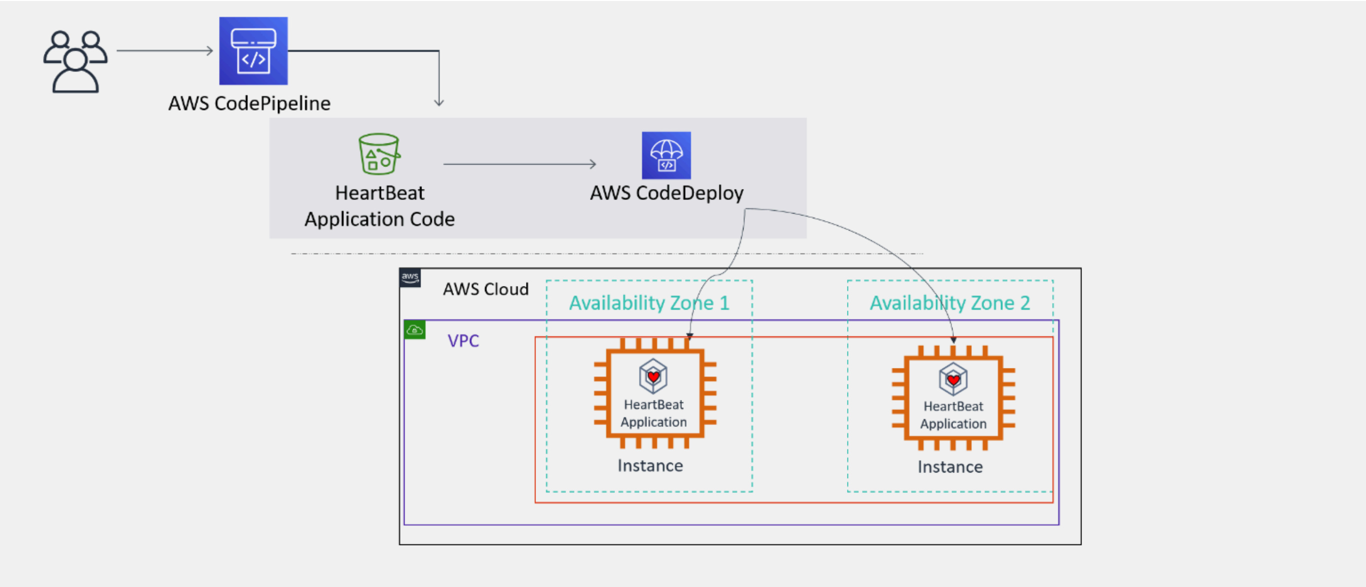
Lab 3: Automating Code Deployments Using AWS CodePipeline

In this lab, we will learn how to automate deployments from AWS CodeDeploy by creating a 2-stage pipeline with AWS CodePipeline. You will deploy a Windows Service application to an Amazon EC2 Fleet running Windows Server OS.

Lab given objectives:

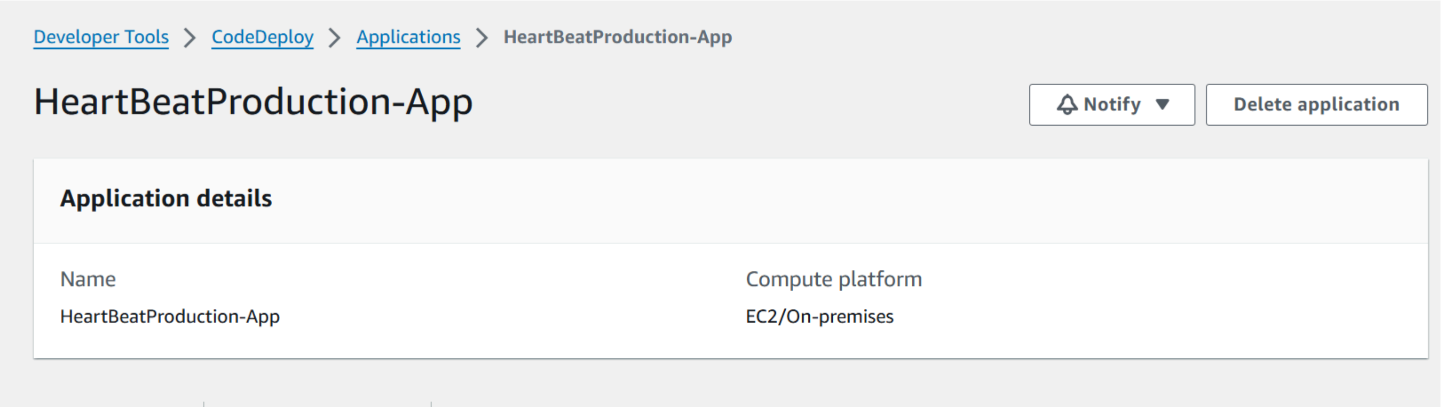
1. Use AWS Cloud9 to prepare your code and upload it to an Amazon S3 bucket.
2. Create a pipeline that automatically deploys your code from the S3 bucket to your servers using AWS CodePipeline and AWS CodeDeploy.
3. Check the deployment settings and automatically deploy your code with AWS CodeDeploy.



