

AWS CODEDEPLOY AND CODE PIPELINE

CODE PIPELINE:

AWS CodePipeline is a continuous delivery service that enables you to model, visualize, and automate the steps required to release your software. ... AWS CodePipeline then builds, tests, and deploys your application according to the defined workflow every time there is a code change

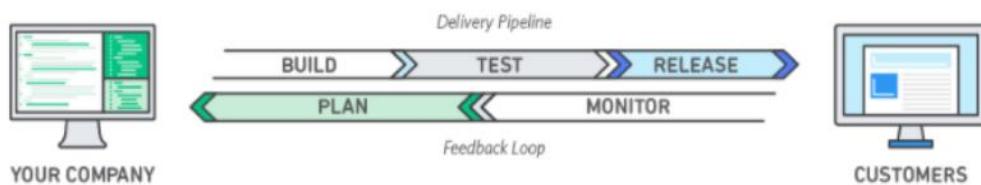
CODE DEPLOY:

AWS CodeDeploy is a service that automates code deployments to Elastic Compute Cloud (EC2) and on-premises servers. Accelerating how fast a developer can release code allows him to release new features for an application faster and avoid deployment errors in complex applications.

OBJECTIVE TO THE CODE DEPLOY AND CODE PIPELINE:

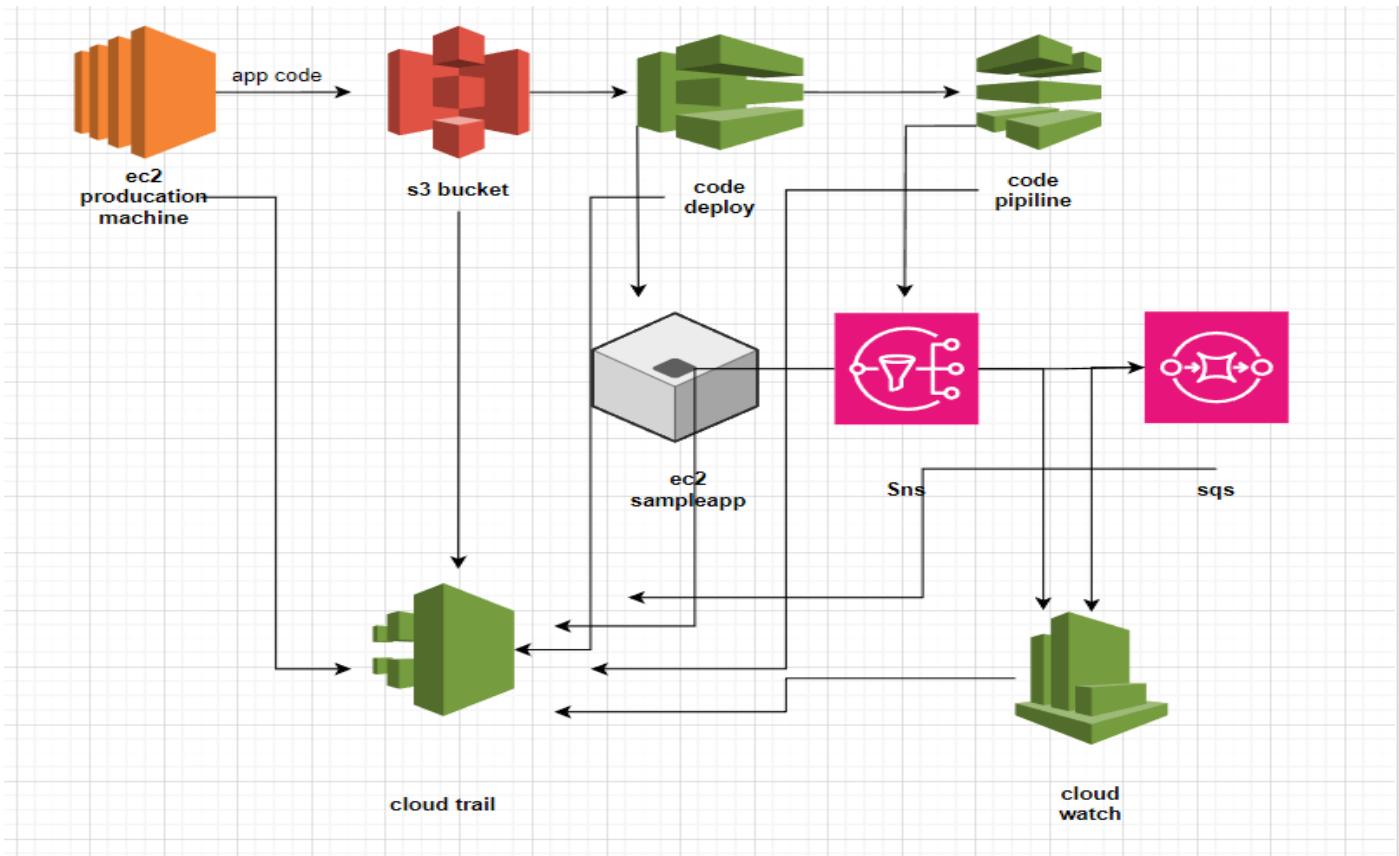
Automation: Automate repetitive tasks such as building, testing, and deploying code. This reduces manual effort and the risk of human error. Ensure that each build and deployment follows the same set of steps, leading to more predictable and reliable outcomes.

Continuous Integration and Continuous Delivery (CI/CD): Implement CI/CD practices to frequently integrate code changes and deliver updates to users quickly and efficiently. additional services sns,sqs,cloud deploy,cloud trail Support the scaling of development and deployment processes to handle increasing complexity and volume of code changes.



CODE DEPLOY AND CODE PIPELINE PROJECT:

Code deploy and code pipeline project



CODE DEPLOY AND CODE PIPELINE PROJECT STEPS:

CodeDeploy is a deployment service from AWS which can automate application deployments to Amazon EC2 instances, on-premises instances or Lambda functions. This does a onetime deployment, for scheduling of deployment you may have to use AWS CodePipeline also.

Application: A CodeDeploy application can be defined from AWS CodeDeploy web console.

Revision: Represents the code need to be deployed on EC2 instance

Appspec file: This contains the instruction to CodeDeploy, like copying of files, executing the scripts etc during the code deployment process. It is present in the root directory of unzipped code with

Deployment Group: Represent set of machines of Lambda function where code has to be deployed

Deployment: The process of deployment.

Setup in Brief:

I have used two EC2 instance of AMZ2 Linux. First one is the web server we will be configuring, also called CodeDeploy agent. Second EC2 machine is supposed to use by developer where the codes are programmed. The names of the resources in the experiment are arbitrary and may name the resources your own

