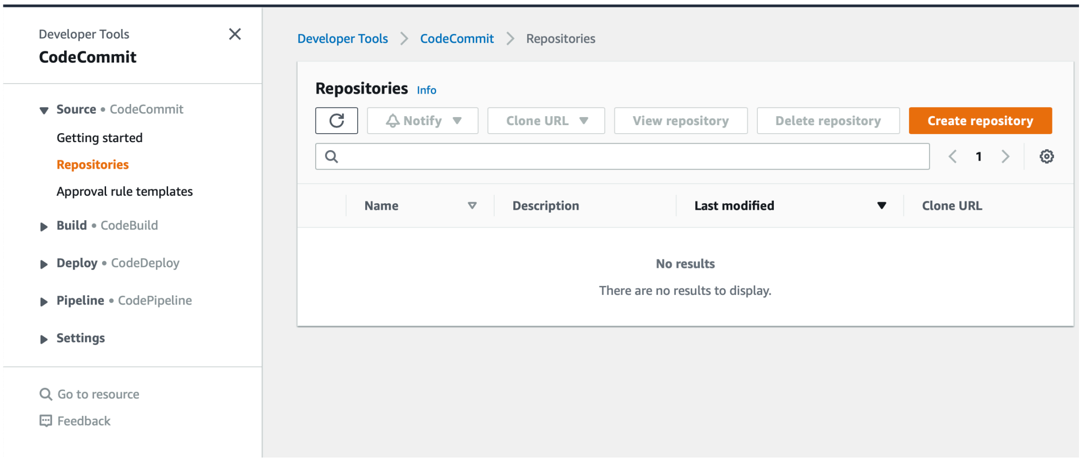
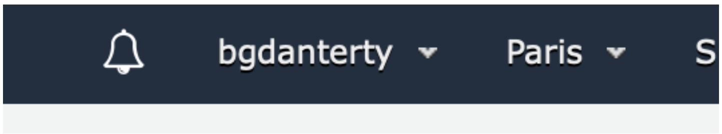
1. Open the CodeCommit console at https://console.aws.amazon.com/codecommit/



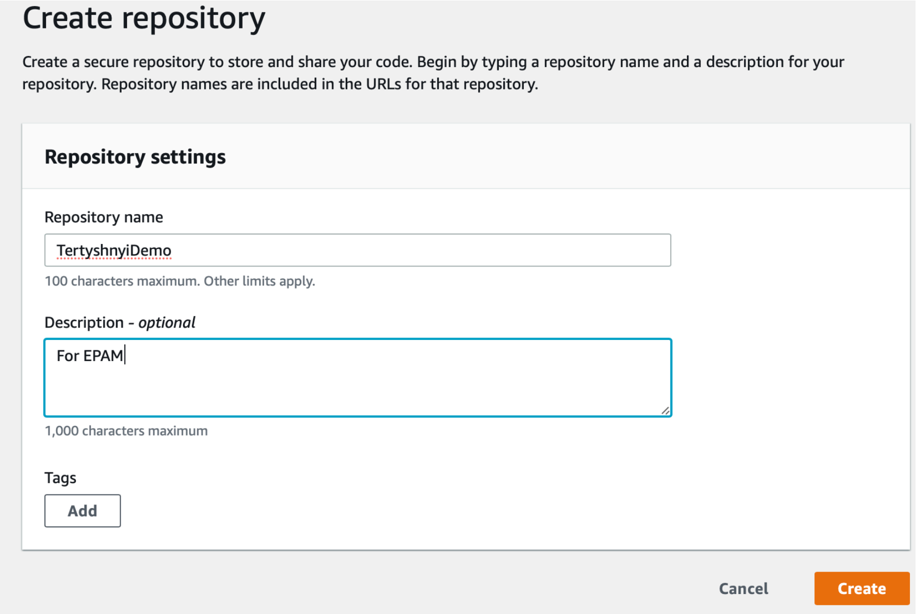
No results

There are no results to display.

2. In the Region selector, choose the AWS Region where you want to create the repository and pipeline. For more information, see AWS Regions and Endpoints.



3. On the Repositories page, choose Create repository. On the **Create repository** page, in **Repository name**, enter a name for your repository (for example, `MyDemoRepo`). Choose **Create.**



Create repository

Create a secure repository to store and share your code. Begin by typing a repository name and a description for your repository. Repository names are included in the URLs for that repository.

Repository settings

Repository name: TertyshnyiDemo

100 characters maximum. Other limits apply.

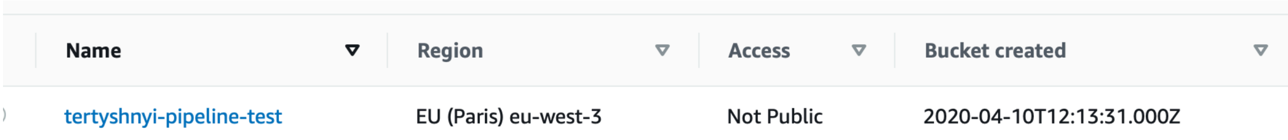
Description - *optional*: For EPAM

1,000 characters maximum

Tags

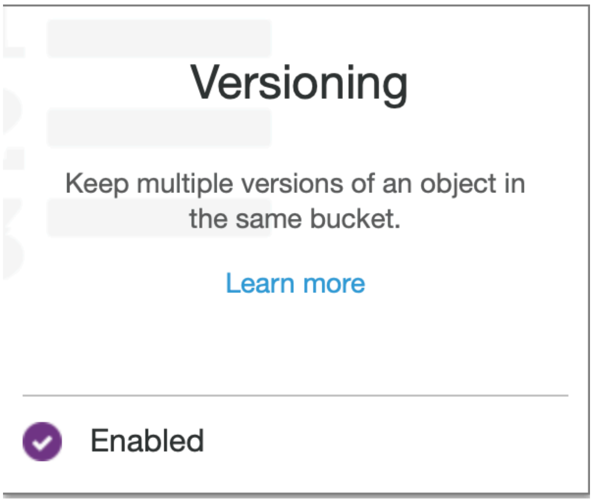
4. With your new repository open in the console, choose **Clone URL** on the top right of the page, and then choose **Clone SSH**. The address to clone your Git repository is copied to your clipboard. In your terminal or command line, navigate to a local directory where you'd like your local repository to be stored. In this tutorial, we use `/tmp`. Run the following command to clone the repository, replacing the SSH address with the one you copied in the previous step. This command creates a directory called `MyDemoRepo`. You copy a sample application to this directory. ;://git-codecommit.us-west-2.amazonaws.com/v1/repos/MyDemoRepo

1. Choose **Create bucket**. In Bucket name, enter a name for your bucket (for example, `awscodepipeline-demobucket-example-date`)



Name	Region	Access	Bucket created
tertyshnyi-pipeline-test	EU (Paris) eu-west-3	Not Public	2020-04-10T12:13:31.000Z

2. After the bucket is created, a success banner displays. Choose **Go to bucket details**. On the **Properties** tab, choose **Versioning**. Choose **Enable versioning**, and then choose **Save**.