The image shows users accessing AWS CodePipeline. The Pipeline pulls the HeartBeat application code and deploys it using AWS CodeDeploy to two EC2 instances in AZ1 and AZ2.

Objective 1: Review AWS CodeDeploy application that is already configured

1.1 Open CodeDeploy and select HBP-App

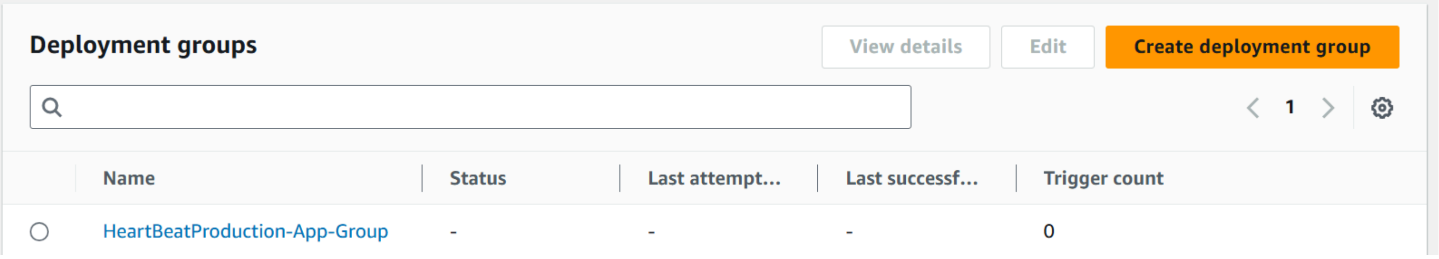

Developer Tools > CodeDeploy > Applications > HeartBeatProduction-App
Notify ▼
Delete application

HeartBeatProduction-App

Application details

Name	Compute platform
HeartBeatProduction-App	EC2/On-premises

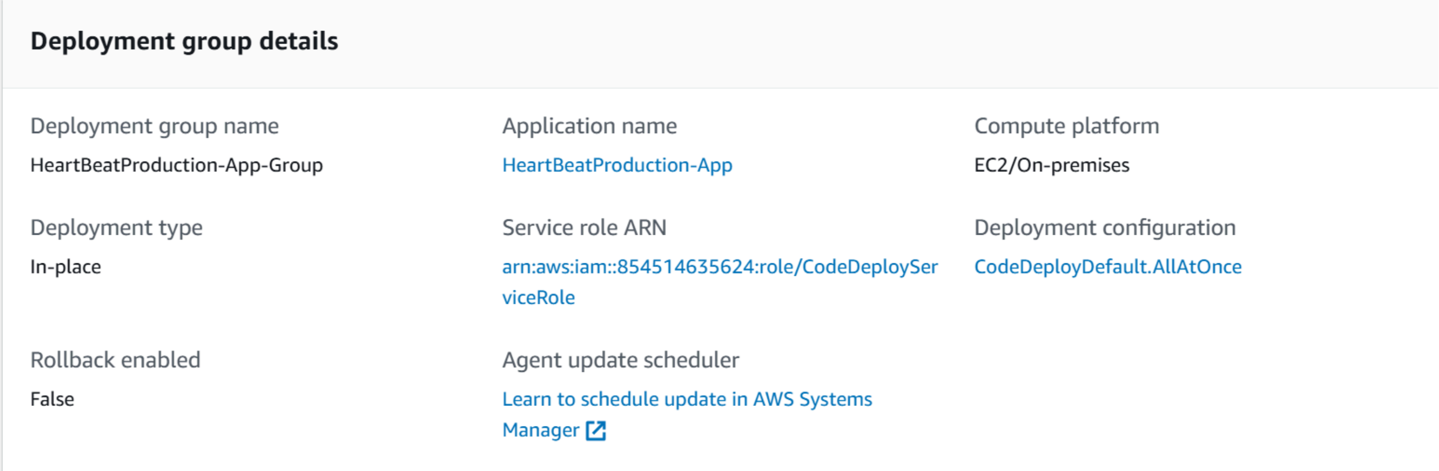
1.2 Open HBP-App in Deployment Groups



Deployment groups

	Name	Status	Last attempt...	Last successf...	Trigger count
○	HeartBeatProduction-App-Group	-	-	-	0

1.3 Verify these details



Deployment group details

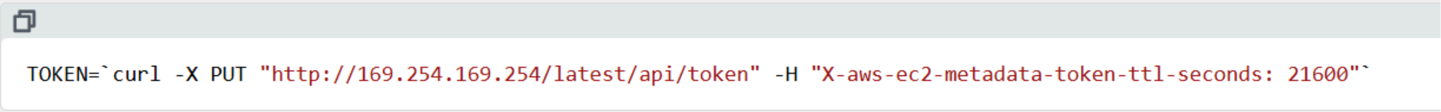
Deployment group name	Application name	Compute platform
HeartBeatProduction-App-Group	HeartBeatProduction-App	EC2/On-premises
Deployment type	Service role ARN	Deployment configuration
In-place	arn:aws:iam::854514635624:role/CodeDeployServiceRole	CodeDeployDefault.AllAtOnce
Rollback enabled	Agent update scheduler	
False	Learn to schedule update in AWS Systems Manager	

