Virginia'. The main content area is divided into two panels: 'Advanced details' and 'Summary'.

Advanced details:

- Purchasing option:** Request Spot Instances. Request Spot Instances at the Spot price, capped at the On-Demand price.
- Domain join directory:** A dropdown menu set to 'Select'. A 'Create new directory' link is visible.
- IAM instance profile:** A dropdown menu set to 'EnablesEC2ToAccessSystemsManagerRole' (highlighted with a red box). A 'Create new IAM profile' link is visible.
- Hostname type:** A dropdown menu set to 'IP name'.
- DNS Hostname:** Three checkboxes: 'Enable IP name IPv4 (A record) DNS requests' (checked), 'Enable resource-based IPv4 (A record) DNS requests' (checked), and 'Enable resource-based IPv6 (AAAA record) DNS requests' (unchecked).
- Instance auto-recovery:** A dropdown menu set to 'Select'.

Summary:

- Number of instances:** A text input field containing '1'.
- t2.micro**
- Firewall (security group):** New security group.
- Storage (volumes):** 1 volume(s) - 8 GiB.

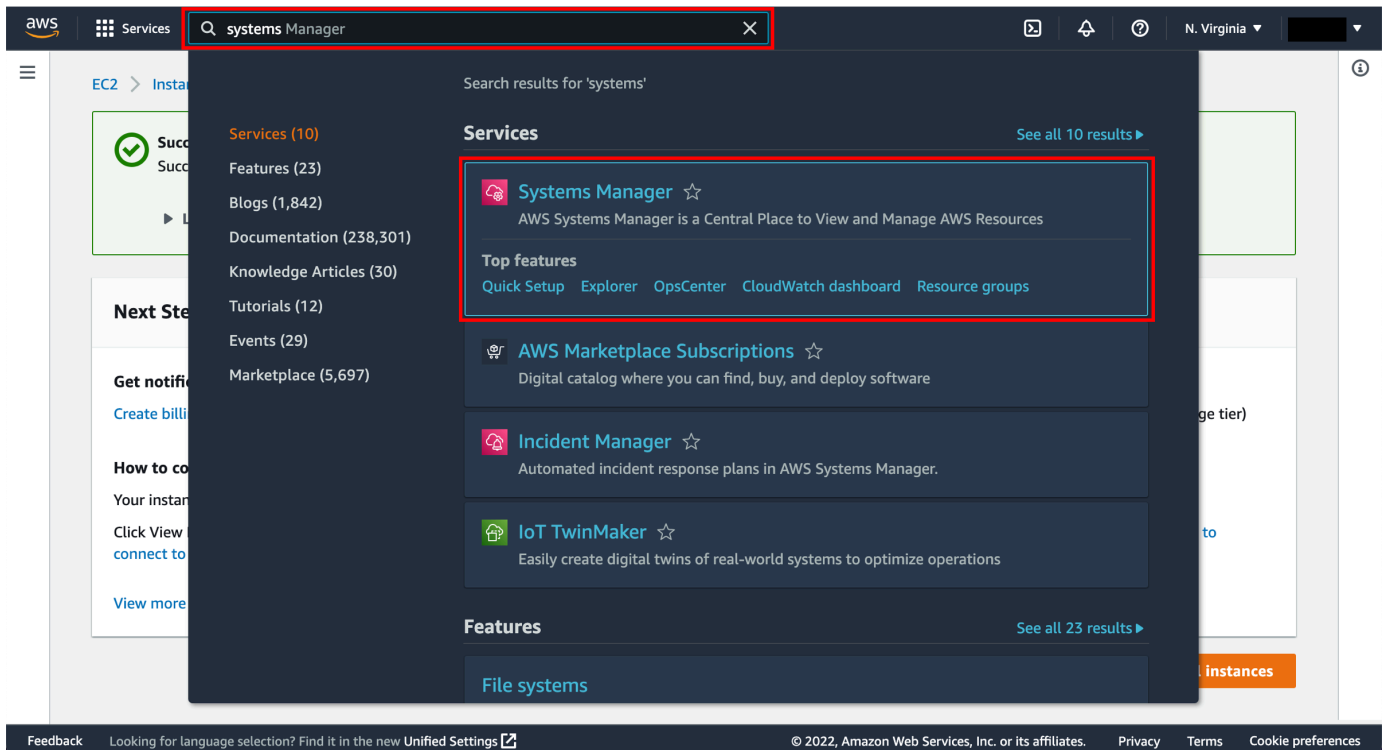
A 'Free tier' notification box is present, stating: 'Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.' Below this box are 'Cancel' and 'Launch instance' buttons (the latter is highlighted with a red box).