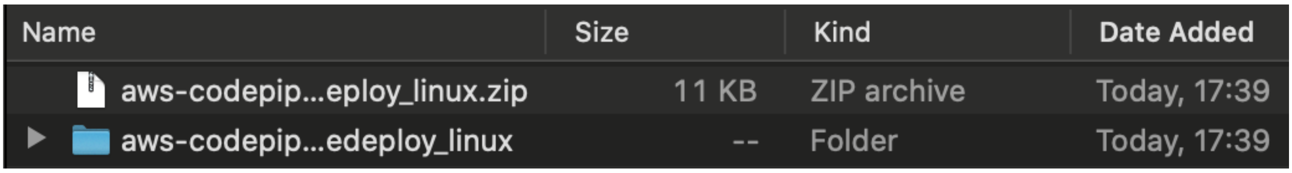
Versioning

Keep multiple versions of an object in the same bucket.

Learn more

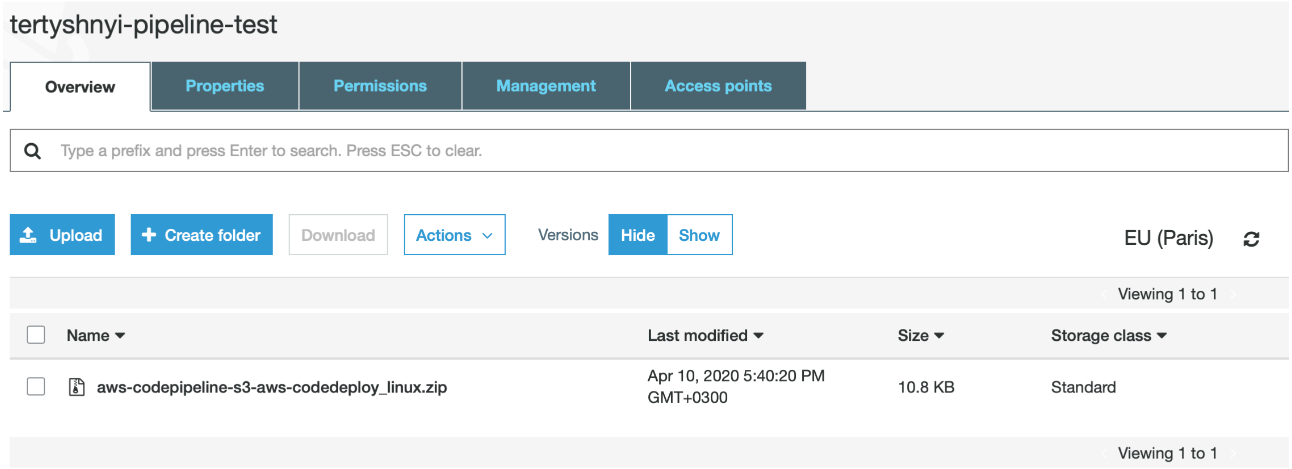
✓ Enabled

3. Next, download a sample from a GitHub repository and save it into a folder or directory on your local computer.



Name	Size	Kind	Date Added
aws-codepip...eploy_linux.zip	11 KB	ZIP archive	Today, 17:39
▶ aws-codepip...edeploy_linux	--	Folder	Today, 17:39

4. In the Amazon S3 console, for your bucket, upload the file:
   - Choose **Upload**.
   - Drag and drop the file or choose **Add files** and browse for the file.
   - Choose **Upload**.



tertyshnyi-pipeline-test

□	Name	Last modified	Size	Storage class
□	aws-codepipeline-s3-aws-codedeploy_linux.zip	Apr 10, 2020 5:40:20 PM GMT+0300	10.8 KB	Standard

5. From the console dashboard, choose **Launch instance**, and select **Launch instance** from the options that pop up

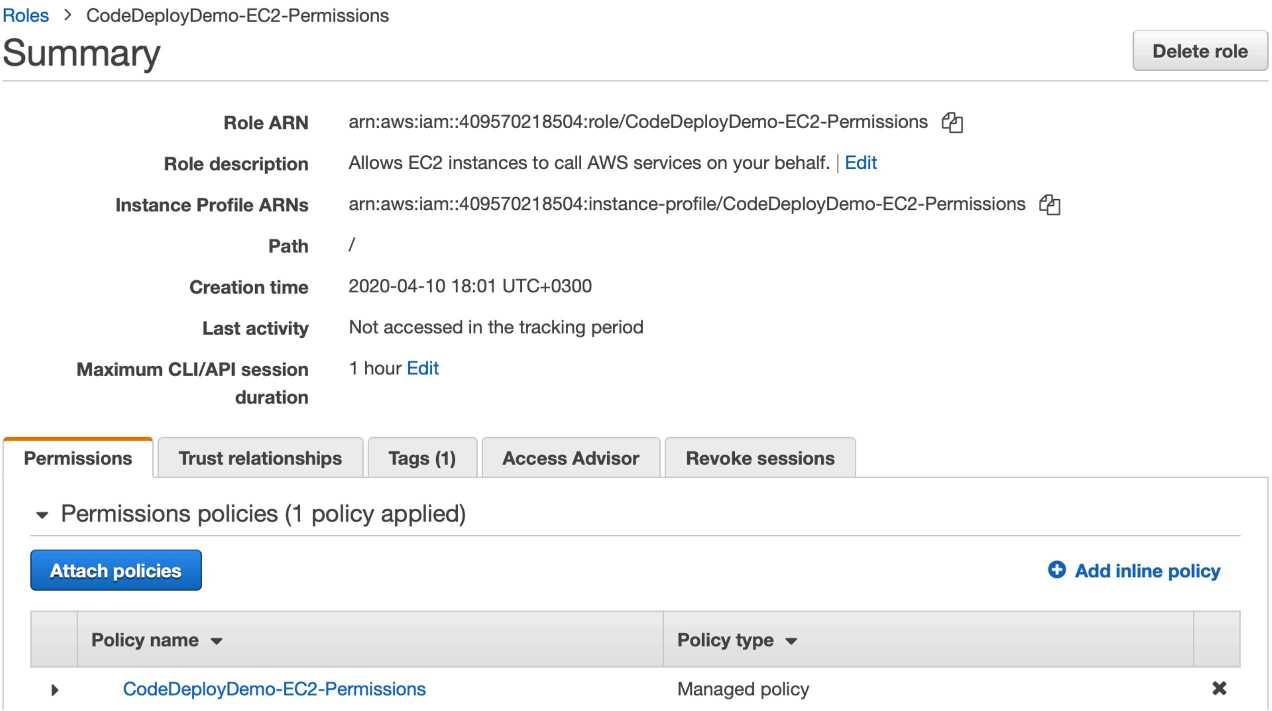
   On the **Step 1: Choose an Amazon Machine Image (AMI)** page, locate the **Microsoft Windows Server 2019 Base** option, and then choose **Select**. (This AMI is labeled "Free tier eligible" and can be found at the top of the list.)