Reviewed the CodeDeploy application

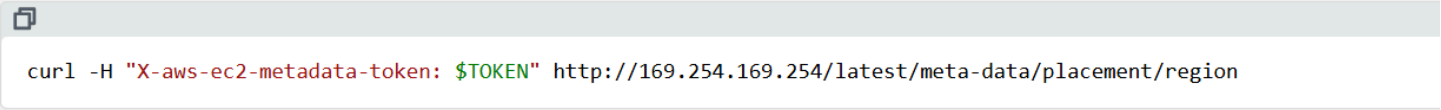
Objective 2 - Preparing the application code for deployment

We store the HeartBeatProduction-App object in this Amazon S3 bucket. AWS CodePipeline deploys the object from this Amazon S3 bucket to the target EC2 fleet.

2.1 Open Cloud9, and execute these codes, to get the region in which Cloud9 instance is running

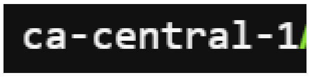


```
TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"`
```



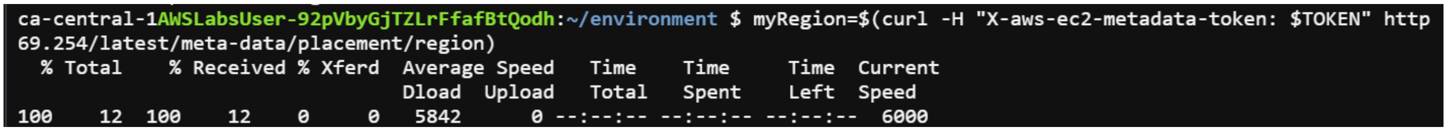
```
curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/placement/region
```

o/p-



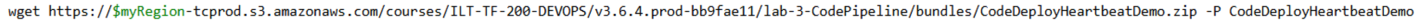
```
ca-central-1
```

2.2 Run the following command to create a myRegion variable to be used in later commands



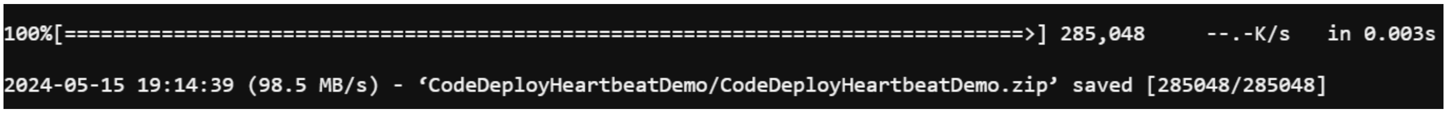
```
ca-central-1AWSLabsUser-92pVbyGjTZLrFfafBtQodh:~/environment $ myRegion=$(curl -H "X-aws-ec2-metadata-token: $TOKEN" http
69.254/latest/meta-data/placement/region)
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100    12  100    12    0     0   5842      0 --:--:-- --:--:-- --:--:--  6000
```

2.3 Run the following code



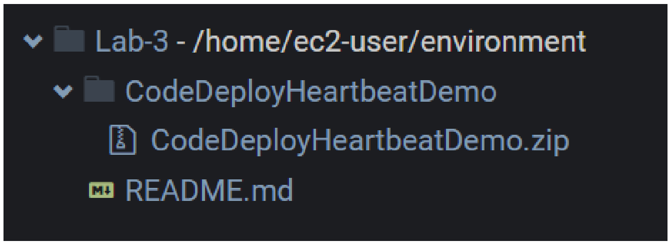
```
wget https://$myRegion-tcprod.s3.amazonaws.com/courses/ILT-TF-200-DEVOPS/v3.6.4.prod-bb9fae11/lab-3-CodePipeline/bundles/CodeDeployHeartbeatDemo.zip -P CodeDeployHeartbeatDemo
```

The command downloads a file named CodeDeployHeartbeatDemo.zip from an Amazon S3 bucket. The URL structure indicates that this file is hosted on an S3 bucket specific to a region, which is specified by the placeholder $myRegion.
Downloaded:-



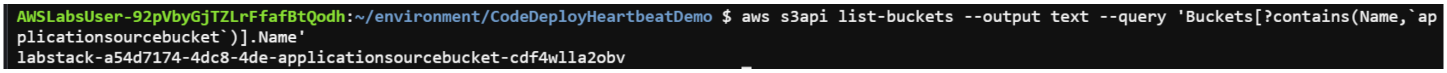
```
100%[==========================================================================>] 285,048     --.-K/s   in 0.003s

2024-05-15 19:14:39 (98.5 MB/s) - ‘CodeDeployHeartbeatDemo/CodeDeployHeartbeatDemo.zip’ saved [285048/285048]
```



