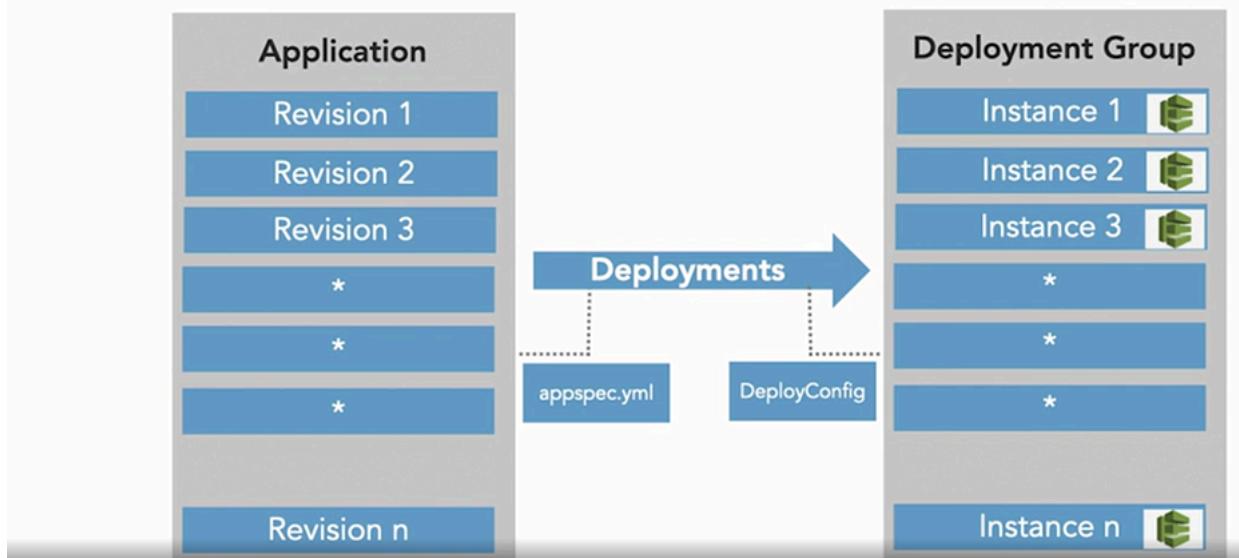


Establishing automated code deployment is the heart of CI/CD pipeline.

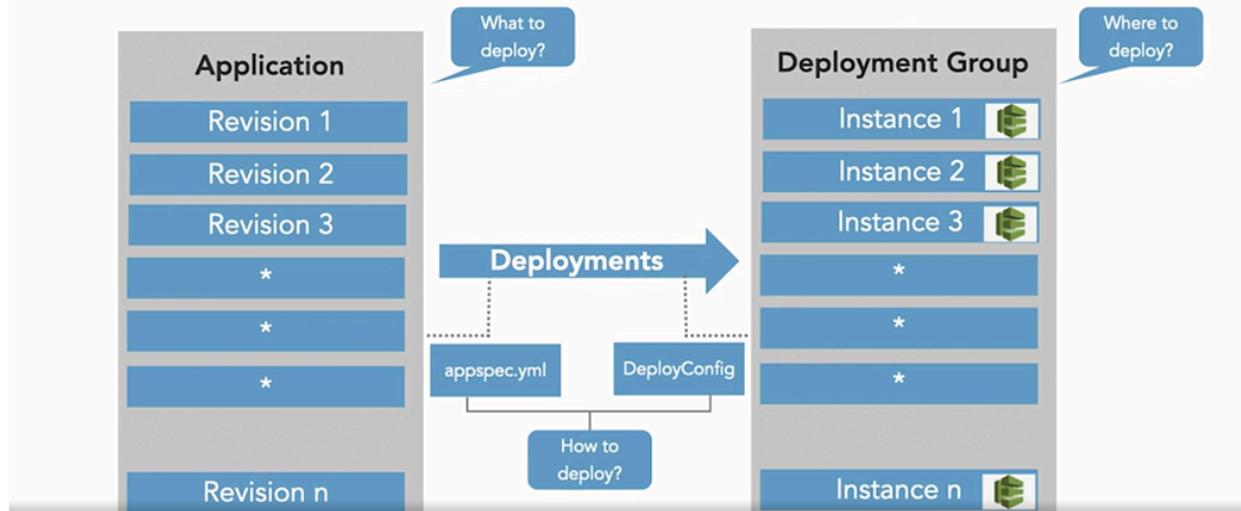
CodeDeploy: Overview

- Automate code deployment on EC2, on-premises, Lambda, and ECS
- Minimal downtime
- Rollback automatically, in case of failures
- Integrate with third-party tools like Jenkins