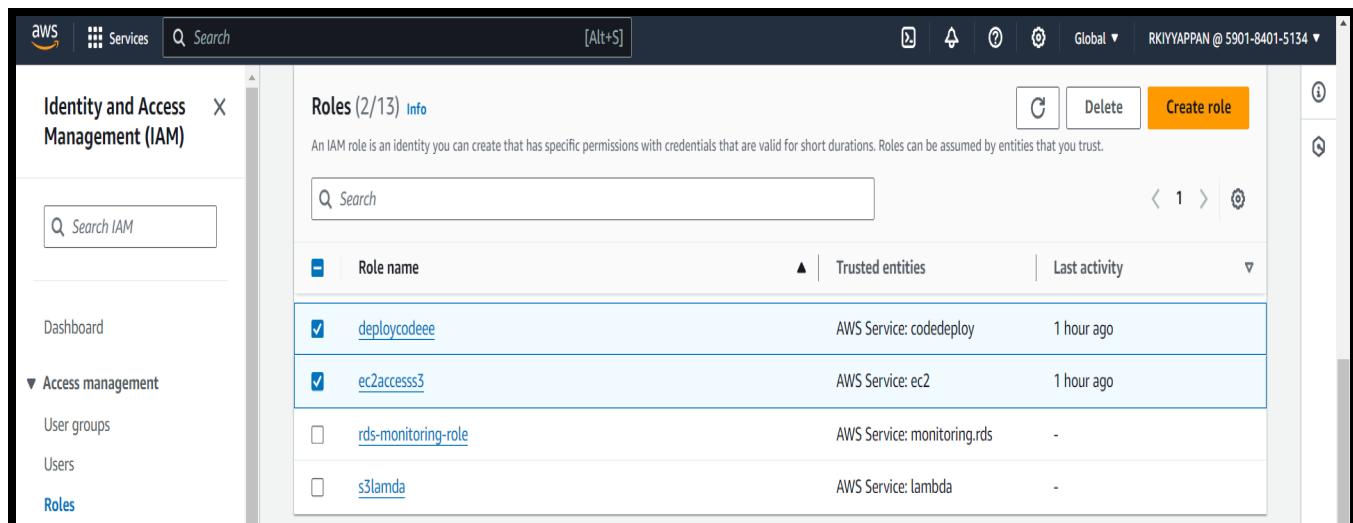
1. Create IAM Roles for EC2-S3-CodeDeploy access
2. Create IAM user account for developer
3. Install and prepare the CodeDeploy agent on webserver.
4. Create the code from Developer machine
5. Create Codedeploy Application and Push the code to S3 bucket from Developer machine
6. Create Deployment Group to include web server
7. Create Deployment to push the code to the Webserver
8. Create sns,sqs,cloud watch and cloud trail, connecting to instance
9. Test the website configuration

SERVICES REQUIRED:

EC2,S3,CD,CP,CW,IAM,SNS,Sqs,Cloudwatch,cloud trail:

Step 1: Create IAM Roles for EC2-S3-CodeDeploy access:

a) s3 to ec2 full access: s3-ec2-full select IAM service goto Roles: select EC2 and click next Permissions:



Then select S3fullaccess policy and click Next: Tags

The screenshot shows a list of AWS policies. A search bar at the top right contains the text 's3'. The table has two columns: 'Policy name' and 'Used as'. The 'AmazonS3FullAccess' policy is selected, indicated by a checked checkbox in the first column. Other policies listed include AmazonDMSRedshiftS3Role, AmazonS3ObjectLambdaExecutionRolePolicy, AmazonS3OutpostsFullAccess, AmazonS3OutpostsReadOnlyAccess, AmazonS3ReadOnlyAccess, AWSRecordToS3, and QuickSightAccessForS3StorageManagementAnalyticsReadOnly.

Choose one or more policies to attach to your new role.		
Create policy		
Filter policies Q s3		Showing 9 results
	Policy name	Used as
<input type="checkbox"/>	AmazonDMSRedshiftS3Role	None
<input checked="" type="checkbox"/>	AmazonS3FullAccess	None
<input type="checkbox"/>	AmazonS3ObjectLambdaExecutionRolePolicy	None
<input type="checkbox"/>	AmazonS3OutpostsFullAccess	None
<input type="checkbox"/>	AmazonS3OutpostsReadOnlyAccess	None
<input type="checkbox"/>	AmazonS3ReadOnlyAccess	None
<input type="checkbox"/>	AWSRecordToS3	None
<input type="checkbox"/>	QuickSightAccessForS3StorageManagementAnalyticsReadOnly	None

b) Create an Code-Deploy role Select service CodeDeploy and use case CodeDeploy

The screenshot shows the AWS IAM Roles list. The left sidebar shows 'Identity and Access Management (IAM)' with 'Roles' selected. The main area displays a table of roles. The 'deploycodeeee' role is selected, indicated by a checked checkbox in the first column. Other roles listed include 'ec2accessss3', 'rds-monitoring-role', and 's3lambda'. The table has columns for 'Role name', 'Trusted entities', and 'Last activity'.

Roles (2/13) Info		
An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.		
Create role		
	Role name	Trusted entities
<input checked="" type="checkbox"/>	deploycodeeee	AWS Service: codedeploy
<input checked="" type="checkbox"/>	ec2accessss3	AWS Service: ec2
<input type="checkbox"/>	rds-monitoring-role	AWS Service: monitoring.rds
<input type="checkbox"/>	s3lambda	AWS Service: lambda

Step 2: Create IAM user account for developer

- Goto users and click add users

The screenshot shows the 'Permissions policies' list in the AWS IAM console. There are 1228 policies listed. One policy, 'AdministratorAccess-Amplify', is selected and highlighted with a blue border. The columns in the table are 'Policy name', 'Type', and 'Attached entities'. The 'Attached entities' column shows the count of entities that have been assigned this policy.

Policy name	Type	Attached entities
AccessAnalyzerServiceRolePolicy	AWS managed	0
AdministratorAccess-Amplify	AWS managed	0
AdministratorAccess-AWSElasticBea...	AWS managed	0
AlexaForBusinessDeviceSetup	AWS managed	0
AlexaForBusinessFullAccess	AWS managed	0

Download .csv file

The screenshot shows the 'User' details page for a user named 'developer'. A success message box is displayed, stating: 'You successfully created the users shown below. You can view and download user security credentials. You can also email users instructions for signing in to the AWS Management Console. This is the last time these credentials will be available to download. However, you can create new credentials at any time.' Below the message, there is a 'Download .csv' button. The user details table includes columns for 'User', 'Access key ID', 'Secret access key', and 'Email login instructions'. The 'Access key ID' for the 'developer' user is AKIA6KKT2J4XQ6KIKBDN, and the 'Secret access key' is masked as ***** Show. There is also a 'Send email' link.

User	Access key ID	Secret access key	Email login instructions
developer	AKIA6KKT2J4XQ6KIKBDN	***** Show	Send email

The screenshot shows the 'Users' list page in the AWS IAM console. It displays one user, 'RKIYAPPAN', with the following details: User name (RKIYAPPAN), Path (/), Groups (0), Last activity (1 hour ago), MFA (0), Password age (2 hours), and Console last sign-in (August 25, 2024, 1). The 'Create user' button is visible at the top right of the user list table.

User name	Path	Groups	Last activity	MFA	Password age	Console last sign-in
RKIYAPPAN	/	0	1 hour ago	0	2 hours	August 25, 2024, 1

Create Amazon Linux EC2 Server with t2.micro instance type ... open all tcp

The screenshot shows the AWS EC2 Dashboard with the search bar set to "Instances (2/4) Info". There are two instances listed:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
production mac...	i-0a264124e9bce23cc	Running	t2.micro	2/2 checks passed	View alarms	us-east-2b	ec2-3-1...
devoleper mac...	i-0b80478d9794706b3	Running	t2.micro	2/2 checks passed	View alarms	us-east-2b	ec2-3-1...