   On the **Step 2: Choose an Instance Type** page, choose the free tier eligible `t2.micro` type as the hardware configuration for your instance, and then choose **Next: Configure Instance Details**.

   On the **Step 3: Configure Instance Details** page, do the following:

   - In **Number of instances**, enter `2`.
   - In **Auto-assign Public IP**, choose **Enable**.

- In **IAM role**, choose an IAM role that has been configured for use as an IAM instance profile for use with CodeDeploy.

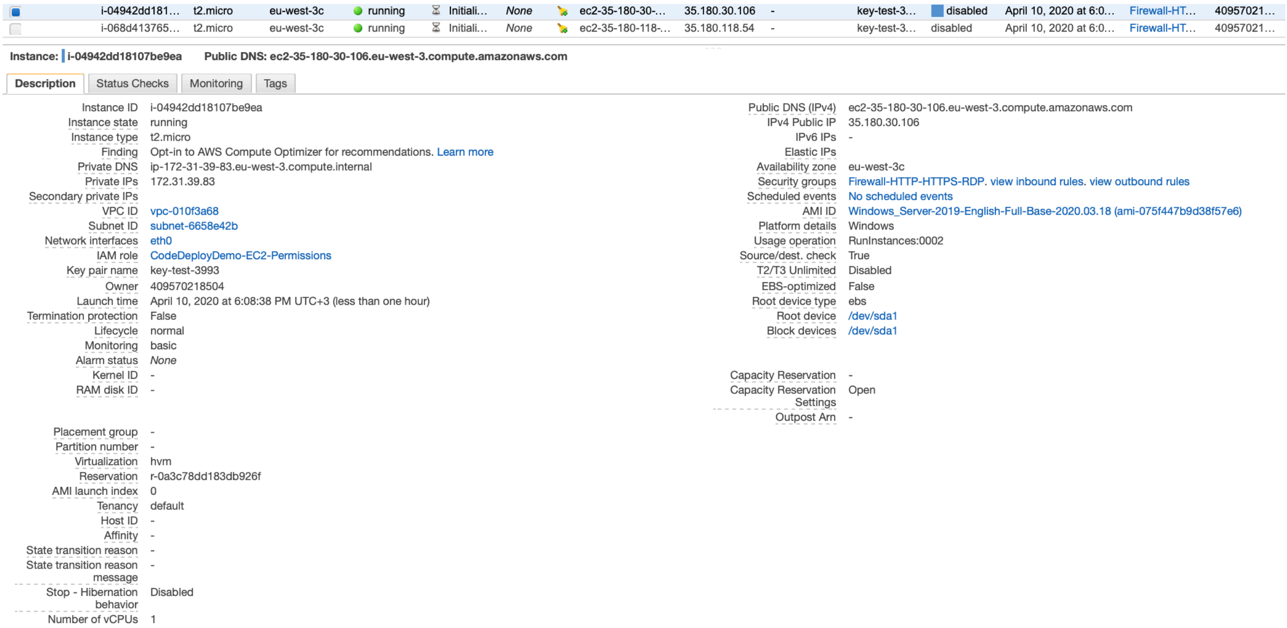


Summary

Role ARN	arn:aws:iam::409570218504:role/CodeDeployDemo-EC2-Permissions
Role description	Allows EC2 instances to call AWS services on your behalf. | Edit
Instance Profile ARNs	arn:aws:iam::409570218504:instance-profile/CodeDeployDemo-EC2-Permissions
Path	/
Creation time	2020-04-10 18:01 UTC+0300
Last activity	Not accessed in the tracking period
Maximum CLI/API session duration	1 hour Edit

Permissions policies (1 policy applied)

Policy name	Policy type	
CodeDeployDemo-EC2-Permissions	Managed policy	✖



■	i-04942dd181…	t2.micro	eu-west-3c	● running	Initiali…	None	ec2-35-180-30-…	35.180.30.106	-	key-test-3…	disabled	April 10, 2020 at 6:0…	Firewall-HT…	40957021…
□	i-068d413765…	t2.micro	eu-west-3c	● running	Initiali…	None	ec2-35-180-118-…	35.180.118.54	-	key-test-3…	disabled	April 10, 2020 at 6:0…	Firewall-HT…	40957021…

Instance: **i-04942dd18107be9ea** **Public DNS: ec2-35-180-30-106.eu-west-3.compute.amazonaws.com**

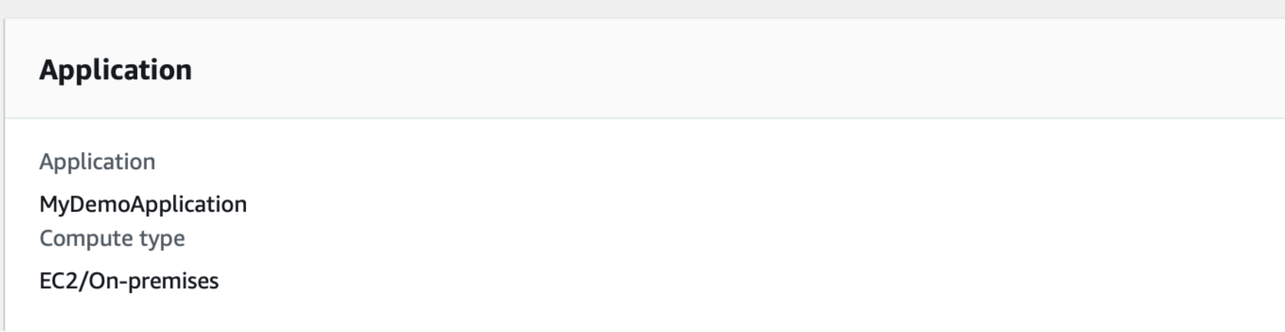
Instance ID	i-04942dd18107be9ea
Instance state	running
Instance type	t2.micro
Finding	Opt-in to AWS Compute Optimizer for recommendations. Learn more
Private DNS	ip-172-31-39-83.eu-west-3.compute.internal
Private IPs	172.31.39.83
Secondary private IPs	
VPC ID	vpc-010f3a68
Subnet ID	subnet-6658e42b
Network interfaces	eth0
IAM role	CodeDeployDemo-EC2-Permissions
Key pair name	key-test-3993
Owner	409570218504
Launch time	April 10, 2020 at 6:08:38 PM UTC+3 (less than one hour)
Termination protection	False
Lifecycle	normal
Monitoring	basic
Alarm status	*None*
Kernel ID	-
RAM disk ID	-
Placement group	-
Partition number	-
Virtualization	hvm
Reservation	r-0a3c78dd183db926f
AMI launch index	0
Tenancy	default
Host ID	-
Affinity	-
State transition reason	-
State transition reason message	-
Stop - Hibernation behavior	Disabled
Number of vCPUs	1