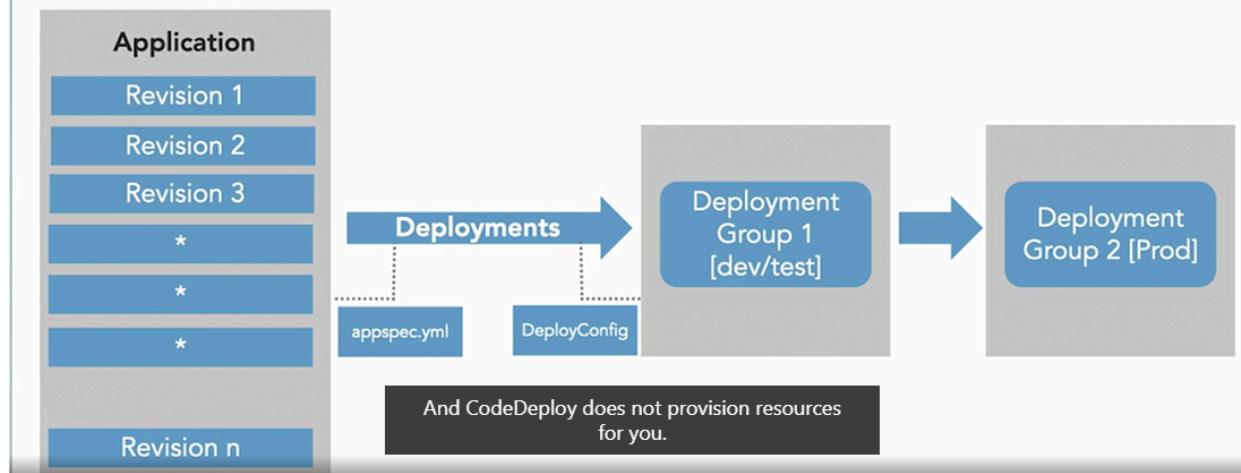
CodeDeploy: Components



CodeDeploy: Components



CodeDeploy: Components



The screenshot shows the AWS CodeCommit interface. The left sidebar has a 'Source' section with 'CodeCommit' selected, showing options like 'Getting started', 'Repositories', 'Approval rule templates', 'Artifacts', 'Build', 'Deploy', 'Pipeline', and 'Settings'. The main content area is titled 'Repositories' with a 'Create repository' button. It lists a single repository named 'cicd-repo' with details: Last modified 6 hours ago, Clone URL (HTTPS, SSH, HTTPS (GRC)).

Instance Setup

The screenshot shows the AWS EC2 Instances dashboard. The left sidebar has sections for 'Events', 'Tags', 'Limits', 'Instances' (with 'Instances' highlighted), 'Instance Types', 'Launch Templates', 'Spot Requests', 'Savings Plans', 'Reserved Instances', and 'Dedicated Hosts'. The main content area is titled 'Resources' and shows a summary of resources in the US East (N. Virginia) Region: Instances (running) 0, Dedicated Hosts 0, Elastic IPs 0, Instances 0, Key pairs 4, Load balancers 1, Placement groups 0, Security groups 6, Snapshots 1, Volumes 0. A call-to-action at the bottom says 'and launch a new instance.'

Step 1: Choose an Amazon Machine Image (AMI)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

Search for an AMI by entering a search term e.g. "Windows"

Quick Start

My AMIs	Amazon Linux
AWS Marketplace	Free tier eligible
Community AMIs	

Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-048f6ed62451373d9 (64-bit x86) / ami-00315de4391ce4f6d (64-bit Arm)

Amazon Linux 2 comes with five years support. It provides Linux kernel 4.14 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras. This AMI is the successor of the Amazon Linux AMI that is

Select (button circled in yellow)

64-bit (x86) (radio button selected)

64-bit (Arm)

So here, I'm taking this Amazon Linux 2 AMI.

Step 2: Choose an Instance Type

Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
t2	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="checkbox"/> t2	t2.micro (Free tier eligible)	1	1	EBS only	-	Low to Moderate	Yes
t2						Low to Moderate	Yes
t2						Low to Moderate	Yes
t2						Low to Moderate	Yes

Micro instances are eligible for the AWS free usage tier. For the first 12 months following your AWS sign-up date, you get up to 750 hours of micro instances each month. When your free usage tier expires or if your usage exceeds the free tier restrictions, you pay standard, pay-as-you-go service rates.

Learn more about free usage tier eligibility and restrictions

Cancel **Previous** **Review and Launch** (button highlighted with orange box)

And let's go ahead with T2 micro instances.

Next: Configure Instance Details (button highlighted with orange box)

Step 3: Configure Instance Details

Placement group	<input type="checkbox"/> Add instance to placement group
Capacity Reservation	<input type="button" value="Open"/>
Domain join directory	<input type="text" value="No directory"/> <input type="button" value="Create new directory"/>
IAM role	<input type="text" value="None"/> <input type="button" value="Create new IAM role"/>
Shutdown behavior	<input type="button" value="Stop"/>
Stop - Hibernate behavior	<input type="checkbox"/> Enable hibernation as an additional stop behavior
Enable termination protection	<input type="checkbox"/> Protect against accidental termination

Cancel **Previous** **Review and Launch** (button highlighted with orange box) **Next: Add Storage**

So let's go ahead and create an IAM role

AWS Services ▾ Search for services, features, marketplace products, and docs [Alt+S] dipalik Global Support

Create role

Attach permissions policies

Choose one or more policies to attach to your new role.

Create policy

Filter policies ▾ s3re Showing 1 result

	Policy name	Used as
<input checked="" type="checkbox"/>	AmazonS3ReadOnlyAccess	None

* Required And we will add S3 read-only permission to this role. Cancel Previous Next: Tags

Review Policy Trust Relationship Conditions

Create role

Review

Provide the required information below and review this role before you create it.

Role name* EC2RoleforCodeDeploy

Use alphanumeric and '+-, @-' characters. Maximum 64 characters.

Role description Allows EC2 instances to call AWS services on your behalf.