Make this devmachine as a developer machine by configuring user called “developer”.

```
Amazon Linux 2
AL2 End of Life is 2025-06-30.
A newer version of Amazon Linux is available!
Amazon Linux 2023, GA and supported until 2028-03-15.
https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@ip-172-31-17-212 ~]$ sudo -i
[root@ip-172-31-17-212 ~]# aws configure
AWS Access Key ID [None]: AKIAYS2NVMUPASNHUM63
AWS Secret Access Key [None]: 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
Default region name [None]: us-east-2
Default output format [None]: json
[root@ip-172-31-17-212 ~]# aws s3 ls
2024-08-25 07:22:28 s3bucket221
[root@ip-172-31-17-212 ~]# mkdir deploy-dir
[root@ip-172-31-17-212 ~]# cd deploy-dir
[root@ip-172-31-17-212 deploy-dir]# mkdir sampleapp
[root@ip-172-31-17-212 deploy-dir]# cd sampleapp
[root@ip-172-31-17-212 sampleapp]# pwd
/root/deploy-dir/sampleapp
[root@ip-172-31-17-212 sampleapp]# vi index.html
[root@ip-172-31-17-212 sampleapp]# vi appspec.yaml
[root@ip-172-31-17-212 sampleapp]# mkdir scripts
```

Step 3: Install and prepare the CodeDeploy agent on webserver.-black screen Create an Amazon Linux Ec2 server and associate s3-ec2-full roll

```
Amazon Linux 2
AL2 End of Life is 2025-06-30.
A newer version of Amazon Linux is available!
Amazon Linux 2023, GA and supported until 2028-03-15.
https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@ip-172-31-21-217 ~]$ sudo -i
```

yum install ruby -y

```
[root@ip-172-31-21-217 ~]# yum install ruby -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
Resolving Dependencies
--> Running transaction check
--> Package ruby.x86_64 0:2.0.0.648-36.amzn2.0.10 will be installed
--> Processing Dependency: ruby-libs(x86-64) = 2.0.0.648-36.amzn2.0.10 for package: ruby-2.0.0.648-36.amzn2.0.10.x86_64
--> Processing Dependency: ruby(rubygems) >= 2.0.14.1 for package: ruby-2.0.0.648-36.amzn2.0.10.x86_64
--> Processing Dependency: rubygem(bigdecimal) >= 1.2.0 for package: ruby-2.0.0.648-36.amzn2.0.10.x86_64
--> Processing Dependency: libruby.so.2.0() (64bit) for package: ruby-2.0.0.648-36.amzn2.0.10.x86_64
--> Running transaction check
--> Package ruby-libs.x86_64 0:2.0.0.648-36.amzn2.0.10 will be installed
--> Package rubygem-bigdecimal.x86_64 0:1.2.0-36.amzn2.0.10 will be installed
--> Package rubygems.noarch 0:2.0.14.1-36.amzn2.0.10 will be installed
--> Processing Dependency: rubygem(io-console) >= 0.4.2 for package: rubygems-2.0.14.1-36.amzn2.0.10.noarch
--> Processing Dependency: rubygem/json >= 2.0.0 for package: rubygems-2.0.14.1-36.amzn2.0.10.noarch
```

yum install wget -y

```
opsworks-cm           | runtime.sagemaker
history              | help
[root@ip-172-31-21-217 ~]# wget https://aws-codedeploy-us-east-1.s3.amazonaws.com/latest/install
--2024-08-25 07:37:10-- https://aws-codedeploy-us-east-1.s3.amazonaws.com/latest/install
Resolving aws-codedeploy-us-east-1.s3.amazonaws.com (aws-codedeploy-us-east-1.s3.amazonaws.com)... 3.5.10.67, 52.217.114.17, 3.5.22.121, ...
Connecting to aws-codedeploy-us-east-1.s3.amazonaws.com (aws-codedeploy-us-east-1.s3.amazonaws.com)|3.5.10.67|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 19045 (19K) []
Saving to: 'install'

100%[=====] 19,045      --.-K/s   in 0s

2024-08-25 07:37:11 (218 MB/s) - 'install' saved [19045/19045]
```

```
# wget https://aws-codedeploy-us-east-1.s3.amazonaws.com/latest/install
# chmod +x install
#./install auto
# service codedeploy-agent status
```

wget <https://aws-codedeploy-us-east-1.s3.amazonaws.com/latest/install>
download install script to install the code deploy agent

```
[root@ip-172-31-43-214 ~]# wget https://aws-codedeploy-us-east-1.s3.amazonaws.com/latest/install
--2021-09-24 07:03:01-- https://aws-codedeploy-us-east-1.s3.amazonaws.com/latest/install
Resolving aws-codedeploy-us-east-1.s3.amazonaws.com (aws-codedeploy-us-east-1.s3.amazonaws.com)... 52.217.73.76
Connecting to aws-codedeploy-us-east-1.s3.amazonaws.com (aws-codedeploy-us-east-1.s3.amazonaws.com)|52.217.73.76|:443...
HTTP request sent, awaiting response... 200 OK
Length: 17482 (17K) []
Saving to: 'install'

100%[=====] 17,482      92.4KB/s   in 0.2s

2021-09-24 07:03:02 (92.4 KB/s) - 'install' saved [17482/17482]
```

Chmod +x install

```
[root@ip-172-31-43-214 ~]# ls -lrt
total 20
-rw-r--r-- 1 root root 17482 May  5 17:48 install
[root@ip-172-31-43-214 ~]# chmod +x install
[root@ip-172-31-43-214 ~]# ls -lrt
total 20
-rwxr-xr-x 1 root root 17482 May  5 17:48 install
[root@ip-172-31-43-214 ~]# █
```

Install the script: By Default it checks on the ruby version.

```
[root@ip-172-31-43-214 ~]# ./install auto
I, [2021-09-24T07:10:55.821574 #4241] INFO -- : Starting Ruby version check.
I, [2021-09-24T07:10:55.821809 #4241] INFO -- : Starting update check.
I, [2021-09-24T07:10:55.821912 #4241] INFO -- : Attempting to automatically detect supported package manager type for system...
```

```
pre hook : 1
Checking the ruby version.
Checking if there is already a process named codedeploy-agent running.
  Installing : codedeploy-agent-1.3.2-1902.noarch

post hook : 1
Check if there is a codedeployagent config file.
Start codedeploy-agent in post hook if this is a first install.
  Verifying : codedeploy-agent-1.3.2-1902.noarch
```

```
[root@ip-172-31-43-214 ~]# service codedeploy-agent status
The AWS CodeDeploy agent is running as PID 4326
[root@ip-172-31-43-214 ~]# █
```

- Step 4:

Create the code from Developer machine – white machine Work in root

a)mkdir deploy_dir
cd deploy_dir
mkdir sampleapp
cd sampleapp
ls -lrt

```
[root@ip-172-31-17-212 ~]# mkdir deploy-dir
[root@ip-172-31-17-212 ~]# cd deploy-dir
[root@ip-172-31-17-212 deploy-dir]# mkdir sampleapp
[root@ip-172-31-17-212 deploy-dir]# cd sampleapp
[root@ip-172-31-17-212 sampleapp]# pwd
```

b) vi index.html

```
[root@ip-172-31-17-212 sampleapp]# pwd  
/root/deploy-dir/sampleapp  
[root@ip-172-31-17-212 sampleapp]# vi index.html
```

Enter below content and save using wq!

```
</header>  
  
<section id="about">  
  <h2>Profile</h2>  
  <p>A highly motivated fresher Cloud Engineer with hands-on experience in AWS  
ic and innovative environment.</p>  
  <h2>Projects</h2>  
  <div class="project">  
    <h3>Portfolio Website</h3>  
    <p>Created a personal portfolio website using HTML and CSS to showcase my  
    </div>  
    <div class="project">  
      <h3>AWS CODE DEPLOY AND CODE PIPELINE PROJECT</h3>  
      <p>Deployed code and hosted dynamic websites using AWS services (EC2, S3  
age and scale a web application</p>  
    </div>  
    <!-- Add more projects as needed -->  
  </section>  
  
<section id="skill">  
  <h2>Skills</h2>  
  <h3>AWS (amazon web services)</h3>  
  <h3>SQL</h3>  
  <h3>linux</h3>  
  <h3>python</h3>  
  
  </section>  
  
<section id="contact">  
  <h2>Contact</h2>
```