Public DNS (IPv4)	ec2-35-180-30-106.eu-west-3.compute.amazonaws.com
IPv4 Public IP	35.180.30.106
IPv6 IPs	-
Elastic IPs	
Availability zone	eu-west-3c
Security groups	Firewall-HTTP-HTTPS-RDP. view inbound rules. view outbound rules
Scheduled events	No scheduled events
AMI ID	Windows_Server-2019-English-Full-Base-2020.03.18 (ami-075f447b9d38f57e6)
Platform details	Windows
Usage operation	RunInstances:0002
Source/dest. check	True
T2/T3 Unlimited	Disabled
EBS-optimized	False
Root device type	ebs
Root device	/dev/sda1
Block devices	/dev/sda1
Capacity Reservation	-
Capacity Reservation Settings	Open
Outpost Arn	-

5. To create an application in CodeDeploy

1. Open the CodeDeploy console at https://console.aws.amazon.com/codedeploy.
2. If the **Applications** page does not appear, on the AWS CodeDeploy menu, choose **Applications**.

3. Choose **Create application**.
4. Leave **Custom application** selected. In **Application name**, enter `MyDemoApplication`.
5. In **Compute Platform**, choose **EC2/On-premises**.
6. Choose **Create application**.



Application

Application

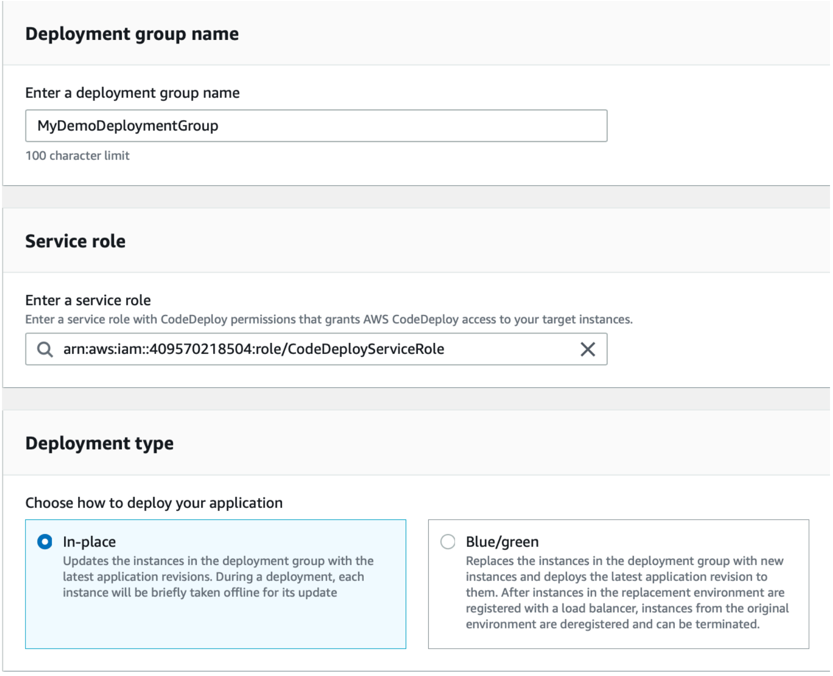
MyDemoApplication

Compute type

EC2/On-premises

6. To create a deployment group in CodeDeploy

1. On the page that displays your application, choose **Create deployment group**.
2. In **Deployment group name**, enter `MyDemoDeploymentGroup`.
3. In **Service Role**, choose a service role that trusts AWS CodeDeploy with, at minimum, the trust and permissions described in Create a Service Role for CodeDeploy. To get the service role ARN, see Get the Service Role ARN (Console).
4. Under **Deployment type**, choose **In-place**.
5. Under **Environment configuration**, choose **Amazon EC2 Instances**. Choose **Name** in the **Key** field, and in the **Value** field, enter `MyCodePipelineDemo`.



Deployment group name

Enter a deployment group name

MyDemoDeploymentGroup

100 character limit

Service role

Enter a service role

Enter a service role with CodeDeploy permissions that grants AWS CodeDeploy access to your target instances.

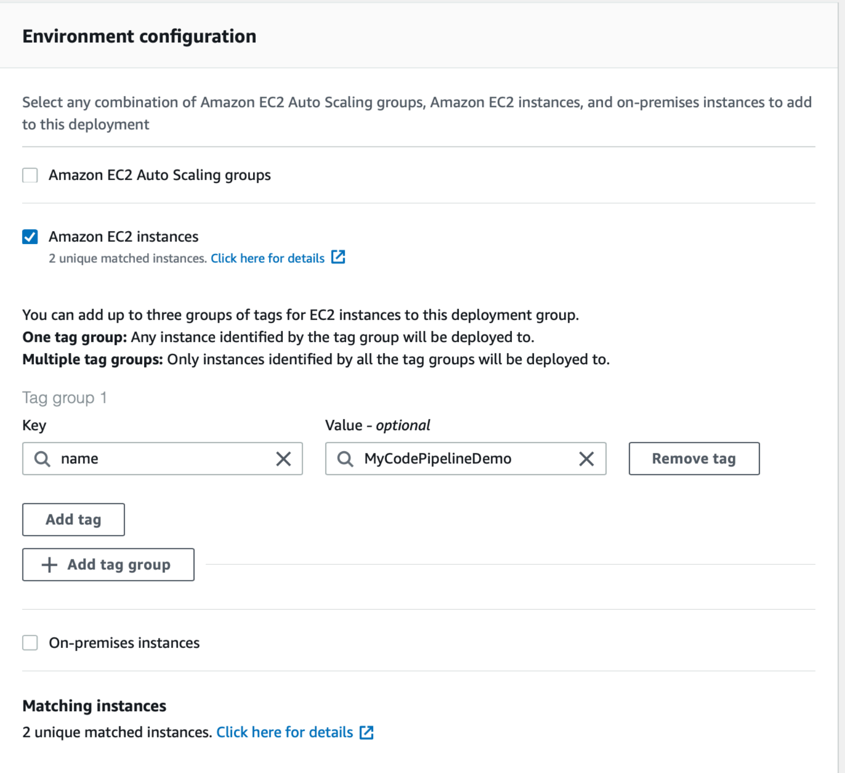
arn:aws:iam::409570218504:role/CodeDeployServiceRole