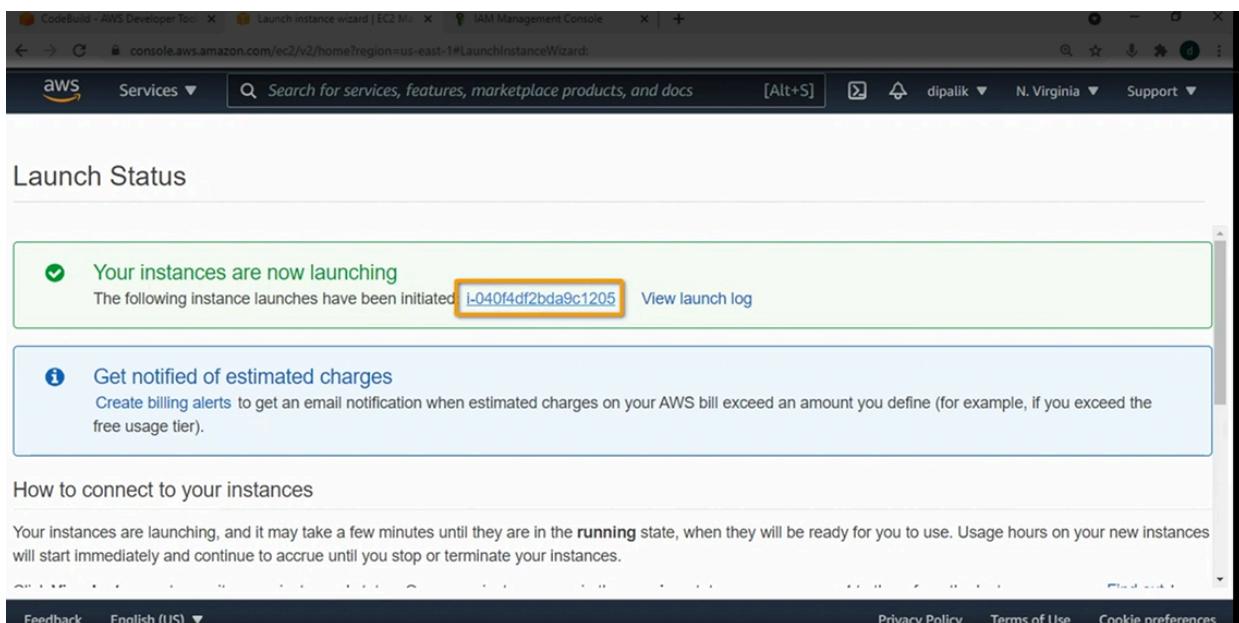
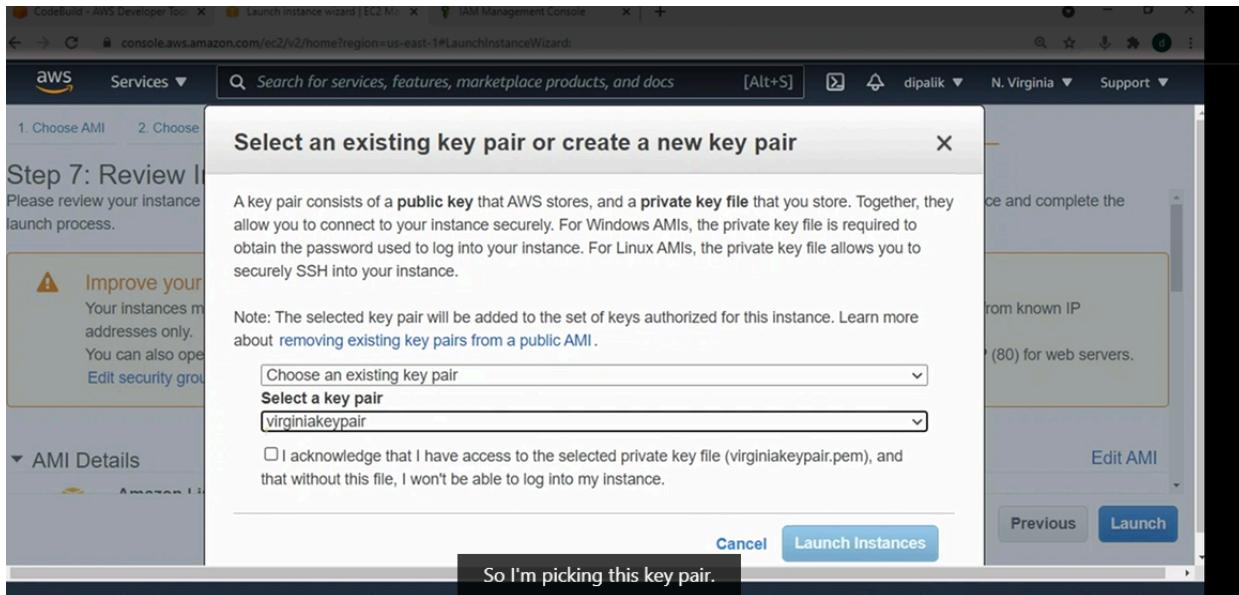
Maximum 1000 characters. Use alphanumeric and '+-, @-' characters.

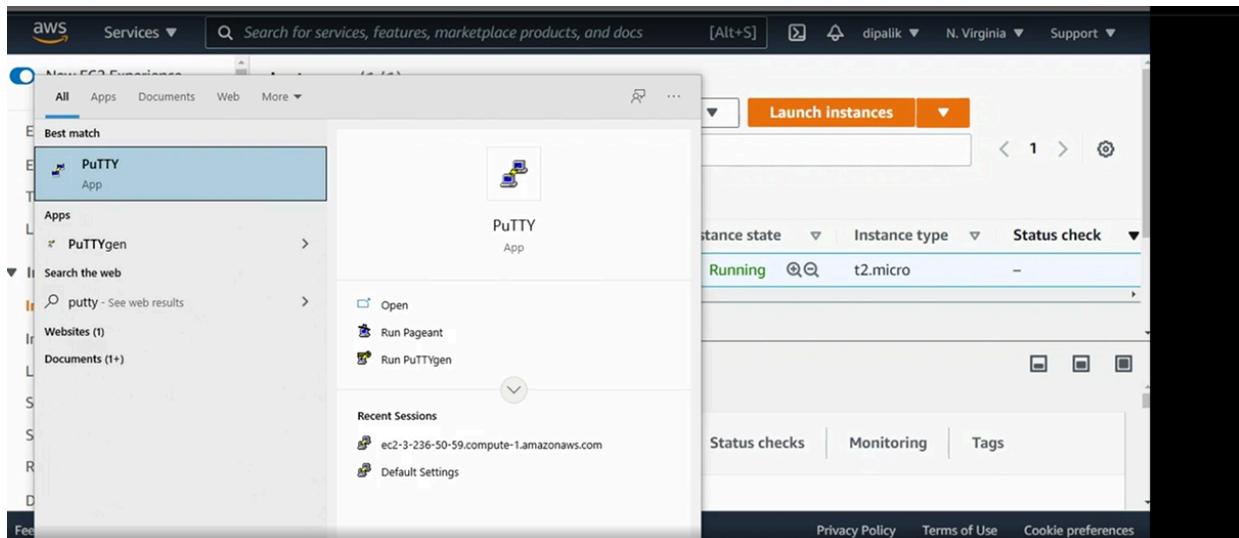
Trusted entities AWS service: ec2.amazonaws.com