:wq! Then enter

c) Create appspec.yml file – This helps to deploy the source code into webserver automatically

```
[root@ip-172-31-17-212 sampleapp]# vi index.html  
[root@ip-172-31-17-212 sampleapp]# vi appspec.yaml
```

```
version: 0.0  
os: linux  
files:  
  - source: /index.html  
    destination: /var/www/html/  
hooks:  
  BeforeInstall:  
    - location: scripts/httpd_install.sh  
      timeout: 300  
      runas: root  
    - location: scripts/httpd_start.sh  
      timeout: 300  
      runas: root  
  ApplicationStop:  
    - location: scripts/httpd_stop.sh  
      timeout: 300  
      runas: root
```

mkdir scripts

```
root@ip-10-1-10-136:~/deploy_dir/sampleapp
version: 0.0
os: linux
files:
- source: /index.html
  destination: /var/www/html/
hooks:
BeforeInstall:
- location: scripts/httpd_install.sh
  timeout: 300
  runas: root
- location: scripts/httpd_start.sh
  timeout: 300
  runas: root
ApplicationStop:
- location: scripts/httpd_stop.sh
  timeout: 300
  runas: root
-
```

mkdir scripts

cd scripts

```
vi httpd_install.sh
```

```
vi httpd_start.sh
```

```
vi httpd_stop.sh
```

```
[root@ip-172-30-0-178 deploy_dir]# cat sampleapp/scripts/httpd_install.sh
#!/bin/bash
yum install -y httpd
```

```
[root@ip-172-30-0-178 deploy_dir]# cat sampleapp/scripts/httpd_start.sh
#!/bin/bash
systemctl start httpd
systemctl enable httpd
```

```
[root@ip-172-30-0-178 deploy_dir]# cat sampleapp/scripts/httpd_stop.sh
#!/bin/bash
systemctl stop httpd
```

```
[root@ip-172-31-17-212 scripts]# ls
[root@ip-172-31-17-212 scripts]# vi httpd_install.sh
[root@ip-172-31-17-212 scripts]# vi httpd_start.sh
[root@ip-172-31-17-212 scripts]# vi httpd_stop.sh
[root@ip-172-31-17-212 scripts]# cd ..
```

Step 5: Create Codedeploy Application and Push the code to S3 bucket from Developer machine

a) **Create bucket name called s3bucket221 – make it public policy**

General purpose buckets (2) Info All AWS Regions			
Create bucket Empty Delete			
Buckets are containers for data stored in S3.			
<input type="text"/> Find buckets by name < 1 > ⚙			
Name	AWS Region	IAM Access Analyzer	Creation date
codepipeline-us-east-2-361029882343	US East (Ohio) us-east-2	View analyzer for us-east-2	August 25, 2024, 14:07:48 (UTC+05:30)
s3bucket221	US East (Ohio) us-east-2	View analyzer for us-east-2	August 25, 2024, 12:52:26 (UTC+05:30)

b) Create bucket name called s3bucket221– make it public policy Run the cli and create an application in the code deploy Note: We might run the command where the appspec.yml kept

```
# aws deploy create-application --application-name sampleapp
```

```
[root@ip-172-31-17-212 sampleapp]# aws deploy create-application --application-name sampleapp
{
    "applicationId": "52a8b40b-4206-4c90-84bd-62452aa7dfe2"
}
[root@ip-172-31-17-212 sampleapp]# aws deploy push --application-name sampleapp --s3-location s3://s3bucket221/sampleapp.zip
```

The screenshot shows the AWS CodeDeploy Applications page. On the left, there's a sidebar with 'Developer Tools' and 'CodeDeploy' selected. Under 'CodeDeploy', there are sections for Source (CodeCommit), Artifacts (CodeArtifact), Build (CodeBuild), and Deploy (CodeDeploy). Under 'Deploy', 'Getting started', 'Deployments', and 'Applications' are listed. The main content area shows the 'Applications' section with a search bar and buttons for 'Notify', 'View details', 'Deploy application', and 'Create application'. A table lists the application 'sampleapp' with details: Compute platform 'EC2/On-premises', and Created '22 minutes ago'.

c- Now upload the code to S3 by executing the command below.

Directory of execution is important. S3bucket221

```
# aws deploy push --application-name sampleapp --s3-location
s3://s3bucket221/sampleapp.zip
```

```
[root@ip-172-31-17-212 sampleapp]# aws deploy push --application-name sampleapp --s3-location s3://s3bucket221/sampleapp.zip
/root/deploy-dir/sampleapp/appspec.yaml was not found
[root@ip-172-31-17-212 sampleapp]# mv appspec.yaml appspec.yml
[root@ip-172-31-17-212 sampleapp]# aws deploy push --application-name sampleapp --s3-location s3://s3bucket221/sampleapp.zip
To deploy with this revision, run:
```

d- Now browse the s3 bucket to see that sampleapp.zip is present.

The screenshot shows the AWS S3 'Objects' page. At the top, there are buttons for 'Copy S3 URI', 'Copy URL', 'Download', 'Open', 'Delete', 'Actions', 'Create folder', and 'Upload'. Below this is a note about granting permissions. A search bar and a 'Show versions' button are also present. The main table lists one object:

Name	Type	Last modified	Size	Storage class
sampleapp.zip	zip	August 26, 2024, 16:39:06 (UTC+05:30)	3.1 KB	Standard

Step 6: Create Deployment Group to include web server

a- Login to Codedeply AWS web console

The screenshot shows the AWS CodeDeploy 'Applications' page. The left sidebar has sections for Source (CodeCommit), Artifacts (CodeArtifact), Build (CodeBuild), and Deploy (CodeDeploy). Under Deploy, 'Getting started' and 'Deployments' are listed, while 'Applications' is selected. The main area shows a table of applications:

Application name	Compute platform	Created
sampleapp	EC2/On-premises	1 day ago

b- Select sampleapp and click Create Deployment Group from Deployment Groups tab.

The screenshot shows the AWS CodeDeploy 'Application' configuration page for 'sampleapp'. It includes fields for 'Deployment group name' (set to 'ec2-deploy') and 'Service role' (set to 'arn:aws:iam::590184015134:role/deploycodeee').

aws | Services | Search | [Alt+S]

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
0 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	Value - optional
<input type="text" value="Name"/> X	<input type="text" value="production machine"/> X

[Remove tag](#)

aws | Services | Search | [Alt+S] | Ohio | RKIYAPPAN @ 5901

Developer Tools | **CodeDeploy**

- Source • CodeCommit
- Artifacts • CodeArtifact
- Build • CodeBuild
- Deploy • **CodeDeploy**

Getting started

Deployments

Applications

Application

Settings

Developer Tools > CodeDeploy > Applications > sampleapp > ec2-deploy

ec2-deploy

[Edit](#) | [Delete](#) | [Create deployment](#)

Deployment group details

Deployment group name	Application name	Compute platform
ec2-deploy	sampleapp	EC2/On-premises
Deployment type	Service role ARN	Deployment configuration
In-place	arn:aws:iam::590184015134:role/deploycodeeee	CodeDeployDefault.AllAtOnce
Rollback enabled	Agent update scheduler	
False	Learn to schedule update in AWS Systems Manager	

c- Enter the values like below and leave the other parameters default

Enter a deployment group name: mygrp

Choose a service role: cdrole

Deployment type: in-place

Environment configuration: choose Amazon EC2 instances

Key as AppName Value as SampleApp

Load balancer: uncheck Enable load balancing

Click Create Deployment Group button to finish creation of deployment group.

7-Create Deployment which pushes code to the webserver

In the sampleapp click Create Deployment. Enter values like below. Other parameter can can be kept default.

Deployment group : mygrp

Revision type: My application is stored in Amazon S3

Revision location : s3://select_location_from_list

```
version: 0.0
os: linux
files:
- source: /index.html
  destination: /var/www/html/
hooks:
BeforeInstall:
- location: scripts/httpd_install.sh
  timeout: 300
  runas: root
- location: scripts/httpd_start.sh
  timeout: 300
  runas: root
applicationStop:
- location: scripts/httpd_stop.sh
  timeout: 300
  runas: root
```

AWS Services Search [Alt+S] Ohio RKİYYAPPAN @ 5901-8401-5134

Developer Tools > CodeDeploy > Deployments > d-SHJ0JDTP6

d-SHJ0JDTP6

Deployment status

Installing application on your instances 100%
1 of 1 instances updated Succeeded

Copy deployment Retry deployment

Source • CodeCommit
Artifacts • CodeArtifact
Build • CodeBuild
Deploy • CodeDeploy
Getting started
Deployments
Deployment

Launch an instance | X | Modify IAM role | EC | X | EC2 Instance Connect | X | EC2 Instance Connect | X | Deployments | CodeDeploy | X | Roles | IAM | Global | X | Sample Registration | X | +

Not secure 3.144.158.200

View site information

Registration form

Birthday: Day Month Year

*First name: Name

Middle name: Middle Name

*Last name: Last Name

*Username: Username Name

*Password: Password

Remember me

* Must be filled out

Cancel

Forgot password? Already register?