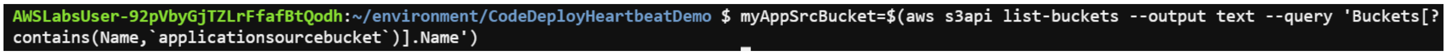
Lab-3 - /home/ec2-user/environment
CodeDeployHeartbeatDemo
CodeDeployHeartbeatDemo.zip
README.md

2.4 We need the Amazon S3 application source bucket name that was provided to complete this task and the next task.



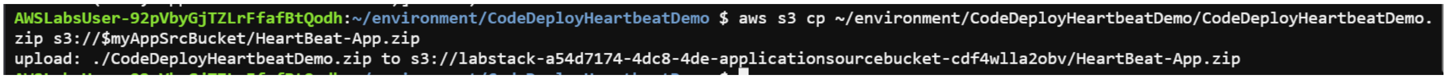
```
AWSLabsUser-92pVbyGjTZLrFfafBtQodh:~/environment/CodeDeployHeartbeatDemo $ aws s3api list-buckets --output text --query 'Buckets[?contains(Name,`ap
plicationsourcebucket`)].Name'
labstack-a54d7174-4dc8-4de-applicationsourcebucket-cdf4wlla2obv
```

2.5 Run the following command to create a variable with that bucket name:



```
AWSLabsUser-92pVbyGjTZLrFfafBtQodh:~/environment/CodeDeployHeartbeatDemo $ myAppSrcBucket=$(aws s3api list-buckets --output text --query 'Buckets[?
contains(Name,`applicationsourcebucket`)].Name')
```

2.6 Run the following command to copy the zipped file to your Amazon S3 bucket:



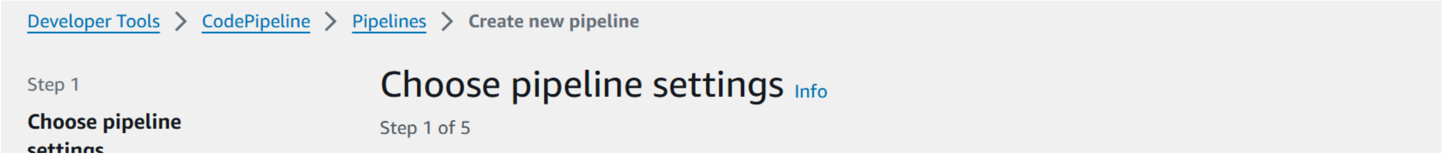
```
AWSLabsUser-92pVbyGjTZLrFfafBtQodh:~/environment/CodeDeployHeartbeatDemo $ aws s3 cp ~/environment/CodeDeployHeartbeatDemo/CodeDeployHeartbeatDemo.
zip s3://$myAppSrcBucket/HeartBeat-App.zip
upload: ./CodeDeployHeartbeatDemo.zip to s3://labstack-a54d7174-4dc8-4de-applicationsourcebucket-cdf4wlla2obv/HeartBeat-App.zip
```

This copies the application files from the AWS Cloud9 IDE environment to the Amazon S3 Application Source bucket.

Objective 3-Creating an AWS CodePipeline

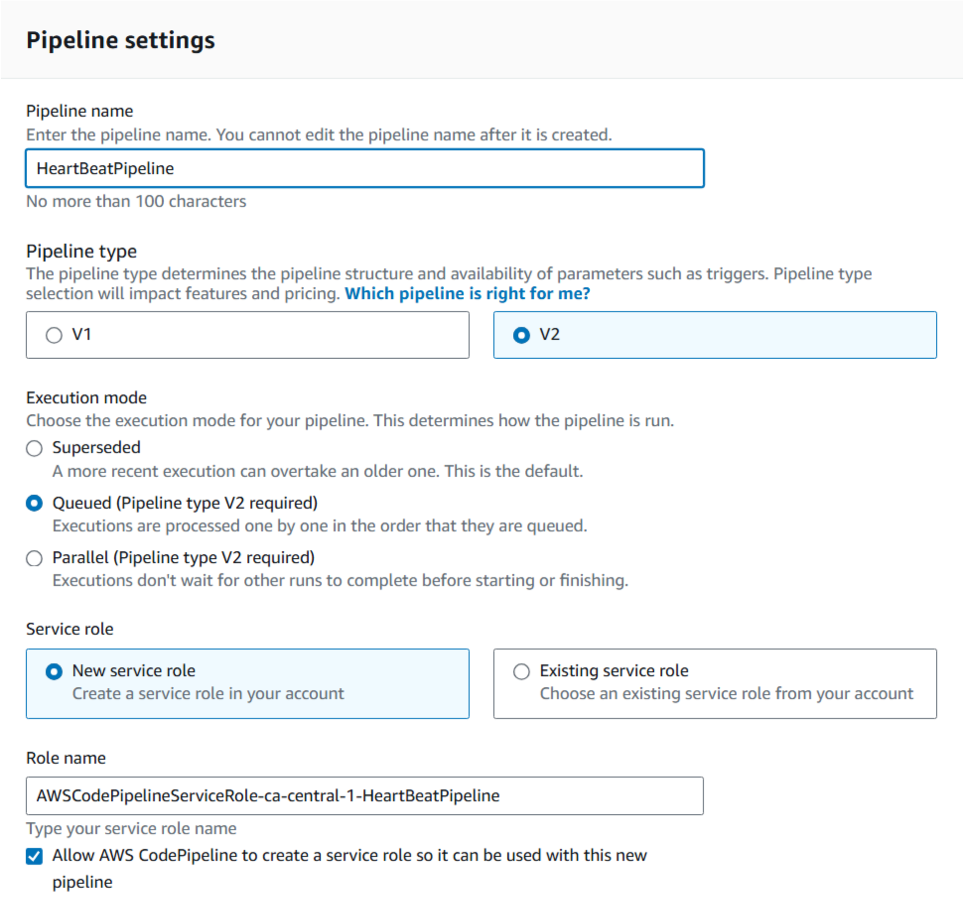
The purpose of this pipeline is to automate the deployment process from Amazon S3 source objects to AWS CodeDeploy deployment group targets.