Deployment type

Choose how to deploy your application

● In-place
Updates the instances in the deployment group with the latest application revisions. During a deployment, each instance will be briefly taken offline for its update

○ Blue/green
Replaces the instances in the deployment group with new instances and deploys the latest application revision to them. After instances in the replacement environment are registered with a load balancer, instances from the original environment are deregistered and can be terminated.



Environment configuration

Select any combination of Amazon EC2 Auto Scaling groups, Amazon EC2 instances, and on-premises instances to add to this deployment

☐ Amazon EC2 Auto Scaling groups

☑ Amazon EC2 instances
2 unique matched instances. Click here for details

You can add up to three groups of tags for EC2 instances to this deployment group.
One tag group: Any instance identified by the tag group will be deployed to.
Multiple tag groups: Only instances identified by all the tag groups will be deployed to.

Tag group 1

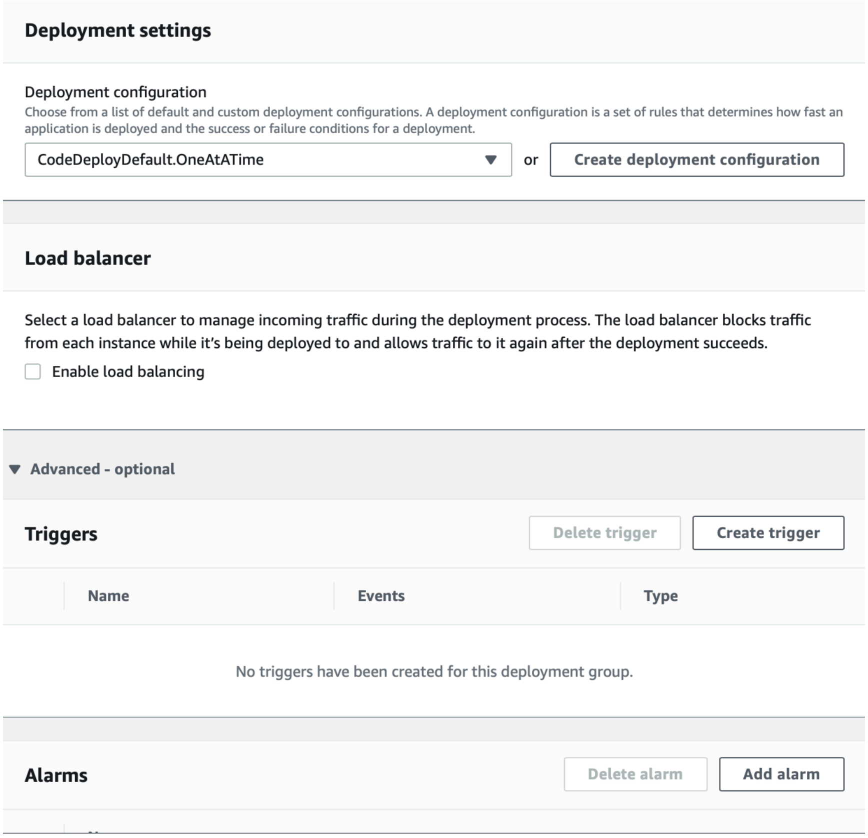
Key: name

Value - *optional*: MyCodePipelineDemo

☐ On-premises instances

Matching instances
2 unique matched instances. Click here for details

6. Under **Deployment configuration**, choose `CodeDeployDefault.OneAtaTime`.
7. Under **Load Balancer**, clear **Enable load balancing**. You do not need to set up a load balancer or choose a target group for this example.
8. Expand the **Advanced** section. Under **Alarms**, choose **Ignore alarm configuration**.
9. Choose **Create deployment group**.



Deployment settings

Deployment configuration
Choose from a list of default and custom deployment configurations. A deployment configuration is a set of rules that determines how fast an application is deployed and the success or failure conditions for a deployment.

CodeDeployDefault.OneAtATime ▼ or

Load balancer

Select a load balancer to manage incoming traffic during the deployment process. The load balancer blocks traffic from each instance while it's being deployed to and allows traffic to it again after the deployment succeeds.