91°F Mostly sunny

Search

13:46 25-08-2024 ENG IN

Successful done the deployment

CREATE PIPELINE TO ENABLE AUTOMATIC DEPLOYMENT MOMENT THE NEW VERSION REACHES THE REPOSITORY CALLED S3

The screenshot shows the AWS CodePipeline Pipelines page. On the left, there is a sidebar with navigation links: Source (CodeCommit), Artifacts (CodeArtifact), Build (CodeBuild), Deploy (CodeDeploy), Pipeline (CodePipeline), Getting started, and Pipelines. The main area has a header 'Pipelines' with tabs for 'Info' and 'Notify'. It includes buttons for 'View history', 'Release change', 'Delete pipeline', and a prominent orange 'Create pipeline' button. Below the header is a search bar and a table with columns: Name, Most recent execution, Latest source revisions, and Last executed. A message 'No results' indicates there are no pipelines to display.

The screenshot shows the 'Choose pipeline settings' step in the AWS CodePipeline pipeline creation wizard. The left sidebar shows the same navigation as the previous screenshot. The main area has a title 'Choose pipeline settings' with a 'Info' link. It features a 'Pipeline settings' section. Under 'Pipeline name', the value 'mypl' is entered into a text input field with a note: 'Enter the pipeline name. You cannot edit the pipeline name after it is created.' Below this is a note: 'No more than 100 characters'. Under 'Service role', there are two options: 'New service role' (selected) and 'Existing service role'. The 'New service role' option includes a sub-note: 'Create a service role in your account'. In the 'Role name' field, the value 'AWSCodePipelineServiceRole-ap-south-1-mypl' is entered, with a note: 'Type your service role name'. A checked checkbox says: 'Allow AWS CodePipeline to create a service role so it can be used with this new pipeline'. At the bottom, there is an 'Advanced settings' section and a 'Cancel' button next to an orange 'Next' button.

SKIP THE BUILD STAGE AND MOVE ON..

The screenshot shows the AWS CodePipeline 'Create new pipeline' wizard. The left sidebar lists navigation options: Source (CodeCommit), Artifacts (CodeArtifact), Build (CodeBuild), Deploy (CodeDeploy), Pipeline (CodePipeline), Getting started, Pipelines, and Pipelines (selected). The main pane shows the 'Add build stage' step, which is optional. It includes a 'Build provider' section with a dropdown menu. Navigation buttons at the bottom include 'Cancel', 'Previous', 'Skip build stage' (highlighted in orange), and 'Next'.

EXCEPTION OCCURES LIKE BELOW,

The screenshot shows the AWS CodePipeline pipeline details page for 'codepip21'. The left sidebar shows the same navigation options as the previous screenshot. The main pane displays the pipeline name 'codepip21', pipeline type 'V2', and execution mode 'QUEUED'. A green success status is shown for the 'Source' stage, which is configured to use 'Amazon S3'. The pipeline execution ID is listed as [630b09e9-9fb2-4dcb-8cdc-f653d3537fab](#). A 'View details' button is available for the source stage. A note at the bottom states 'Source: Amazon S3 version id: yhz_0mBCecC6YNANV5mB4ls4uvH2qlp'.

TESTING THE PIPELINE PROCESS: GOTO DEVELOPER MACHINE AND PUSH
NEW VERSION OF THE sampleapp.zip Edit the content:

```
[root@ip-172-31-13-20 sampleapp]# vi index.html
```

```
<html>
<h2>sample app version 2 </h2>
</html>
```

EXECUTE THE BELOW COMMAND AND CREATE ZIP FILE FIRST

```
zip -r ..../sampleapp.zip .
[root@ip-172-31-13-20 sampleapp]# zip -r ..../sampleapp.zip .

adding: appspec.yml (deflated 53%)
adding: scripts/ (stored 0%)
adding: scripts/httpd_install.sh (stored 0%)
adding: scripts/httpd_start.sh (deflated 21%)
adding: scripts/httpd_stop.sh (deflated 21%)
adding: index.html (deflated 9%)
```

```
[root@ip-172-31-17-212 deploy-dir]# cd sampleapp
[root@ip-172-31-17-212 sampleapp]# ls
appspec.yml  index.html  scripts
[root@ip-172-31-17-212 sampleapp]# zip -r ..../sampleapp.zip .
  adding: index.html (deflated 73%)
  adding: scripts/ (stored 0%)
  adding: scripts/httpd_install.sh (stored 0%)
  adding: scripts/httpd_start.sh (deflated 21%)
  adding: scripts/httpd_stop.sh (stored 0%)
  adding: appspec.yml (deflated 50%)
[root@ip-172-31-17-212 sampleapp]# cd ..
[root@ip-172-31-17-212 deploy-dir]# ls
sampleapp  sampleapp.zip
[root@ip-172-31-17-212 deploy-dir]# aws s3 cp sampleapp.zip s3://s3bucket221
upload: ./sampleapp.zip to s3://s3bucket221/sampleapp.zip
[root@ip-172-31-17-212 deploy-dir]# █
```

```
[root@ip-172-31-17-212 deploy-dir]# cd sampleapp
[root@ip-172-31-17-212 sampleapp]# ls
appspec.yml  index.html  scripts
[root@ip-172-31-17-212 sampleapp]# zip -r ..../sampleapp.zip .
  adding: index.html (deflated 73%)
  adding: scripts/ (stored 0%)
  adding: scripts/httpd_install.sh (stored 0%)
  adding: scripts/httpd_start.sh (deflated 21%)
  adding: scripts/httpd_stop.sh (stored 0%)
  adding: appspec.yml (deflated 50%)
[root@ip-172-31-17-212 sampleapp]# cd ..
[root@ip-172-31-17-212 deploy-dir]# ls
sampleapp  sampleapp.zip
[root@ip-172-31-17-212 deploy-dir]# aws s3 cp sampleapp.zip s3://s3bucket221
upload: ./sampleapp.zip to s3://s3bucket221/sampleapp.zip
[root@ip-172-31-17-212 deploy-dir]# █
```

NEW SOURCE CODE HAS BEEN SUCCESSFULLY TRIGGERED FROM S3 BY CODE PIPELINE AND DEPLOYMENT NEW VERSION INTO WEB SERVER

Source Succeeded

Pipeline execution ID: [3602ac3b-b72e-49fd-9392-aba2fb89e5c1](#)

Source

Amazon S3

Succeeded - Just now

[View details](#)

Source: Amazon S3 version id: DwZ1xF9kSzVILyCYUPO1AZbxqHgGbKEp

[Disable transition](#)

Deploy Succeeded

Pipeline execution ID: [3602ac3b-b72e-49fd-9392-aba2fb89e5c1](#)