Required We can give name as EC2RoleforCodeDeploy. Cancel Previous Create role

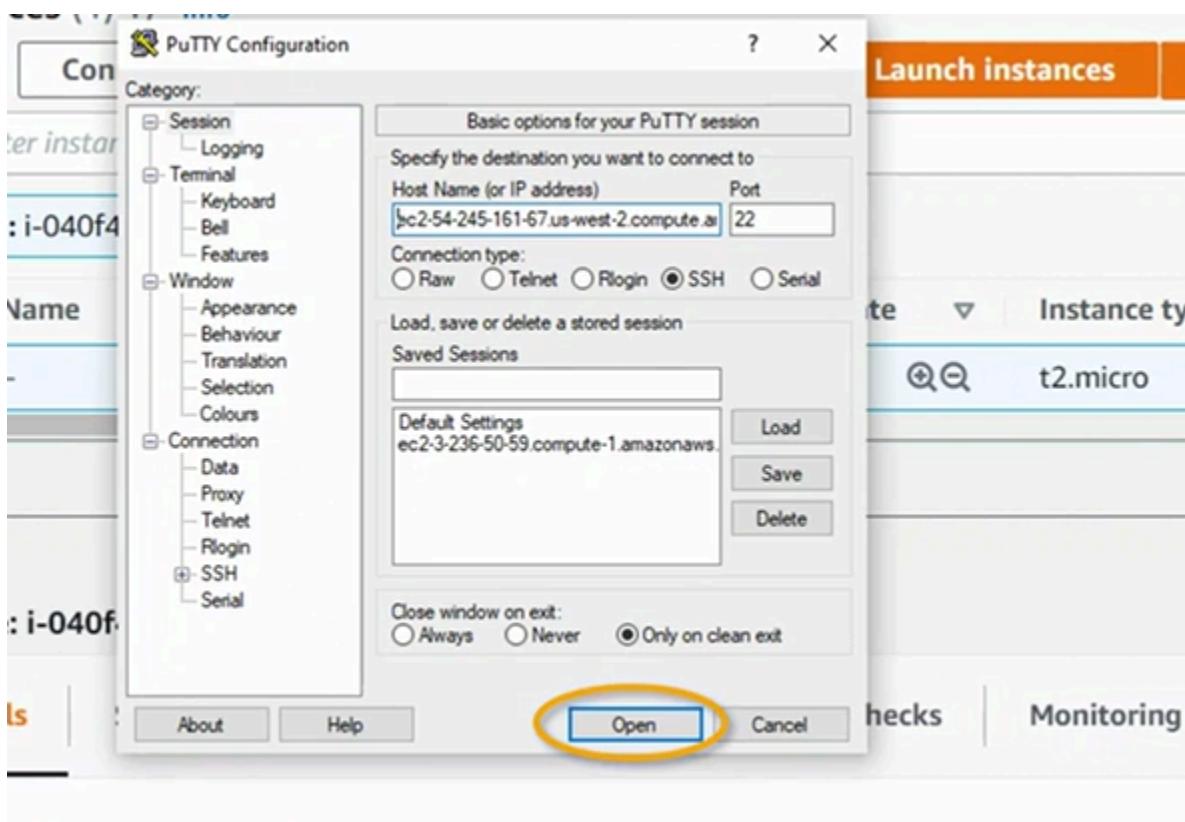
The screenshot shows the AWS Launch Instance Wizard at Step 3: Configure Instance Details. In the IAM role dropdown, the option 'EC2RoleforCodeDeploy' is selected, highlighted with a blue background. Other options visible in the dropdown include 'None', 'EC2RolesDDSNSSQS', 'ecsInstanceRole', and 'meetrole'. The IAM Management Console tab is visible in the top navigation bar.

The screenshot shows the AWS Launch Instance Wizard at Step 7: Review Instance Launch. A yellow callout box highlights a warning message: "⚠ Improve your instances' security. Your security group, launch-wizard-5, is open to the world." It advises updating security group rules to allow access from known IP addresses only and notes that additional ports can be opened for web servers like HTTP (80). Below this, the 'Edit AMI' link is shown. The 'Launch' button is circled in yellow. The IAM Management Console tab is visible in the top navigation bar.





Putty is a SSH Client and connects using this.



AWS Services Search for services, features, marketplace products, and docs [Alt+S] dipalik N. Virginia Support

New EC2 Experience Tell us what you think

EC2 Dashboard Instances Instances New Instance Types Launch Templates Spot Requests Savings Plans Reserved Instances New Dedicated Hosts

Instances (1/1) Info

Connect Instance state Actions Launch instances

Filter instances search: i-040f4df2bda9c1205 Clear filters

Name Instance ID Instance state Instance type Status check

— i-040f4df2bda9c1205 Running t2.micro —

i-040f4df2bda9c1205 34.229.56.183 | open address 172.31.62.123

Instance state Public IPv4 DNS Private IPv4 DNS

Running Public IPv4 DNS copied

Public IPv4 DNS copied

PUTTY Configuration

Category: Features Window Appearance Behaviour Translation Selection Colours Connection Data Proxy Telnet Rlogin SSH Kex Host keys Cipher Auth TTY X11 Tunnels Bugs More bugs

Options controlling SSH authentication

Display pre-authentication banner (SSH-2 only) Bypass authentication entirely (SSH-2 only)

Authentication methods Attempt authentication using Pageant Attempt TIS or CryptoCard auth (SSH-1) Attempt "Keyboard-interactive" auth (SSH-2)

Authentication parameters Allow agent forwarding Allow attempted changes of username in SSH-2

Private key file for authentication: Browse...