☐ Enable load balancing

▼ Advanced - optional

Triggers

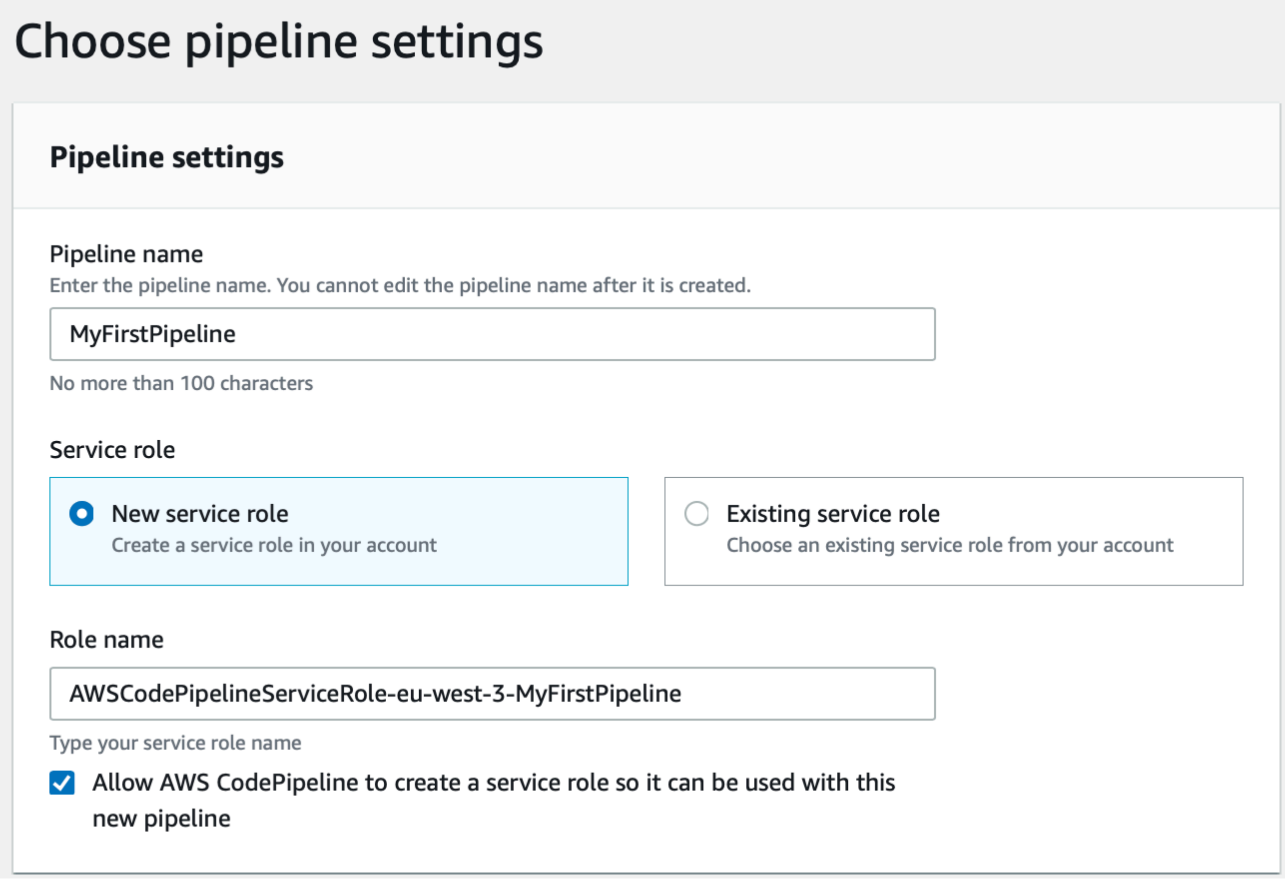
Name	Events	Type

No triggers have been created for this deployment group.

Alarms

To create a CodePipeline automated release process

1. Sign in to the AWS Management Console and open the CodePipeline console at http://console.aws.amazon.com/codesuite/codepipeline/home.
2. On the **Welcome** page, **Getting started** page, or the **Pipelines** page, choose **Create pipeline**.
3. In **Step 1: Choose pipeline settings**, in **Pipeline name**, enter `MyFirstPipeline`.
4. **In Service role, Choose Existing service role to use a service role already created in IAM. In Role name, choose your service role from the list.**



Choose pipeline settings

Pipeline settings

Pipeline name

Enter the pipeline name. You cannot edit the pipeline name after it is created.

MyFirstPipeline

No more than 100 characters

Service role

● New service role
Create a service role in your account

○ Existing service role
Choose an existing service role from your account

Role name

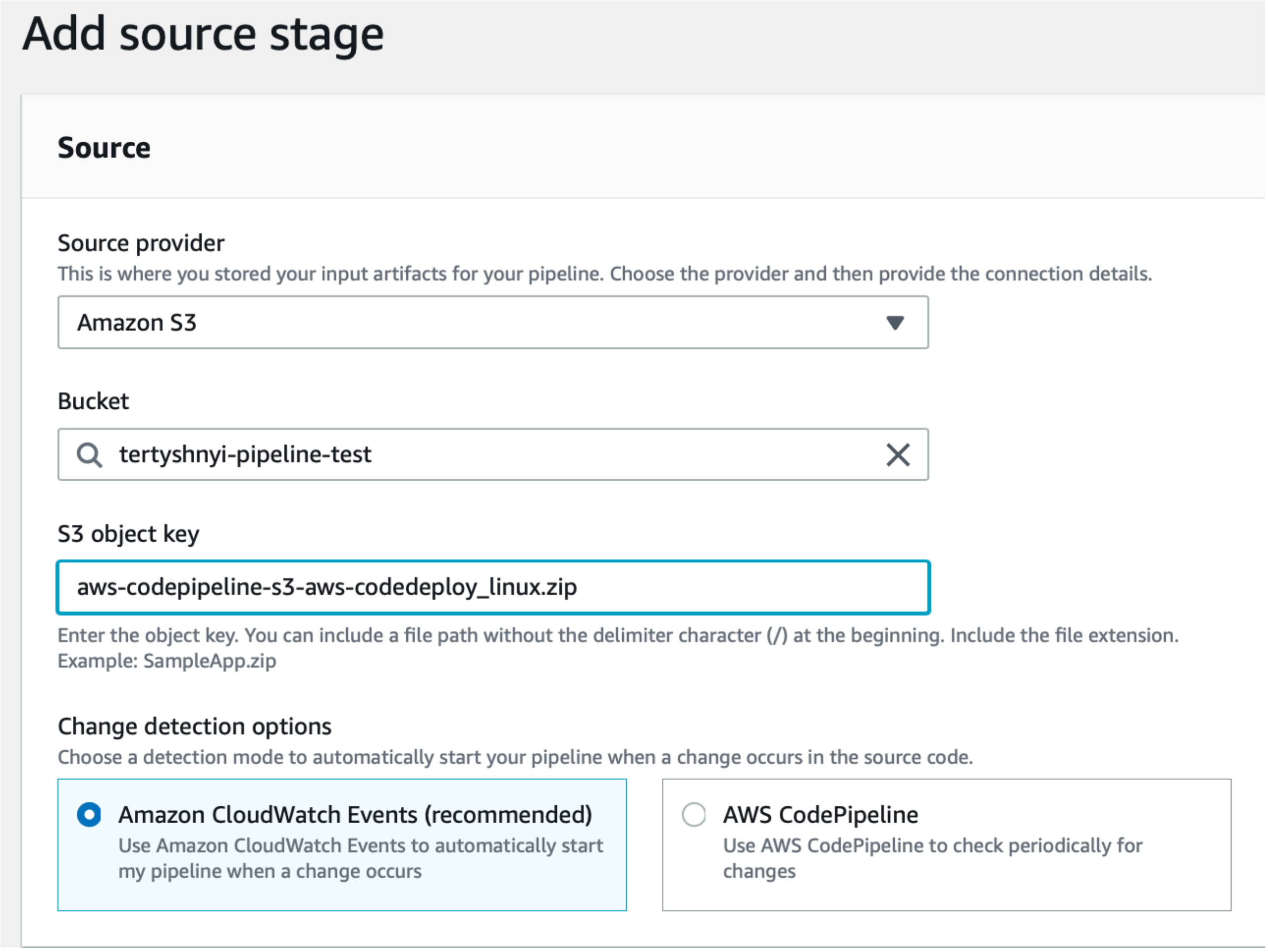
AWSCodePipelineServiceRole-eu-west-3-MyFirstPipeline

Type your service role name

☑ Allow AWS CodePipeline to create a service role so it can be used with this new pipeline

5. Leave the settings under **Advanced settings** at their defaults, and then choose **Next**.
6. In **Step 2: Add source stage**, in **Source provider**, choose **Amazon S3**. In **Bucket**, enter the name of the S3 bucket you created in Step 1: Create an S3 Bucket for Your Application. In **S3 object key**, enter the object key with or without a file path, and remember to include the file extension. For example, if

you named your downloaded ZIP file `SampleApp_Linux.zip`, enter the sample file name as shown in this example: SampleApp_Linux.zip



Add source stage

Source

Source provider

This is where you stored your input artifacts for your pipeline. Choose the provider and then provide the connection details.