3.1 Open CodePipeline to create a new pipeline



Developer Tools > CodePipeline > Pipelines > Create new pipeline
Step 1
Choose pipeline settings
Choose pipeline settings Info
Step 1 of 5

3.2 Select these settings



Pipeline settings

Pipeline name

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

HeartBeatPipeline

No more than 100 characters

Pipeline type

The pipeline type determines the pipeline structure and availability of parameters such as triggers. Pipeline type selection will impact features and pricing. **Which pipeline is right for me?**

○ V1 ● V2

Execution mode

Choose the execution mode for your pipeline. This determines how the pipeline is run.

○ Superseded
A more recent execution can overtake an older one. This is the default.

● Queued (Pipeline type V2 required)
Executions are processed one by one in the order that they are queued.

○ Parallel (Pipeline type V2 required)
Executions don't wait for other runs to complete before starting or finishing.

Service role

● New service role
Create a service role in your account

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

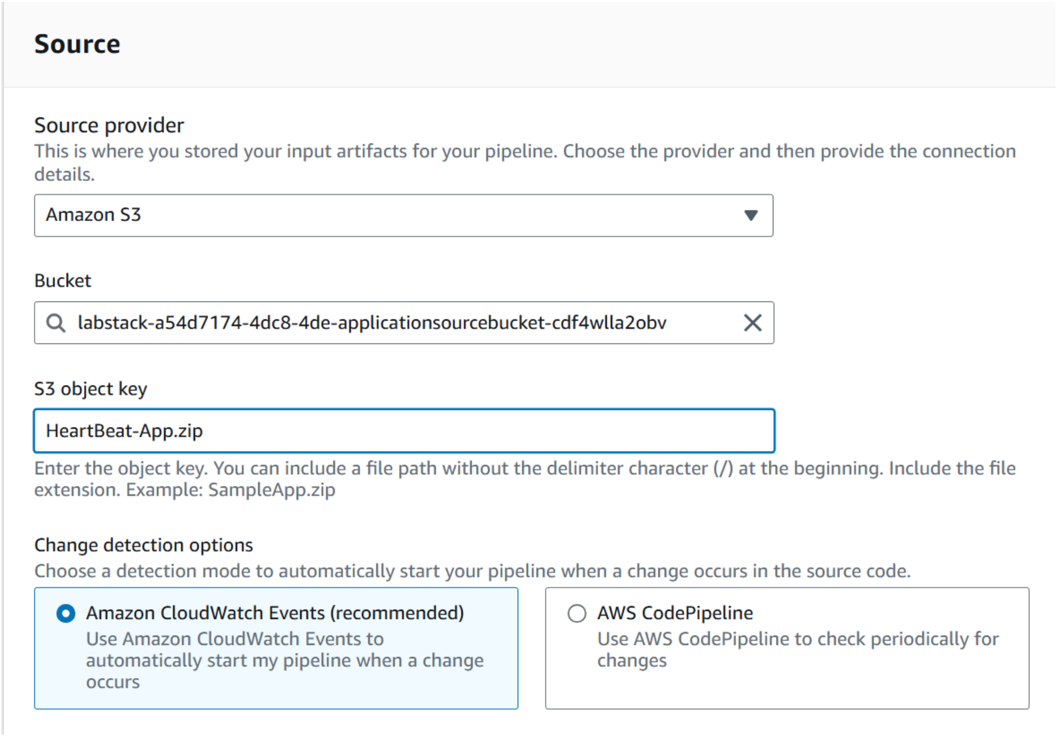
Role name

AWSCodePipelineServiceRole-ca-central-1-HeartBeatPipeline

Type your service role name

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

3.3 Select the source



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

labstack-a54d7174-4dc8-4de-applicationsourcebucket-cdf4wlla2obv

S3 object key

HeartBeat-App.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

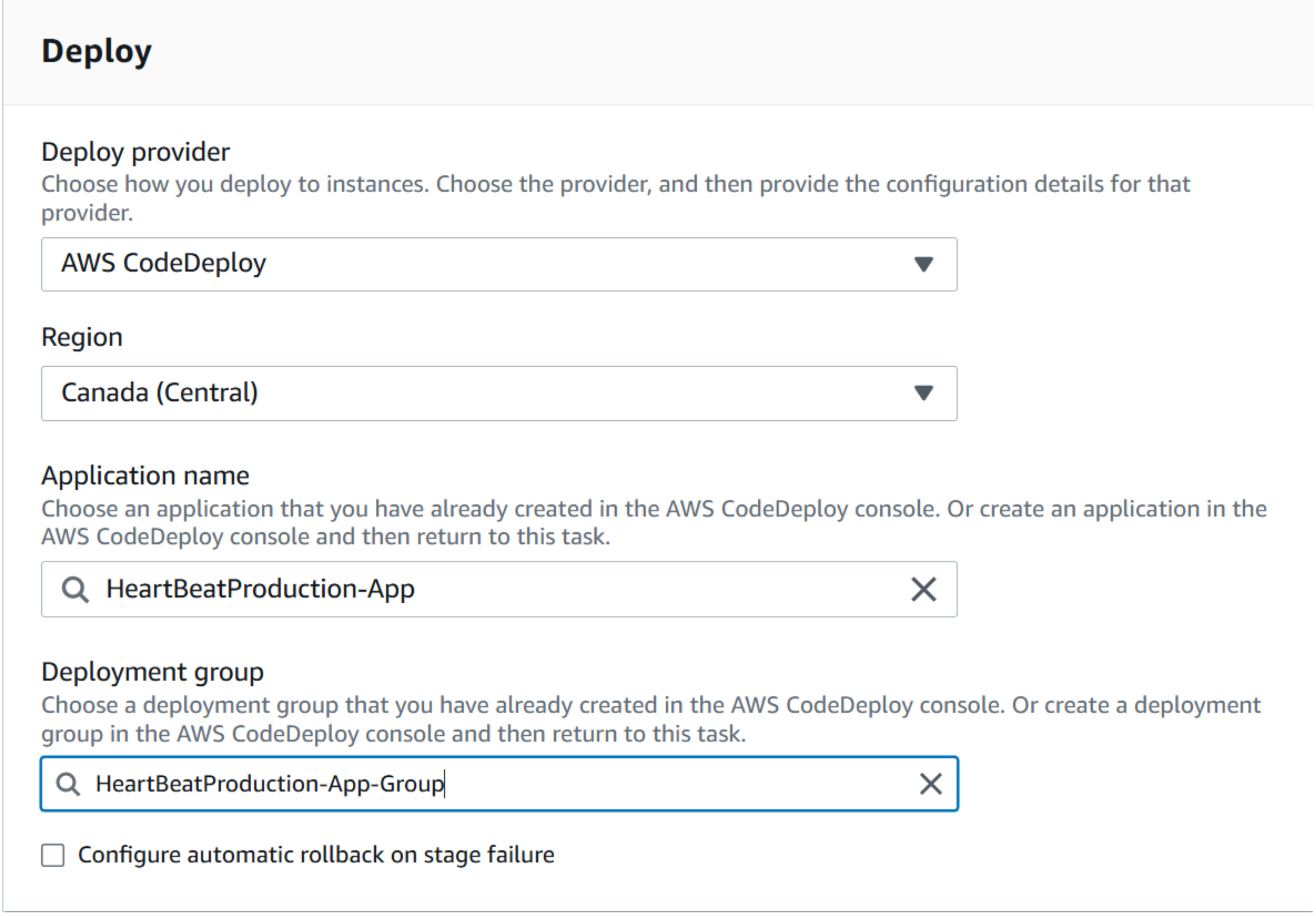
Change detection options

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

3.4 Configure the AWS CodePipeline where the code should be deployed to



Deploy

Deploy provider

Choose how you deploy to instances. Choose the provider, and then provide the configuration details for that provider.

AWS CodeDeploy ▼

Region

Canada (Central) ▼

Application name

Choose an application that you have already created in the AWS CodeDeploy console. Or create an application in the AWS CodeDeploy console and then return to this task.