The footer contains 'Feedback', a link to 'Unified Settings', and copyright information: '© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences'.

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.

The screenshot shows the AWS Systems Manager Fleet Manager console. The 'Managed nodes' tab is selected, displaying a table of nodes. The first node is highlighted with a red box around its ID: 'i-0acdf9192e629ffb1'. The node is in a 'Running' state, named 'MyEC2Tutorial', and is an Amazon Linux EC2 instance.

Node ID	Node state	Node name	Platform ...	Operating Sys...	Source type	Source ID
i-0acdf9192e629ffb1	Running	MyEC2Tutorial	Linux	Amazon Linux	EC2 instance	-

4. Choose Run Command

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

The screenshot shows the Node detail page for 'i-0acdf9192e629ffb1'. The 'Node actions' dropdown menu is open, and the 'Execute run command' option is highlighted with a red box. The page also displays a 'Node overview' section with various details about the node.

Node ID	OS name
i-0acdf9192e629ffb1	Amazon Linux

Platform type	SSM Agent version
Linux	3.1.1188.0

Node type	IP address
t2.micro	172.31.83.42

Source type	Source ID
EC2 instance	-

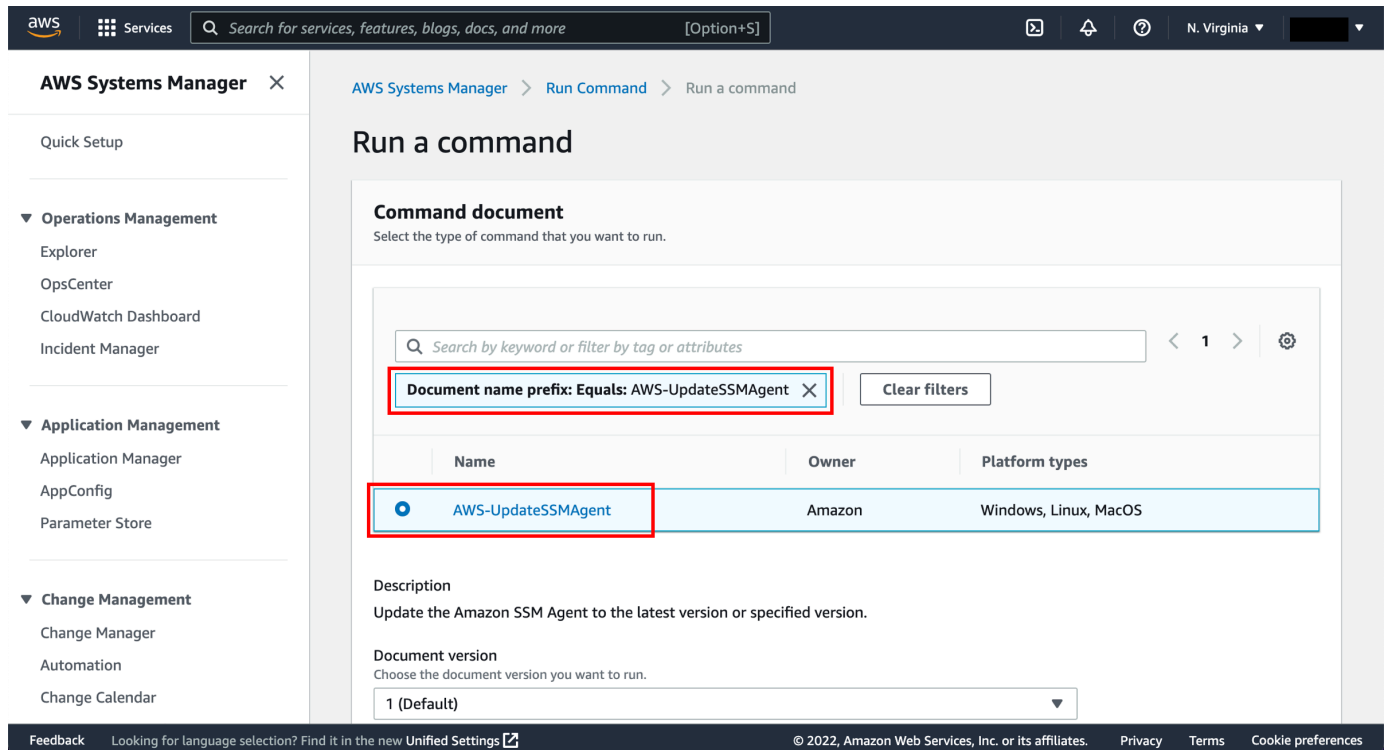
Node actions dropdown menu items:

- Connect
 - Start terminal session
 - Connect with Remote Desktop
- Admin tools
 - Execute run command
 - Patch node
- Node settings
 - Reset password
 - Change node profile
 - Deregister this managed node
- IAM role
 - arn:aws:iam::661972857966:instance-profile/EnablesEC2ToAccessSystemsManagerRole

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.



The screenshot shows the AWS Systems Manager console interface for running a command. The search bar is filtered with 'Document name prefix: Equals: AWS-UpdateSSMAgent'. The table below shows the 'AWS-UpdateSSMAgent' document selected with a radio button.

Name	Owner	Platform types
<input checked="" type="radio"/> AWS-UpdateSSMAgent	Amazon	Windows, Linux, MacOS

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**.

The screenshot shows the AWS Systems Manager console interface. On the left, the navigation menu is visible, with 'Run Command' highlighted in orange. The main content area is titled 'Targets' and contains the following sections:

- Targets:** Three radio button options: 'Specify instance tags', 'Choose instances manually' (selected), and 'Choose a resource group'.
- Instances:** A search bar and a table of instances. The table has columns: Node ID, Source type, Source ID, Name, and Ping status. One instance is selected and highlighted with a red box:

<input checked="" type="checkbox"/>	Node ID	Source type	Source ID	Name	Ping status
<input checked="" type="checkbox"/>	i-0acdf9192e629ffb1	AWS::EC2::Instance	i-0acdf9192e629ffb1	MyEC2Tutorial	Online
- Other parameters:** A 'Comment' text area, a 'Timeout (seconds)' input field with the value '600', and four expandable sections: 'Rate control', 'Output options', 'SNS notifications', and 'AWS command line interface command'.
- Buttons:** 'Cancel' and 'Run' buttons at the bottom right, with the 'Run' button highlighted in orange.

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.

Command ID: b06a32aa-8109-4ace-8598-ded4edafcd65

Cancel command Rerun Copy to new

Command status

Overall status	Detailed status	# targets	# completed	# error	# delivery timed out
Success	Success	1	1	0	0

Targets and outputs View output

Instance ID	Instance name	Status	Detailed Status	Start time	Finish time
i-0acdf9192e629ffb1	ip-172-31-83-42.ec2.internal	Success	Success	Thu, 14 Jul 2022 15:51:19 GMT	Thu, 14 Jul 2022 15:51:19 GMT

► **Command description**

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**.

Management & Governance

AWS Systems Manager Patch Manager

Manage patch compliance across the organization

Using Patch Manager, you can deploy patches simultaneously to applications and nodes across your organization. You can monitor patch compliance account by account.

Patch your instances

Patch instances without a schedule.

[Patch now](#)

Create schedules to patch instances.

[Configure patching](#)

Not ready to configure patching? Learn more about patching options by viewing the predefined patch baselines.

[View predefined patch baselines](#)

How it works

- 1 Use
- 2 Organize
- 3 Automate
- 4 Monitor

Use cases and blog posts

[Learn more](#)

[Latest blog post](#)

Feedback Looking for language selection? Find it in the new Unified Settings [\[?\]](#) © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

2. Choose an instance

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

AWS Systems Manager > Fleet Manager

Fleet Manager [Info](#)

[Managed nodes](#) [Settings](#)

Managed nodes [Refresh](#) [Download report New](#) [Node actions](#) [Account management](#)

[1](#) [Settings](#)

Total: 1 node Last fetched at: 8:14 AM

<input type="checkbox"/>	Node ID	Node state	Node name	Platform ...	Operating Sys...	Source type	Source ID
<input type="checkbox"/>	i-0acdf9192e629ffb1	Running	MyEC2Tutorial	Linux	Amazon Linux	EC2 instance	-

Feedback Looking for language selection? Find it in the new Unified Settings [\[?\]](#) © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