Amazon S3 ▼

Bucket

tertyshnyi-pipeline-test

S3 object key

aws-codepipeline-s3-aws-codedeploy_linux.zip

Enter the object key. You can include a file path without the delimiter character (/) at the beginning. Include the file extension. Example: SampleApp.zip

Change detection options

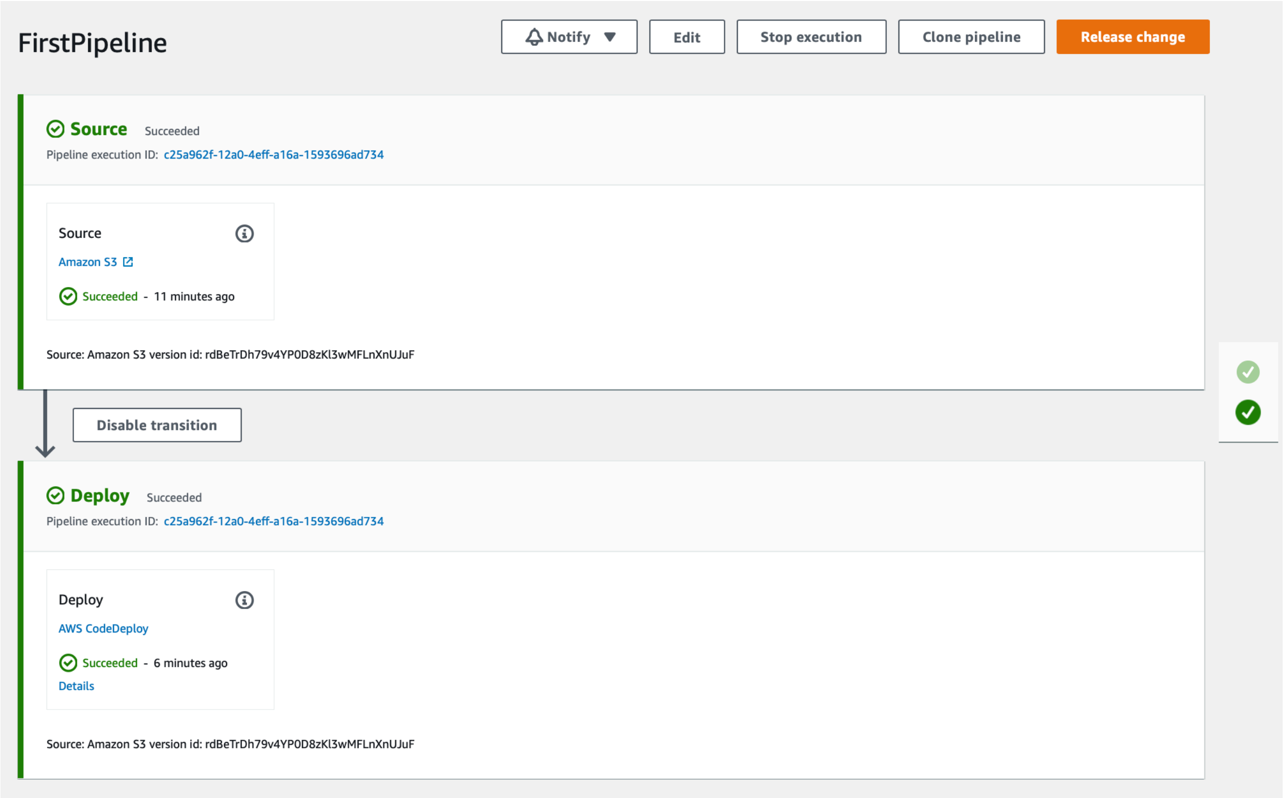
Choose a detection mode to automatically start your pipeline when a change occurs in the source code.

● Amazon CloudWatch Events (recommended)
Use Amazon CloudWatch Events to automatically start my pipeline when a change occurs

○ AWS CodePipeline
Use AWS CodePipeline to check periodically for changes

7. In **Step 3: Add build stage**, choose **Skip build stage**, and then accept the warning message by choosing **Skip** again. Choose **Next**.
8. In **Step 4: Add deploy stage**, in **Deploy provider**, choose **AWS CodeDeploy**. The **Region** field defaults to the same AWS Region as your pipeline. In **Application name**, enter `MyDemoApplication`, or choose the **Refresh** button, and then choose the application name from the list. In **Deployment group**, enter `CodePipelineDemoFleet`, or choose it from the list, and then choose **Next**.

9. In **Step 5: Review**, review the information, and then choose **Create pipeline**.
10. The pipeline starts to run. You can view progress and success and failure messages as the CodePipeline sample deploys a webpage to each of the Amazon EC2 instances in the CodeDeploy deployment.