Deploy

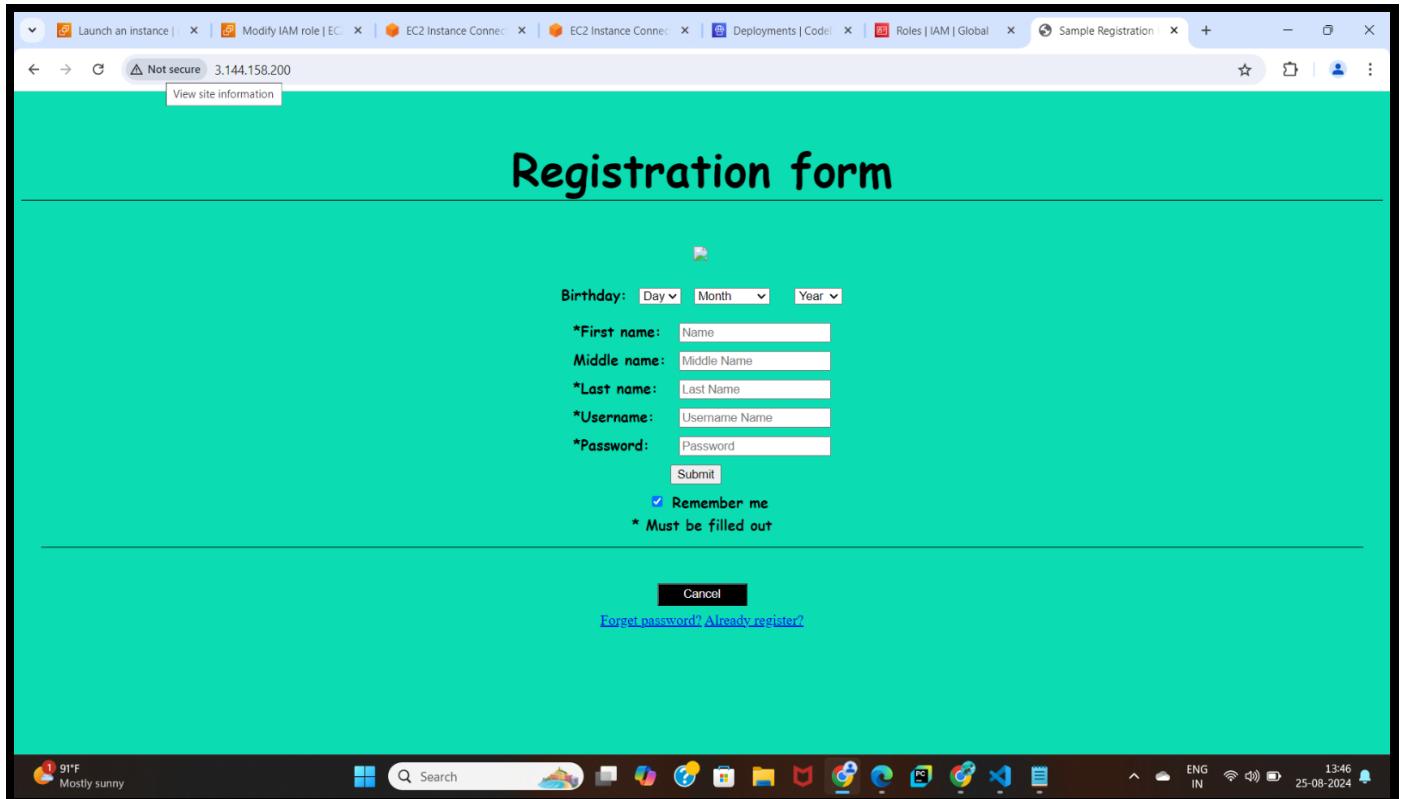
Amazon S3

Succeeded - Just now

[View details](#)

[Start rollback](#)

VERIFY THE SAME BY HITTING THE PUBLIC IP OF THE WEB SERVER MACHINE IN THE BROWSER AND CHECK



ENABLE THE NOTIFICATION SERVICE: ALERTING TO THE USER IN CASE IF THE DEPLOYMENT SUCCEED, FAILED, START ETC

Create SNS name called mytopic and make it as a public

Type: Info
Topic type cannot be modified after topic is created

FIFO (first-in, first-out)

- Strictly-preserved message ordering
- Exactly-once message delivery
- High throughput, up to 300 publishes/second
- Subscription protocols: SQS, Lambda, HTTP, SMS, email, mobile application endpoints

Standard

- Best-effort message ordering
- At-least once message delivery
- Highest throughput in publishes/second
- Subscription protocols: SQS, Lambda, HTTP, SMS, email, mobile application endpoints

Name: my topic.fifo

Maximum 256 characters. Can include alphanumeric characters, hyphens (-) and underscores (_). FIFO topic names must end with ".fifo".

Display name - optional | Info
To use this topic with SMS subscriptions, enter a display name. Only the first 10 characters are displayed in an SMS message.

codedeploy

Maximum 100 characters.

Content-based message deduplication | Info
Enable default message deduplication based on message content. If unchecked, a deduplication ID must be provided for every publish request.

Turn on message deduplication

► Encryption - optional
Amazon SNS provides in-transit encryption by default. Enabling server-side encryption adds at-rest encryption to your topic.

► Access policy - optional | Info

AWS Services Search [Alt+S] ☰ 🔍 ⓘ

Choose method

Basic
Use simple criteria to define a basic access policy.

Advanced
Use a JSON object to define an advanced access policy.

Publishers
Specify who can publish messages to the topic.

Only the topic owner
Only the owner of the topic can publish to the topic

Subscribers
Specify who can subscribe to this topic.

Only the topic owner
Only the owner of the topic can subscribe to the topic

JSON preview

```
{  
    "Version": "2008-10-17",  
    "Id": "__default_policy_ID",  
    "Statement": [  
        {  
            "Sid": "__default_statement_ID",  
            "Effect": "Allow",  
            "Principal": {  
                "AWS": "*"  
            },  
            "Action": [  
                "SNS:Publish",  
                "SNS:RemovePermission",  
                "SNS:GetTopicAttributes"  
            ]  
        }  
    ]  
}
```

▼ Archive policy - new, optional Info

This policy tells Amazon SNS how long to store your messages so that they can be resent to a subscription. By default, Amazon SNS does not retain your messages.

Archive policy
Activate archive policy and select the desired retention period. [Archive policy has additional pricing](#)

AWS Services Search [Alt+S] ☰ 🔍 ⓘ Ohio RKİYYAPPAN @ 5901-8401-5134

Amazon SNS X

New Feature

Amazon SNS now supports in-place message archiving and replay for FIFO topics. [Learn more](#)

Dashboard

Topics

Subscriptions

Mobile

Push notifications

Text messaging (SMS)

Origination numbers

Amazon SNS > Subscriptions

Subscriptions (1)

Edit Delete Request confirmation Confirm subscription Create subscription

Search

ID	Endpoint	Status	Protocol	Topic
b378f25f-8696-42c3-87...	r.iyyappan085@gmail.com	Confirmed	EMAIL	sns://

CREATE SUBSCRIPTION1 –USING EMAIL PROTOCOL

Next cloud watch connecting:

a)cloud watch created:

Screenshot of the AWS CloudWatch Metrics Create Alarm Step 1: Specify metric and conditions.

The graph shows CPUUtilization over time. A blue line represents the metric, and a red dashed line at 21 represents the threshold. The alarm will trigger when the blue line goes below the red line for 1 datapoint within 5 minutes.

Metric

Graph: This alarm will trigger when the blue line goes below the red line for 1 datapoint within 5 minutes.

No unit: 22

21

20

11:30 12:30 13:30

CPUUtilization

Namespace: AWS/EC2
Metric name: CPUUtilization
InstanceId: i-00c13726926c17091
Instance name: No name specified
Statistic: Average
Period: 5 minutes

Whenever CPUUtilization is...

Define the alarm condition.

Greater > threshold Greater/Equal \geq threshold Lower/Equal \leq threshold Lower < threshold

than...

Define the threshold value.

50

Must be a number

► Additional configuration

Connecting Sns:

Notification

Alarm state trigger

Define the alarm state that will trigger this action.

[Remove](#)

In alarm

The metric or expression is outside of the defined threshold.

OK

The metric or expression is within the defined threshold.

Insufficient data

The alarm has just started or not enough data is available.

Send a notification to the following SNS topic

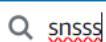
Define the SNS (Simple Notification Service) topic that will receive the notification.