3. Choose Run Command

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

The screenshot displays the AWS Systems Manager console interface. At the top, the navigation bar shows 'AWS Systems Manager > Fleet Manager > Node ID: i-0acdf9192e629ffb1'. The main content area is titled 'Node ID: i-0acdf9192e629ffb1' with a 'Running' status indicator. On the left, a 'Tools' sidebar lists 'Node overview', 'File system', 'Performance counters', 'Processes', and 'Users and groups'. The 'Node overview' section contains a table of node details:

Node overview	
Node ID	i-0acdf9192e629ffb1
OS name	Amazon Linux
Platform type	Linux
Node type	t2.micro
Source type	EC2 instance
SSM Agent version	3.1.1188.0
IP address	172.31.83.42
Source ID	-

Below the table are tabs for 'Tags', 'Inventory', 'Associations', 'Patch', and 'Configuration compliance'. A 'Node actions' dropdown menu is open on the right, showing options like 'Start terminal session', 'Admin tools', and 'Node settings'. The 'Execute run command' option is highlighted with a red box. The footer contains 'Feedback', 'Looking for language selection? Find it in the new Unified Settings', '© 2022, Amazon Web Services, Inc. or its affiliates.', 'Privacy', 'Terms', and 'Cookie preferences'.

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.

The screenshot shows the AWS Systems Manager console. The left sidebar contains navigation options under 'Operations Management', 'Application Management', and 'Change Management'. The main content area is titled 'Run a command' and 'Run a command document'. A search bar is present with the filter 'Document name prefix: Equals: AWS-RunShellScript'. Below the search bar is a table with the following data:

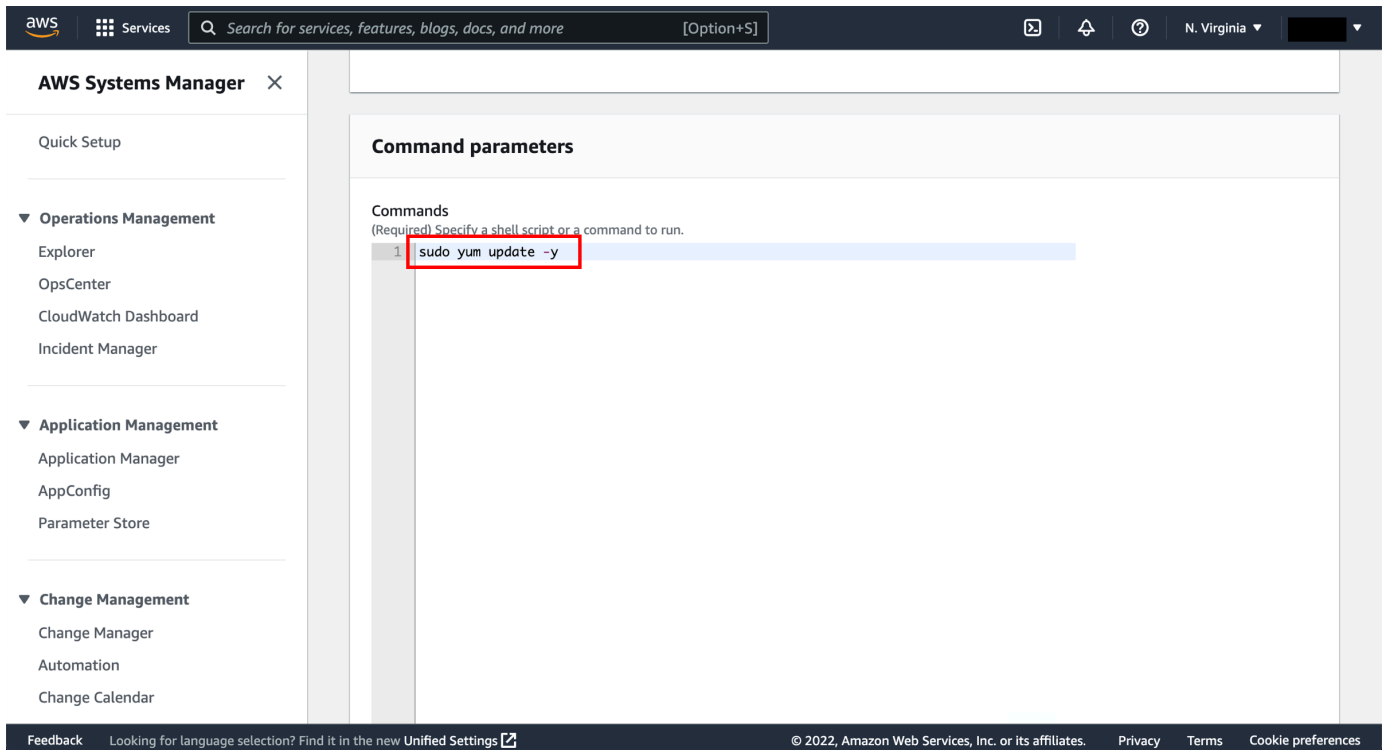
Name	Owner	Platform types
<input checked="" type="radio"/> AWS-RunShellScript	Amazon	Linux, MacOS