Launch instances

i-040f4df2bda9c1205 172.31.62.123

Public IPv4 DNS Private IPv4 DNS

ip-172-31-62-123.ec2.internal

VS Services Search for services, features, marketplace products, and docs [Alt+S] dipalik N. Virginia Support

New EC2 Experience Tell us what you think

Dashboard Instances Instances New Instance Types Launch Templates Spot Requests Savings Plans Reserved Instances New Dedicated Hosts

Instances (1/1) Info

PUTTY Configuration

Category: Features Window Appearance Behaviour Translation Selection Colours Connection Data Proxy Telnet Rlogin SSH Kex Host keys Cipher Auth TTY X11 Tunnels Bugs More bugs

Options controlling SSH authentication

Display pre-authentication banner (SSH-2 only) Bypass authentication entirely (SSH-2 only)

Authentication methods Attempt authentication using Pageant Attempt TIS or CryptoCard auth (SSH-1) Attempt "Keyboard-interactive" auth (SSH-2)

Authentication parameters Allow agent forwarding Allow attempted changes of username in SSH-2

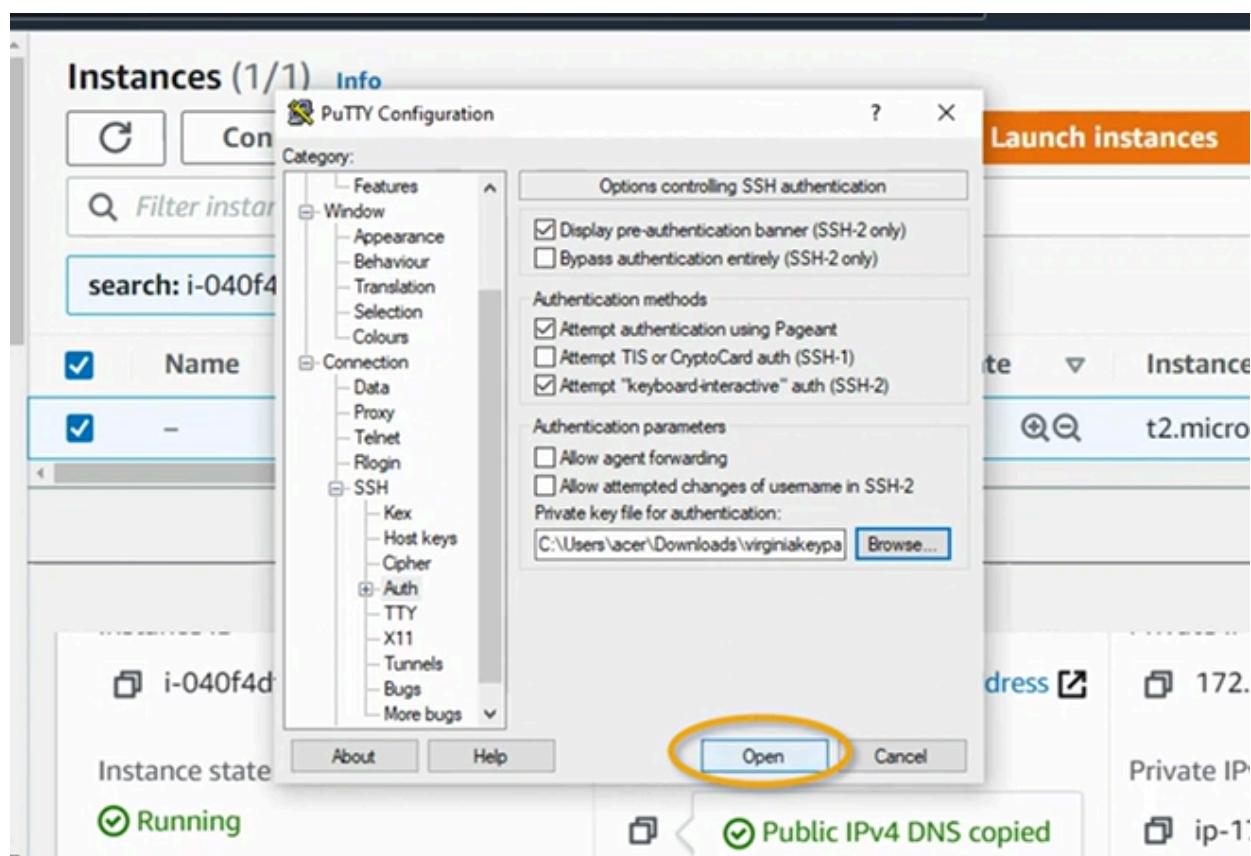
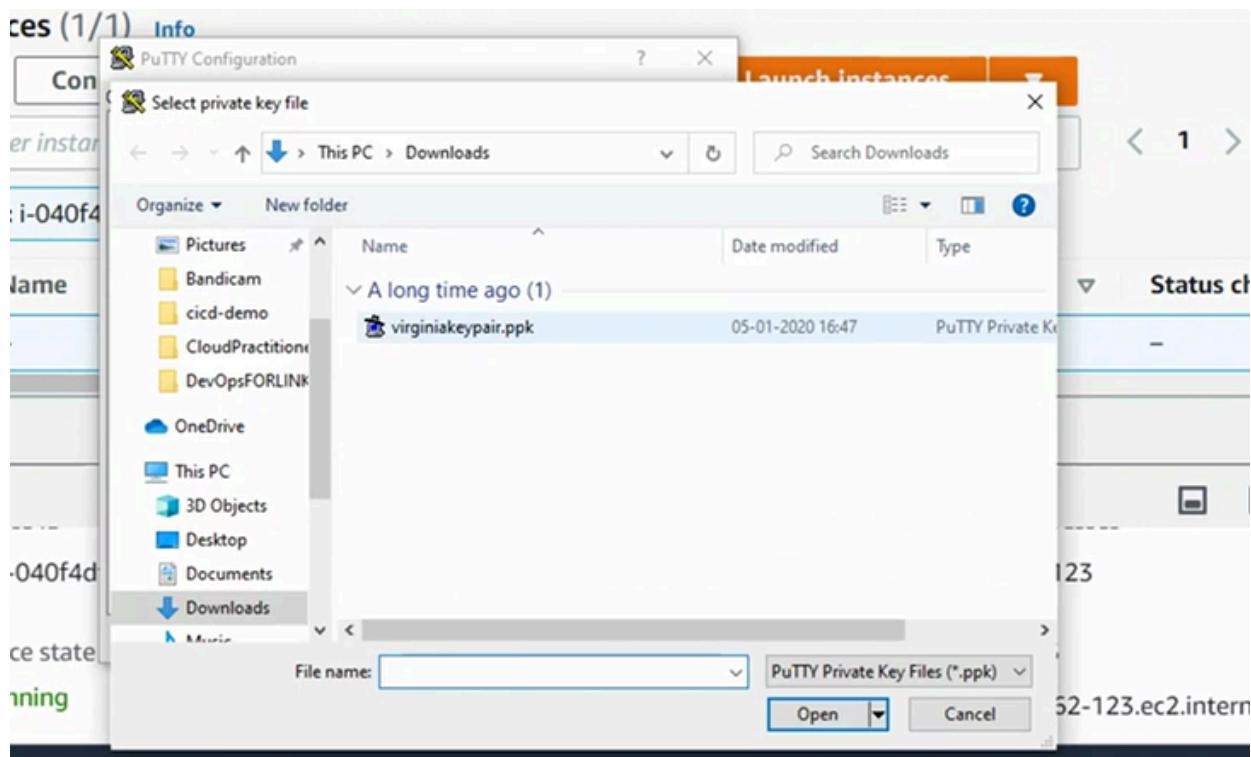
Private key file for authentication: Browse...