FirstPipeline

Source Succeeded
Pipeline execution ID: c25a962f-12a0-4eff-a16a-1593696ad734

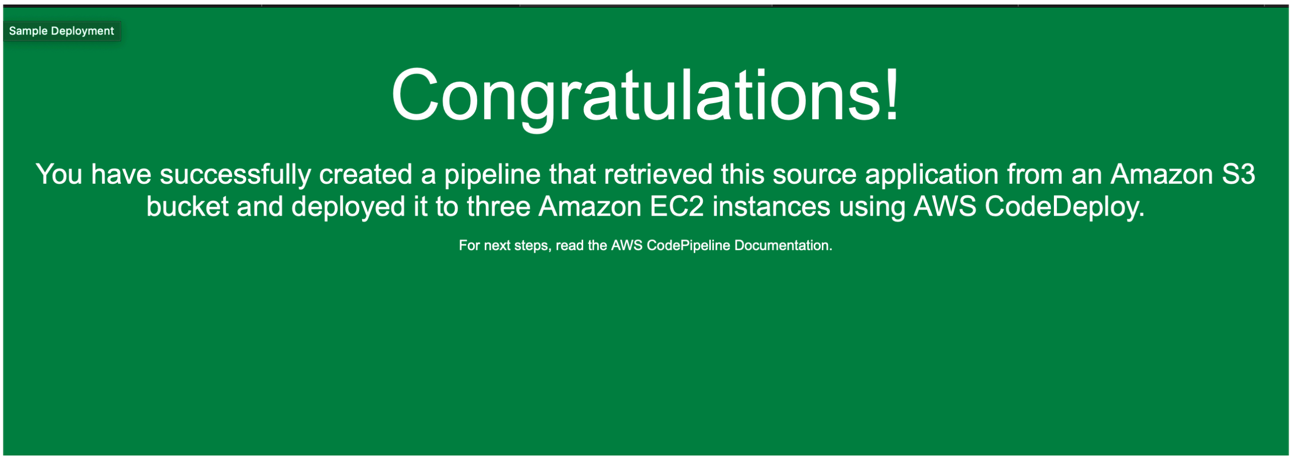
Source
Amazon S3
Succeeded - 11 minutes ago

Source: Amazon S3 version id: rdBeTrDh79v4YP0D8zKl3wMFLnXnUJuF

Deploy Succeeded
Pipeline execution ID: c25a962f-12a0-4eff-a16a-1593696ad734

Deploy
AWS CodeDeploy
Succeeded - 6 minutes ago
Details

Source: Amazon S3 version id: rdBeTrDh79v4YP0D8zKl3wMFLnXnUJuF



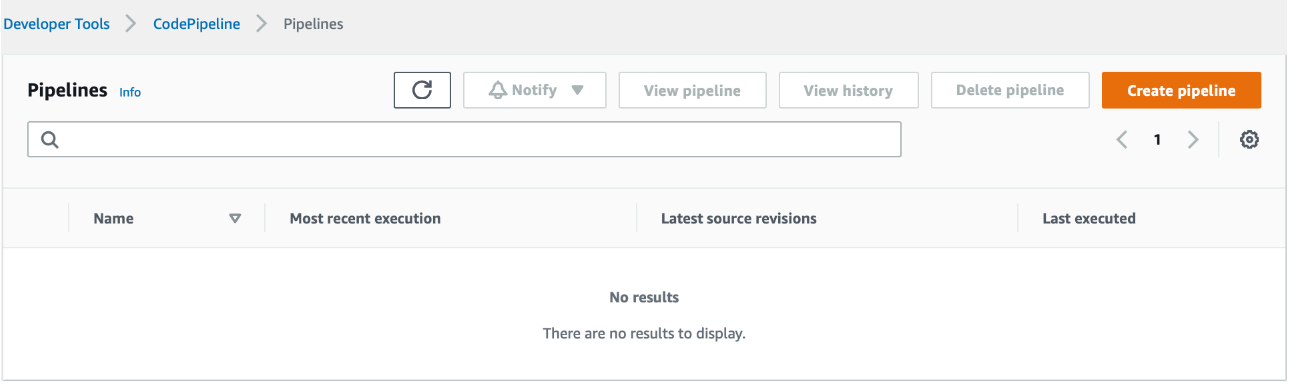
Congratulations!

You have successfully created a pipeline that retrieved this source application from an Amazon S3 bucket and deployed it to three Amazon EC2 instances using AWS CodeDeploy.

For next steps, read the AWS CodePipeline Documentation.

To clean up the resources used in this tutorial

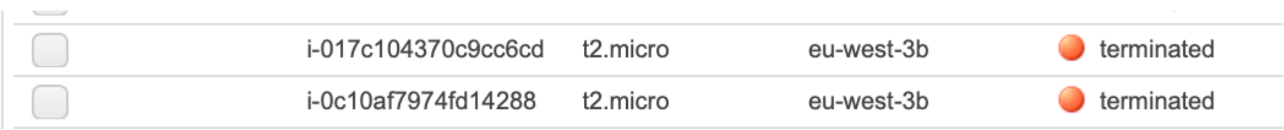
1. To clean up your CodePipeline resources, follow the instructions in Delete a Pipeline in AWS CodePipeline.
2. To clean up your CodeDeploy resources, follow the instructions in Clean Up Deployment Walkthrough Resources.
3. To delete the S3 bucket, follow the instructions in Deleting or Emptying an S3 Bucket. If you do not intend to create more pipelines, delete the S3 bucket created for storing your pipeline artifacts. For more information about this bucket, see CodePipeline Concepts.



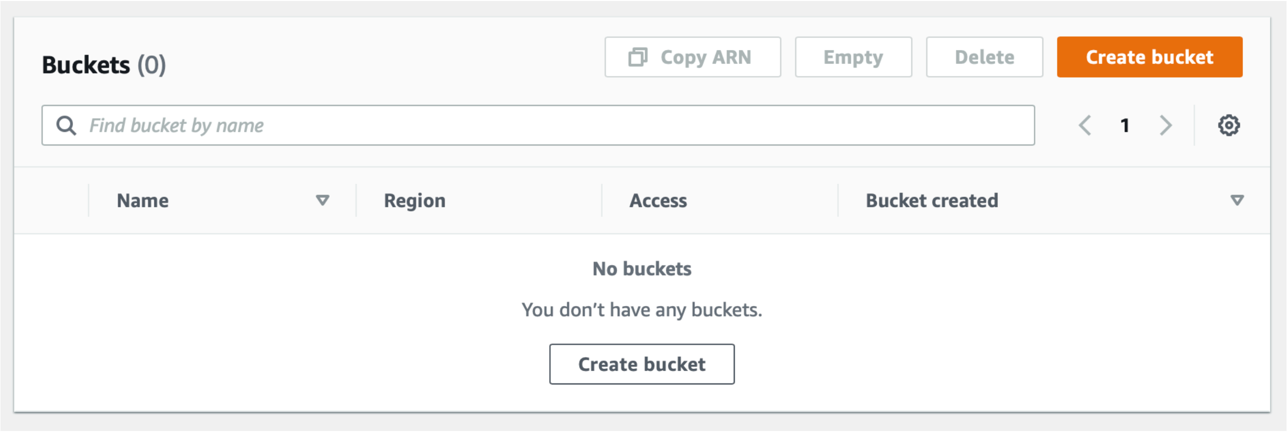
Name	Most recent execution	Latest source revisions	Last executed

No results

There are no results to display.



□	i-017c104370c9cc6cd	t2.micro	eu-west-3b	● terminated
□	i-0c10af7974fd14288	t2.micro	eu-west-3b	● terminated



Name	Region	Access	Bucket created

No buckets

You don't have any buckets.