Below the table, there is a 'Description' section with the text 'Run a shell script or specify the commands to run.' and a 'Document version' dropdown menu set to '1 (Default)'.

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
```



The screenshot displays the AWS Systems Manager console interface. At the top, there is a navigation bar with the AWS logo, a search bar, and the region 'N. Virginia'. The main content area is titled 'Command parameters' and includes a 'Commands' section with the instruction '(Required) Specify a shell script or a command to run.' Below this, a text input field contains the command 'sudo yum update -y', which is highlighted with a red rectangular box. The left-hand navigation pane lists various services under three main categories: 'Operations Management' (Explorer, OpsCenter, CloudWatch Dashboard, Incident Manager), 'Application Management' (Application Manager, AppConfig, Parameter Store), and 'Change Management' (Change Manager, Automation, Change Calendar). The footer contains a feedback link, a language selection note, and copyright information for Amazon Web Services, Inc. or its affiliates, along with links for Privacy, Terms, and Cookie preferences.

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**.

Targets

Targets
Choose a method for selecting targets.

Specify instance tags
Specify one or more tag key-value pairs to select instances that share those tags.

Choose instances manually
Manually select the instances you want to register as targets.

Choose a resource group
Choose a resource group that includes the resources you want to target.

i-0acdf9192e629ffb1 X

Instances

Search

<input checked="" type="checkbox"/>	Node ID	Source type	Source ID	Name	Ping status
<input checked="" type="checkbox"/>	i-0acdf9192e629ffb1	AWS::EC2::Instance	i-0acdf9192e629ffb1	MyEC2Tutorial	Online

Other parameters

Comment
(Optional) Type a note about the command

Timeout (seconds)
Specify the timeout for command in seconds

600

► **Rate control**

► **Output options**

► **SNS notifications**

► **AWS command line interface command**

Cancel 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.

Command ID: b63806ba-8431-4ead-839a-7c7eebb670ab

Cancel command Rerun Copy to new

Command status

Overall status	Detailed status	# targets	# completed	# error	# delivery timed out
Success	Success	1	1	0	0

Targets and outputs View output

Instance ID	Instance name	Status	Detailed Status	Start time	Finish time
i-0acdf9192e629ffb1	ip-172-31-83-42.ec2.internal	Success	Success	Thu, 14 Jul 2022 16:11:36 GMT	Thu, 14 Jul 2022 16:11:37 GMT

► Command description

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.

Output on: i-0acdf9192e629ffb1

Command ID: b63806ba-8431-4ead-839a-7c7eebb670ab

Output on i-0acdf9192e629ffb1

Step 1 - Command description and status

Status	Detailed status	Response code
Success	Success	0
Step name	Start time	Finish time
aws:runShellScript	Thu, 14 Jul 2022 16:11:36 GMT	Thu, 14 Jul 2022 16:11:37 GMT

Output

The command output displays a maximum of 48,000 characters. You can view the complete command output in either Amazon S3 or CloudWatch Logs, if you specify an S3 bucket or a logs group when you run the command.