Launch instances

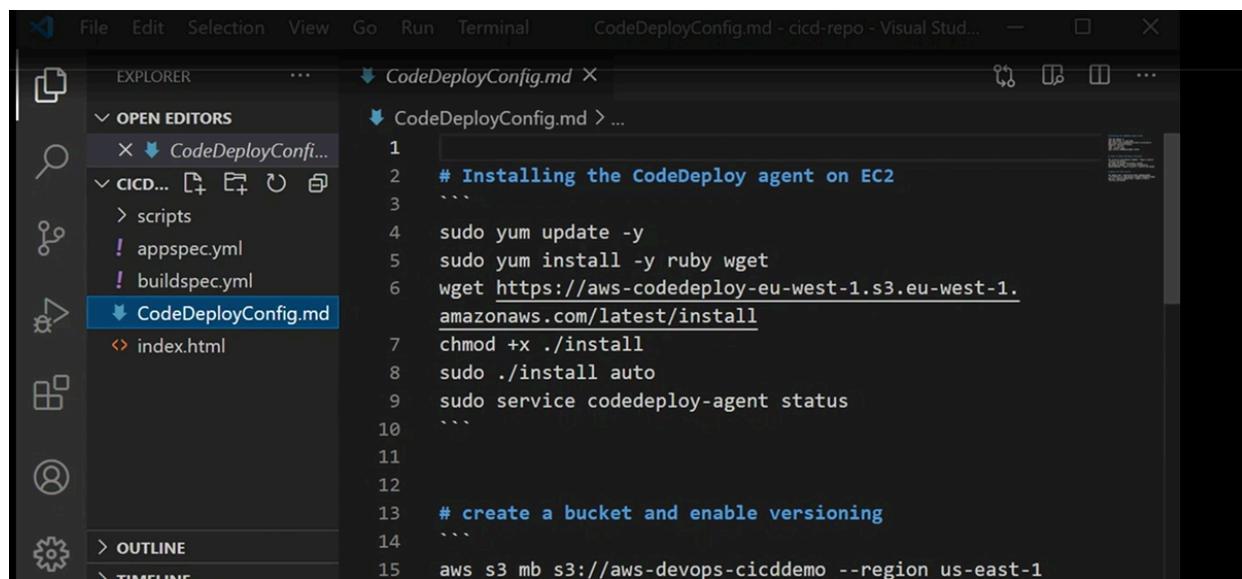
i-040f4df2bda9c1205 172.31.62.123

Public IPv4 DNS Private IPv4 DNS

ip-172-31-62-123.ec2.internal



```
ec2-user@ip-172-31-62-123:~  
login as: ec2-user  
Authenticating with public key "virginiakeypair"  
  
Amazon Linux 2 AMI  
  
https://aws.amazon.com/amazon-linux-2/  
[ec2-user@ip-172-31-62-123 ~]$
```



The screenshot shows the Visual Studio Code interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, CodeDeployConfig.md - cicd-repo - Visual Studio...
- Explorer Panel:** Shows a tree view of files and folders:
 - OPEN EDITORS: CodeDeployConfig.md
 - CICD...: scripts, appspec.yml, buildspec.yml, CodeDeployConfig.md (selected)
 - index.html
- Code Editor:** Displays the contents of the CodeDeployConfig.md file:

```
1 # Installing the CodeDeploy agent on EC2
2 ...
3
4 sudo yum update -y
5 sudo yum install -y ruby wget
6 wget https://aws-codedeploy-eu-west-1.s3.eu-west-1.amazonaws.com/latest/install
7 chmod +x ./install
8 sudo ./install auto
9 sudo service codedeploy-agent status
10 ...
11
12
13 # create a bucket and enable versioning
14 ...
15 aws s3 mb s3://aws-devops-cicddemo --region us-east-1
```

In this file, is the code to install CodeDeploy agent on EC2.

```
[ec2-user@ip-172-31-62-123 ~]
login as: ec2-user
Authenticating with public key "virginiakeypair"

      _\   _ )   Amazon Linux 2 AMI
      \_\|_|_|
https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-62-123 ~]$ sudo yum update -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
No packages marked for update
[ec2-user@ip-172-31-62-123 ~]$ sudo yum install -y ruby wget
```