HeartBeatProduction-App

Deployment group

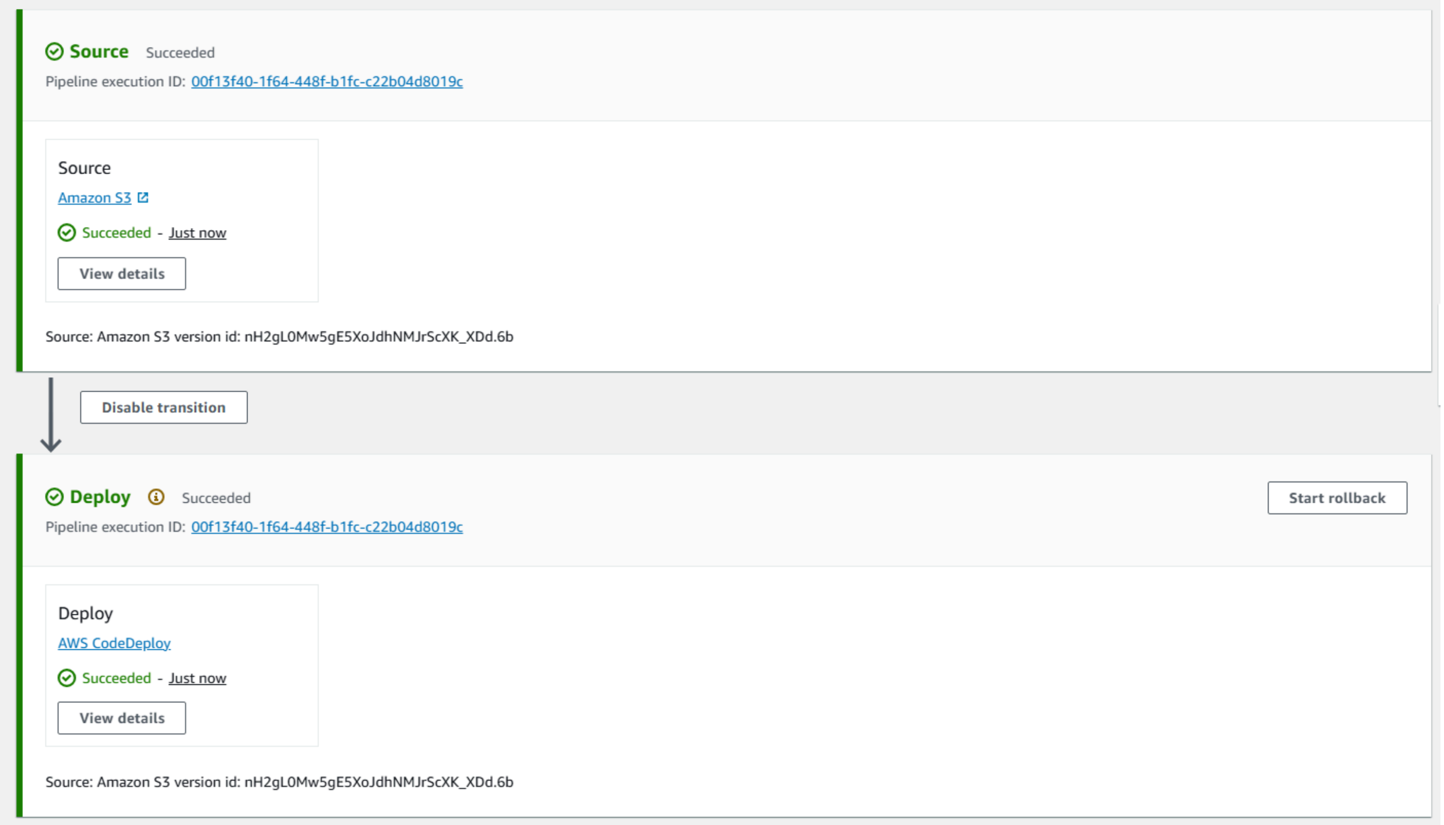
Choose a deployment group that you have already created in the AWS CodeDeploy console. Or create a deployment group in the AWS CodeDeploy console and then return to this task.

HeartBeatProduction-App-Group

□ Configure automatic rollback on stage failure

3.5 Review the settings

3.6 Once the pipeline is created, watch the automatic deployment from the Source stage to the Deploy stage until it completes.