```
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
No packages marked for update
```

Copy Download

Step 5: Update the Systems Manager Agent

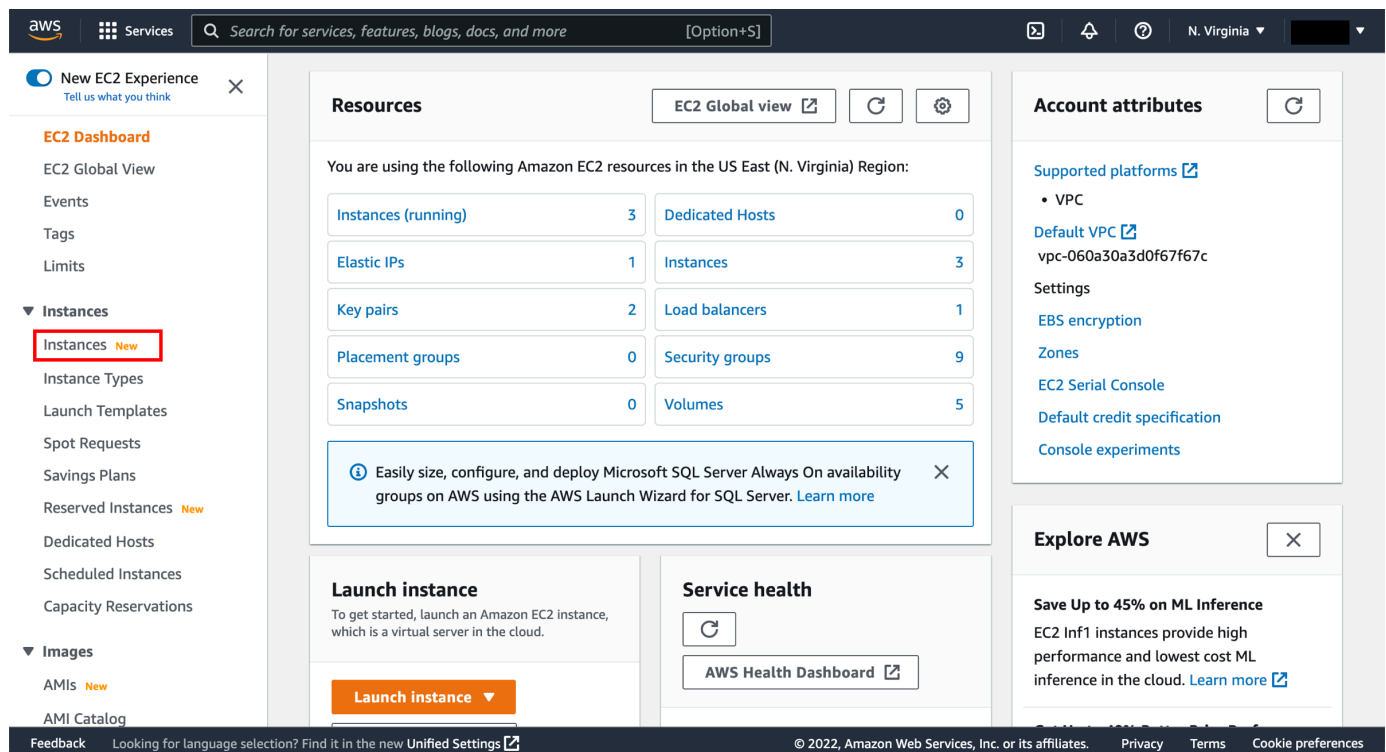
In this step, you will terminate your Systems Manager and EC2 related resources.

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**.



The screenshot shows the Amazon EC2 console interface. The top navigation bar includes the AWS logo, 'Services', a search bar, and the region 'N. Virginia'. The left navigation pane shows the 'Instances' section expanded, with 'Instances' highlighted in a red box. The main content area displays a 'Resources' section with a table of EC2 resources in the US East (N. Virginia) region:

Resources	
Instances (running)	3
Elastic IPs	1
Key pairs	2
Placement groups	0
Snapshots	0
Dedicated Hosts	0
Instances	3
Load balancers	1
Security groups	9
Volumes	5

Below the resources table, there is a 'Launch instance' section with a 'Launch instance' button and a 'Service health' section with an 'AWS Health Dashboard' link. The footer of the console shows '© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences'.

2. Terminate your instance

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

The screenshot displays the AWS Management Console interface for EC2 instances. The main content area shows a table of instances with columns for Name, Instance ID, and Instance state. The instance 'MyEC2Tutorial' (ID: i-0acdf9192e629ffb1) is highlighted in blue and has a 'Running' status. A context menu is open over this instance, with the 'Terminate instance' option selected. Below the instance list, there is a 'Monitoring' section with four charts: 'CPU utilization (%)', 'Status check failed (a...)', 'Status check failed (i...)', and 'Status check failed (s...)'.

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.