Select an existing SNS topic

Create new topic

Use topic ARN to notify other accounts

Send a notification to...



sns



Only topics belonging to this account are listed here. All persons and applications subscribed to the selected topic will receive notifications.

Email (endpoints)

r.iyyappan085@gmail.com - [View in SNS Console](#)

[Add notification](#)

Lambda action

The screenshot shows the CloudWatch Alarms page. On the left, there's a sidebar with 'CloudWatch' at the top, followed by 'Favorites and recents', 'Dashboards', and 'Alarms' sections. Under 'Alarms', it says '0 in alarm' and 'All alarms'. The main area is titled 'Alarms (1/1)' and shows one alarm named 'alarmmm'. The condition for this alarm is 'CPUUtilization < 100 for 1 datapoints within 5 minutes'. The 'Actions enabled' checkbox is checked.

Cloud watch created successful:

Instant connecting in cloud watch

The screenshot shows the AWS EC2 Dashboard. The left sidebar includes 'EC2 Dashboard', 'EC2 Global View', 'Events', and 'Instances' sections. The 'Instances' section is expanded, showing two instances: 'production mac...' and 'devoleper mac...'. Both instances are listed as 'Running' with 't2.micro' instance type. Their status is '2/2 checks passed' with a yellow warning icon indicating '1 in alarm'. They are located in 'us-east-2b' availability zone and belong to 'ec2-18-' and 'ec2-3-1' security groups.

Confirm the subscription since gmail is an third party server for the aws

The screenshot shows a Gmail inbox with 707 messages. A new email from "AWS Notifications <no-reply@sns.amazonaws.com>" is highlighted. The subject is "ALARM: "alarmmm" in US East (Ohio)". The message body states: "You are receiving this email because your Amazon CloudWatch Alarm "alarmmm" in the US East (Ohio) region has entered the ALARM state, because "Threshold Crossed: 1 out of the last 1 datapoints [0.269642650911456 (25/08/24 15:56:00)] was less than the threshold (100.0) (minimum 1 datapoint for OK -> ALARM transition)." at "Sunday 25 August, 2024 16:01:40 UTC". A link to view the alarm in the AWS Management Console is provided: <https://us-east-2.console.aws.amazon.com/cloudwatch/deeplink.js?region=us-east-2#alarmsV2.alarm/alarmmm>.

SQS created;

The screenshot shows the AWS SQS "Details" configuration page. Under the "Type" section, "Standard Info" is selected, indicating "At-least-once delivery, message ordering isn't preserved" with options for "At-least once delivery" and "Best-effort ordering". The "FIFO Info" section is also shown, indicating "First-in-first-out delivery, message ordering is preserved" with options for "First-in-first-out delivery" and "Exactly-once processing". A note states: "You can't change the queue type after you create a queue." In the "Name" field, the queue is named "mysqs". A note says: "A queue name is case-sensitive and can have up to 80 characters. You can use alphanumeric characters, hyphens (-), and underscores (_).". Under the "Configuration" section, "Visibility timeout" is set to 30 seconds, and "Message retention period" is set to 4 days. "Delivery delay" and "Maximum message size" fields are also present.

AWS Services Search [Alt+S] □ ▲

Amazon SQS provides in-transit encryption by default. To add at-rest encryption to your queue, enable server-side encryption.

Server-side encryption

Disabled
 Enabled

Access policy Info

Define who can access your queue.

Choose method

Basic
Use simple criteria to define a basic access policy.

Advanced
Use a JSON object to define an advanced access policy.

Define who can send messages to the queue

Only the queue owner
Only the owner of the queue can send messages to the queue.

Only the specified AWS accounts, IAM users and roles
Only the specified AWS account IDs, IAM users and roles can send messages to the queue.

Define who can receive messages from the queue

Only the queue owner
Only the owner of the queue can receive messages from the queue.

Only the specified AWS accounts, IAM users and roles
Only the specified AWS account IDs, IAM users and roles can receive messages from the queue.

JSON (read-only)

```
{  
  "Version": "2012-10-17",  
  "Id": "__default_policy_ID",  
  "Statement": [  
    {  
      "Sid": "__owner_statement",  
      "Effect": "Allow",  
      "Principal": {  
        "AWS": "590184015134"  
      },  
      "Action": [  
        "SQS:*"  
      ]  
    }  
  ]  
}
```

AWS Services Search [Alt+S] □ ▲

Select which source queues can use this queue as the dead-letter queue.

Disabled
 Enabled

Dead-letter queue - Optional Info

Send undeliverable messages to a dead-letter queue.

Set this queue to receive undeliverable messages.

Disabled
 Enabled

Tags - Optional Info

A tag is a label assigned to an AWS resource. Use tags to search and filter your resources or track your AWS costs.

Key	Value - optional
<input type="text"/> Name	<input type="text"/> production machine X
Add new tag	

You can add 49 more tags.

Cancel Create queue

Created in sqs:

The screenshot shows the AWS SQS service in the AWS Management Console. The main title bar includes the AWS logo, services menu, search bar, and user information (Ohio region, RKİYYAPPAN @ 5901-8401-5134). The left sidebar has a 'Queues' section. The main content area is titled 'Queues (2)' and lists two items:

Name	Type	Created	Messages available	Messages in flight	Encryption	Content-based deduplication
mysq5	Standard	2024-08-28T12:19+05:30	1	0	Disabled	-
sqs	Standard	2024-07-31T11:55+05:30	1	0	Disabled	-

Buttons at the top right include 'Edit', 'Delete', 'Send and receive messages', 'Actions', and 'Create queue'.

Connected in cloud watch

The screenshot shows the AWS CloudWatch Metrics service. At the top, there's a 'Select metric' dialog with a graph of 'CPUUtilization' over time from 04:00 to 06:45. Below the graph are buttons for 'Add math', 'Add query', and 'Graph with SQL'. The main area is titled 'Metrics (17)' and shows a list of metrics for the 'i-080bd7b1e53267d1f' instance. The list includes:

Instance name	InstanceId	Metric name	Alarms
17/17	i-080bd7b1e53267d1f	NetworkIn	No alarms

At the bottom right are 'Cancel' and 'Select metric' buttons.

This screenshot shows a detailed view of metrics for the same EC2 instance. The interface is similar to the previous one, with a 'Select metric' dialog at the top. The main list shows the following metrics for the instance:

Instance name	InstanceId	Metric name	Alarms
production machine	i-080bd7b1e5326...	DiskReadOps	No alarms
production machine	i-080bd7b1e5326...	CPUUtilization	No alarms
production machine	i-080bd7b1e5326...	DiskWriteBytes	No alarms
production machine	i-080bd7b1e5326...	NetworkOut	No alarms

At the bottom right are 'Cancel' and 'Select metric' buttons.

This alarm will trigger when the blue line goes below the red line for 1 datapoints within 5 minutes.

CPUUtilization

Namespace
AWS/EC2

Metric name

InstanceId

Instance name
production machine

Statistic
 X

Period
 ▼

Conditions

Threshold type

Static
Use a value as a threshold

Anomaly detection
Use a band as a threshold

Notification

Alarm state trigger

Define the alarm state that will trigger this action.

In alarm
The metric or expression is outside of the defined threshold.

OK
The metric or expression is within the defined threshold.

Insufficient data
The alarm has just started or not enough data is available.

Remove

Send a notification to the following SNS topic

Define the SNS (Simple Notification Service) topic that will receive the notification.

- Select an existing SNS topic**
- Create new topic**
- Use topic ARN to notify other accounts**

Send a notification to...

X

Only topics belonging to this account are listed here. All persons and applications subscribed to the selected topic will receive notifications.