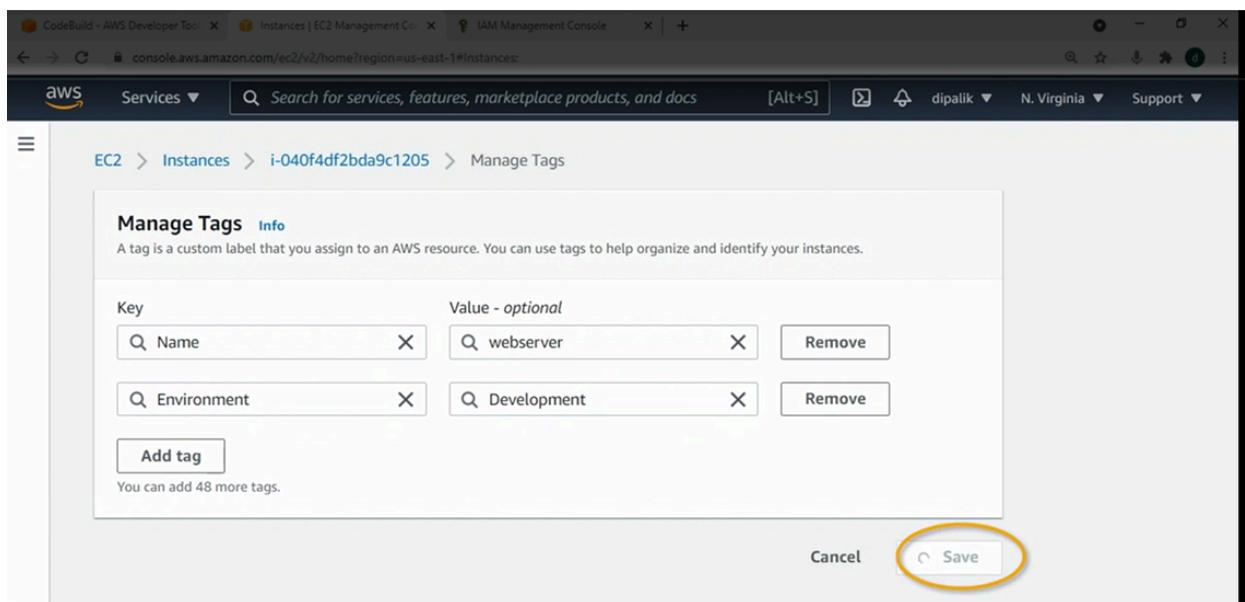
```
Complete!
[ec2-user@ip-172-31-62-123 ~]$ wget https://aws-codedeployamazonaws.com/latest/install
--2021-05-01 16:46:11--  https://aws-codedeploy-eu-west-1/latest/install
Resolving aws-codedeploy-eu-west-1.s3.eu-west-1.amazonaws.com (aws-codedeploy-eu-west-1.s3.eu-west-1.amazonaws.com) ... 52.218.96.171
Connecting to aws-codedeploy-eu-west-1.s3.eu-west-1.amazonaws.com (aws-codedeploy-eu-west-1.s3.eu-west-1.amazonaws.com)|52.218.96.171|:443...
HTTP request sent, awaiting response... 200 OK
Length: 17231 (17K) []
Saving to: 'install'

100%[=====] 17,231
2021-05-01 16:46:11 (249 KB/s) - 'install' saved [17231/17231]

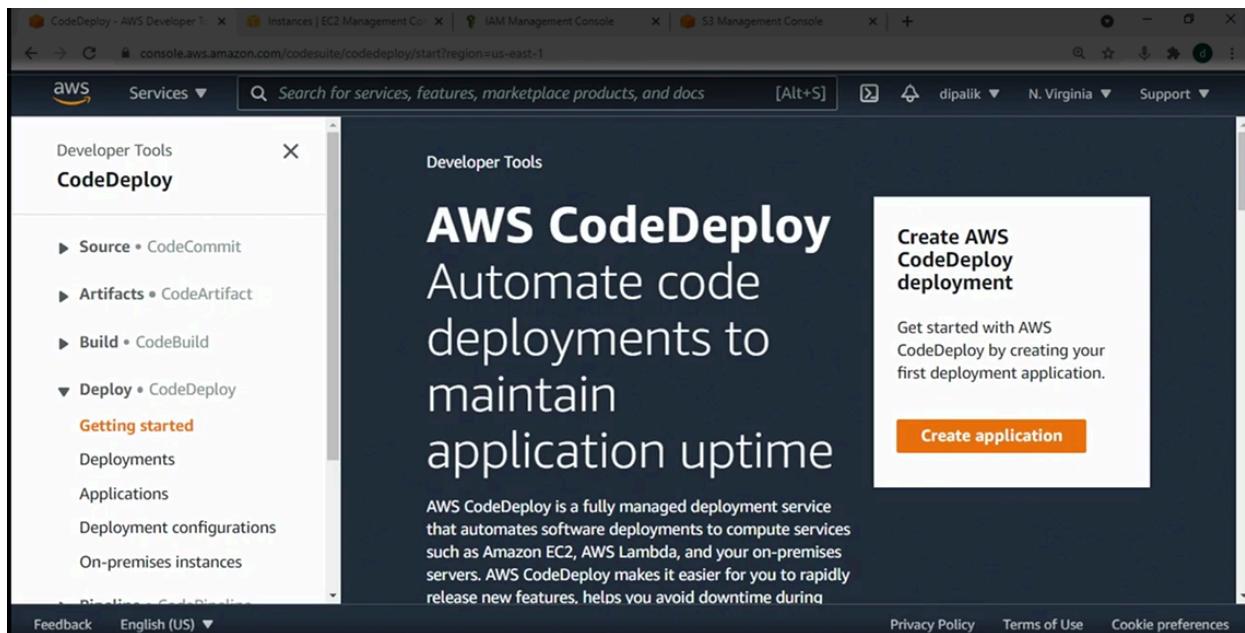
[ec2-user@ip-172-31-62-123 ~]$ chmod +x ./install
[ec2-user@ip-172-31-62-123 ~]$ sudo ./install auto
```

```
plete!
, [2021-05-01T16:46:32.157504 #3367]  INFO -- : Update check complete.
, [2021-05-01T16:46:32.157600 #3367]  INFO -- : Stopping updater.
[ec2-user@ip-172-31-62-123 ~]$ sudo service codedeploy-agent status
The AWS CodeDeploy agent is running as PID 3452
```

Next go to the EC2 instance, and add tags.