✓ Source Succeeded

Pipeline execution ID: 00f13f40-1f64-448f-b1fc-c22b04d8019c

Source

Amazon S3

✓ Succeeded - Just now

Source: Amazon S3 version id: nH2gL0Mw5gE5XoJdhNMJrScXK_XDd.6b

✓ Deploy ⓘ Succeeded

Pipeline execution ID: 00f13f40-1f64-448f-b1fc-c22b04d8019c

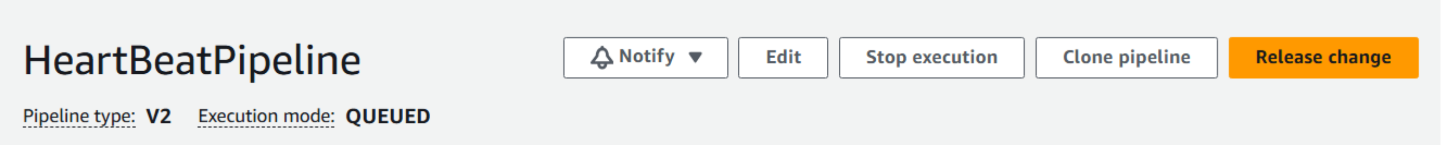
Deploy

AWS CodeDeploy

✓ Succeeded - Just now

Source: Amazon S3 version id: nH2gL0Mw5gE5XoJdhNMJrScXK_XDd.6b

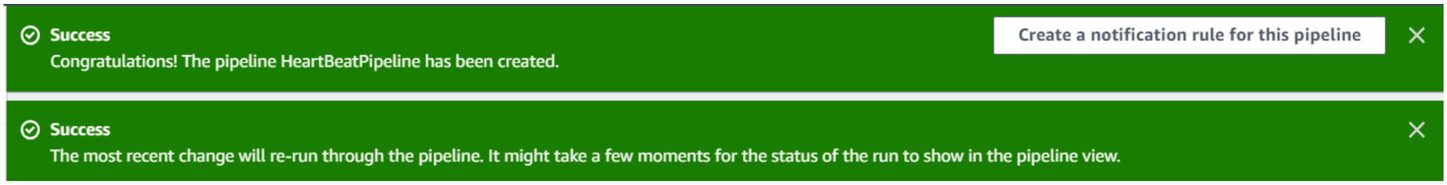
3.7 Force the pipeline to do a deployment by choosing Release change



HeartBeatPipeline

Pipeline type: V2 Execution mode: QUEUED

3.8 The pipeline HeartBeatPipeline has been created.



✓ Success

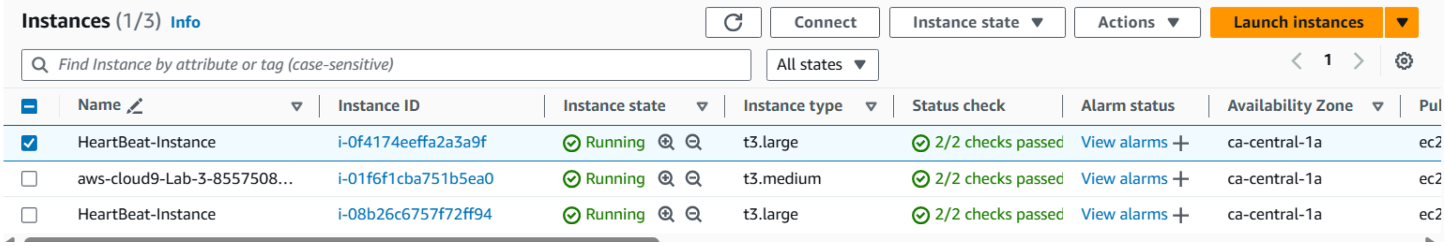
Congratulations! The pipeline HeartBeatPipeline has been created.

✓ Success

The most recent change will re-run through the pipeline. It might take a few moments for the status of the run to show in the pipeline view.

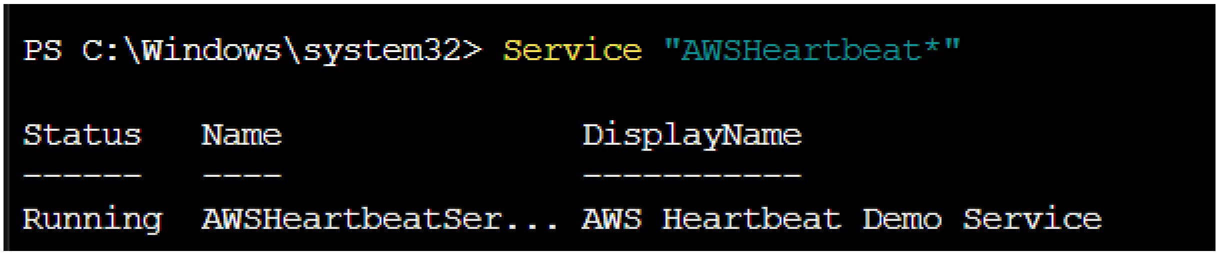
Objective 4- Verifying the CodePipeline deployment

4.1 Connect to HB-Instance in EC2 Insatances



■	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Pul
☑	HeartBeat-Instance	i-0f4174eeffa2a3a9f	✓ Running	t3.large	✓ 2/2 checks passed	View alarms +	ca-central-1a	ec2
□	aws-cloud9-Lab-3-8557508...	i-01f6f1cba751b5ea0	✓ Running	t3.medium	✓ 2/2 checks passed	View alarms +	ca-central-1a	ec2
□	HeartBeat-Instance	i-08b26c6757f72ff94	✓ Running	t3.large	✓ 2/2 checks passed	View alarms +	ca-central-1a	ec2

4.2 Use Windows PowerShell to verify that the HeartBeat service is running as expected after the deployment.

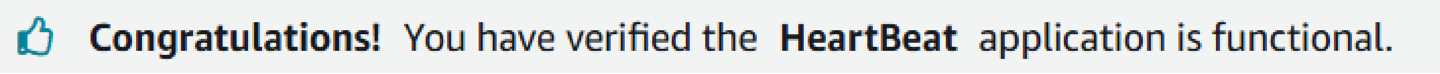


```
PS C:\Windows\system32> Service "AWSHeartbeat*"

Status   Name               DisplayName
------   ----               -----------
Running  AWSHeartbeatSer... AWS Heartbeat Demo Service
```

HBServices is running

***



Congratulations! You have verified the **HeartBeat** application is functional.

***