Email (endpoints)

arn:aws:sqs:us-east-2:590184015134:codedqs and 1 more - [View in SNS Console](#) ↗

Add notification

The screenshot shows the AWS CloudWatch Alarms interface. The left sidebar has 'CloudWatch' selected under 'Alarms'. The main area shows one alarm named 'codeeee' with the condition 'CPUUtilization < 50 for 1 datapoints within 5 minutes'. The status is 'Insufficient data' and it was last updated on 2024-08-28 at 08:13:32.

Sqs will be connected in cloud watch. cloud watch has been connected to instance:

The screenshot shows the AWS EC2 Instances interface. The left sidebar has 'Instances' selected. There are two instances listed: 'production mac...' (running, t2.micro, 2/2 checks passed) and 'developer mac...' (running, t2.micro, 2/2 checks passed). The 'production mac...' instance is currently selected.

Sqs notification has successful:

The screenshot shows the AWS SQS 'Receive messages' interface. It displays a message received from the queue 'codeeee'. The message ID is 335f778a-1149-4020-9490-f162afc15680, it was sent on 2024-08-28T12:27+05:30, and its size is 1.53 KB. The receive count is 1. The message content is not visible.

Message: 335f778a-1149-4020-9490-f162afc15680

X

Body

Attributes

Details

```
{  
  "Type" : "Notification",  
  "MessageId" : "41f1f77f-025f-5456-ae3f-a837cec30c5c",
```

Done

Next cloud trial connected:

a) cloud trial created

General details

A trail created in the console is a multi-region trail. [Learn more](#)

Trail name

Enter a display name for your trail.

codee

3-128 characters. Only letters, numbers, periods, underscores, and dashes are allowed.

Enable for all accounts in my organization

To review accounts in your organization, open AWS Organizations. [See all accounts](#)

Storage location | [Info](#)

Create new S3 bucket

Create a bucket to store logs for the trail.

Use existing S3 bucket

Choose an existing bucket to store logs for this trail.

Trail log bucket name

Enter a new S3 bucket name and folder (prefix) to store your logs. Bucket names must be globally unique.

s3bucket221

X

Browse

Prefix - optional

prefix

Logs will be stored in s3bucket221/AWSLogs/590184015134

Log file SSE-KMS encryption | [Info](#)

Enabled

▼ Additional settings

CloudWatch Logs - optional

Configure CloudWatch Logs to monitor your trail logs and notify you when specific activity occurs. Standard CloudWatch and CloudWatch Logs charges apply. [Learn more](#)

CloudWatch Logs | [Info](#)

Enabled

Log group [Info](#)

New

Existing

Log group name

aws-cloudtrail-logs-590184015134-973da50d

1-512 characters. Only letters, numbers, dashes, underscores, forward slashes, and periods are allowed.

IAM Role [Info](#)

AWS CloudTrail assumes this role to send CloudTrail events to your CloudWatch Logs log group.

New

Existing

Role name

cloudtrial

► [Policy document](#)

Tags - optional [Info](#)

Choose the type of events that you want to log.

Management events

Capture management operations performed on your AWS resources.

Data events

Log the resource operations performed on or within a resource.

Insights events

Identify unusual activity, errors, or user behavior in your account.

Management events [Info](#)

Management events show information about management operations performed on resources in your AWS account.

i No additional charges apply to log management events on this trail because this is your first copy of management events.

API activity

Choose the activities you want to log.

Read

Write

Exclude AWS KMS events

Exclude Amazon RDS Data API events

[Cancel](#)

[Previous](#)

[Next](#)

Management events

i No additional charges apply to log management events on this trail because this is your first copy of management events.

API activity

All

Exclude AWS KMS events

No

Exclude Amazon RDS Data API events

No

Data events

Data event collection is not configured for this trail

Insights events

You can only enable CloudTrail Insights on trails that log management events. [Learn more](#) 

[Cancel](#)

[Previous](#)

[Create trail](#)

<input checked="" type="radio"/> codee	US East (Ohio)	Yes	Disabled	No	s3bucket221 	-	arn:aws:logs:us-east-2:590184015134:log-group:aws-cloudtrail-logs-590184015134-973da50d:*	 Logging
--	----------------	-----	----------	----	--	---	---	---

Cloud trail creating and connected successful;

Cloud logs out-put:

The screenshot shows the AWS CloudWatch Logs interface. The left sidebar navigation includes CloudWatch, Favorites and recents, Dashboards, Alarms (with 1 new), Logs (selected), Log groups, Metrics, X-Ray traces, and Events. The main pane displays a list of log entries from 2024-08-28T07:24:58.863Z to 2024-08-28T07:27:08.990Z. Each entry contains a timestamp, a log message, and a link to the full CloudTrail event. A yellow box highlights the first few log entries.

Timestamp	Log Message (Excerpt)	Link
2024-08-28T07:24:58.863Z	{"eventVersion": "1.09", "userIdentity": {"type": "IAMUser", "p...}	590184015134_CloudTrail_us-east-2_2
2024-08-28T07:24:58.863Z	{"eventVersion": "1.09", "userIdentity": {"type": "AWSService"...}	590184015134_CloudTrail_us-east-2_2
2024-08-28T07:24:58.863Z	{"eventVersion": "1.09", "userIdentity": {"type": "IAMUser", "p...}	590184015134_CloudTrail_us-east-2_2
2024-08-28T07:24:58.863Z	{"eventVersion": "1.09", "userIdentity": {"type": "IAMUser", "p...}	590184015134_CloudTrail_us-east-2_2
2024-08-28T07:24:58.863Z	{"eventVersion": "1.09", "userIdentity": {"type": "AWSService"...}	590184015134_CloudTrail_us-east-2_2
2024-08-28T07:24:58.863Z	{"eventVersion": "1.09", "userIdentity": {"type": "IAMUser", "p...}	590184015134_CloudTrail_us-east-2_2
2024-08-28T07:24:58.863Z	{"eventVersion": "1.09", "userIdentity": {"type": "IAMUser", "p...}	590184015134_CloudTrail_us-east-2_2
2024-08-28T07:24:58.863Z	{"eventVersion": "1.09", "userIdentity": {"type": "AssumedRole...}	590184015134_CloudTrail_us-east-2_2
2024-08-28T07:24:58.863Z	{"eventVersion": "1.09", "userIdentity": {"type": "IAMUser", "p...}	590184015134_CloudTrail_us-east-2_2
2024-08-28T07:24:58.863Z	{"eventVersion": "1.09", "userIdentity": {"type": "IAMUser", "p...}	590184015134_CloudTrail_us-east-2_2
2024-08-28T07:24:58.863Z	{"eventVersion": "1.09", "userIdentity": {"type": "IAMUser", "p...}	590184015134_CloudTrail_us-east-2_2
2024-08-28T07:24:58.863Z	{"eventVersion": "1.09", "userIdentity": {"type": "IAMUser", "p...}	590184015134_CloudTrail_us-east-2_2
2024-08-28T07:24:58.863Z	{"eventVersion": "1.09", "userIdentity": {"type": "AWSService"...}	590184015134_CloudTrail_us-east-2_2
2024-08-28T07:24:58.863Z	{"eventVersion": "1.09", "userIdentity": {"type": "IAMUser", "p...}	590184015134_CloudTrail_us-east-2_2
2024-08-28T07:27:08.990Z	{"eventVersion": "1.09", "userIdentity": {"type": "IAMUser", "p...}	590184015134_CloudTrail_us-east-2_3

[Back to top ^](#)

VERIFY THE NEW SOURCE CODE DEPLOYMENT WITH CODE PIPELINE SUCCESSFULLY DONE

DEMOLISH:

- 1. DELETE INSTANCES**
- 2. DELETE CODE DEPLOY**
- 3. DELETE CODE PIPELINE**
- 4. DELETE BUCKET AND OBJECT**
- 5. DELETE SNS & SQS TOPIC AND SUBSCRIPTIONS**
- 6. DELETE CLOUD WATCH AND CLOUD TRAIL**
- 7. DELETE IAM ROLES AND USERS**