First Deployment



Screenshot of the AWS CodeDeploy 'Create application' configuration page:

Application configuration

Application name
Enter an application name
 100 character limit

Compute platform
Choose a compute platform

Create application

Screenshot of the AWS CodeDeploy 'Applications' page after creating the application:

Application created
In order to create a new deployment, you must first create a deployment group.

Developer Tools > CodeDeploy > Applications > CodeDeployDemo

CodeDeployDemo

Application details

Name	CodeDeployDemo	Compute platform
		EC2/On-premises

Deployments | Deployment groups | Revisions

Notify | Delete application

Feedback English (US) ▾ Privacy Policy Terms of Use Cookie preferences

Salesforce Services ▾ Search for services, features, marketplace products, and docs [Alt+S] dipalik N. Virginia Support

Developer Tools CodeDeploy

Source • CodeCommit
Artifacts • CodeArtifact
Build • CodeBuild
Deploy • CodeDeploy
Getting started
Deployments
Applications
Application
Settings

CodeDeployDemo EC2/On-premises

Deployments Deployment groups Revisions

Deployment groups View details Edit Create deployment group

Name Status Last attempted Last successful Trigger

AWS Services ▾ Search for services, features, marketplace products, and docs [Alt+S] dipalik N. Virginia Support

Application

Application
CodeDeployDemo
Compute type
EC2/On-premises

Deployment group name
Enter a deployment group name
MyDevInstances 100 character limit

Service role

The screenshot shows the AWS IAM Management Console dashboard. On the left, the navigation menu is visible with the 'Roles' option highlighted and circled in orange. The main content area displays the 'IAM dashboard' with sections for 'Sign-in URL for IAM users in this account', 'IAM resources' (listing 2 users, 77 roles, 1 user group, 0 identity providers, and 53 customer managed policies), 'Security alerts' (warning about access keys and MFA), and 'Best practices'. The right sidebar contains links for 'Additional information', 'Tools', 'Quick links', and 'Related services'.

The screenshot shows the process of creating a new IAM role. The URL in the browser is [https://console.aws.amazon.com/iam/home?region=us-east-1#/roles\\$new?step=permissions&selectedService=CodeDeploy&selectedUseCase=CodeDeploy](https://console.aws.amazon.com/iam/home?region=us-east-1#/roles$new?step=permissions&selectedService=CodeDeploy&selectedUseCase=CodeDeploy). The page displays a table with one result, showing a policy named 'AWSCodeDeployRole' with 'None' used as and a description 'Provides CodeDeploy service access to expand...'. At the bottom, there are buttons for 'Cancel', 'Previous', and 'Next: Tags', with 'Next: Tags' highlighted and circled in orange. A note at the top states: 'The type of role that you selected requires the following policy.'

Screenshot of the AWS IAM Management Console showing the 'Create role' step. The role name is 'CodeDeployRole'. The role description is 'Allows CodeDeploy to call AWS services such as Auto Scaling on your behalf.' The trusted entity is 'AWS service: codedeploy.amazonaws.com'. The 'Create role' button is highlighted with a yellow circle.

Create role

Review

Provide the required information below and review this role before you create it.

Role name* Use alphanumeric and '+-=_,@-_.' characters. Maximum 64 characters.

Role description Maximum 1000 characters. Use alphanumeric and '+-=_,@-_.' characters.

Trusted entities AWS service: codedeploy.amazonaws.com

* Required

Cancel Previous Create role

Feedback English (US) ▾ Privacy Policy Terms of Use Cookie preferences

Screenshot of the AWS CodeDeploy console showing the 'Create deployment group' step. The deployment group name is 'MyDevInstances'. The service role is 'CodeDeployRole' (arn:aws:iam::167771397877:role/CodeDeployRole). The deployment type is 'In place'.

Enter a deployment group name
MyDevInstances 100 character limit

Service role
Enter a service role with CodeDeploy permissions that grants AWS CodeDeploy access to your target instances.
CodeDeployRole arn:aws:iam::167771397877:role/CodeDeployRole

Deployment type
Choose how to deploy your application
In place Blue/green

Screenshot of the AWS CodeDeploy console showing deployment type configuration.

The search bar at the top contains the ARN: `arn:aws:iam::167771397877:role/CodeDeployRole`.

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.

Screenshot of the AWS CodeDeploy console showing environment configuration.

The search bar at the top contains the ARN: `arn:aws:iam::167771397877:role/CodeDeployRole`.

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"/>	<input type="text"/>
<input type="button"/>	<input type="button"/>
<input type="button" value="Remove tag"/>	

Feedback English (US) ▾ Privacy Policy Terms of Use Cookie preferences

Screenshot of the AWS CloudFormation console showing the configuration of a deployment group for Amazon EC2 instances.

The search bar shows "Search for services, features, marketplace products, and docs".

The navigation bar includes "Services ▾", "dipalik", and "N. V."

The main content area shows:

- A checkbox for "Amazon EC2 Auto Scaling groups" which is unchecked.
- A checked checkbox for "Amazon EC2 instances". Below it, it says "0 unique matched instances. Click here for details" with a link icon.
- A note: "You can add up to three groups of tags for EC2 instances to this deployment group."
- Two definitions:
 - 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.
- A "Tag group 1" section:
 - Key: "Environment" (highlighted with a blue border).
 - Value - optional: "Development" (highlighted with a blue border).
 - Buttons: "Remove tag" and "+ Add tag group".

Screenshot of the AWS CloudFormation console showing the configuration of a deployment group for Amazon EC2 instances.

The search bar shows "Search for services, features, marketplace products, and docs".

The navigation bar includes "Services ▾", "dipalik", "N. Virginia", and "Support ▾".

The main content area shows:

- A checkbox for "Amazon EC2 Auto Scaling groups" which is unchecked.
- A checked checkbox for "Amazon EC2 instances". Below it, it says "1 unique matched instance. Click here for details" with a link icon.
- A note: "You can add up to three groups of tags for EC2 instances to this deployment group."
- Two definitions:
 - 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.
- A "Tag group 1" section:
 - Key: "Environment" (highlighted with a blue border) and Value: "Development" (highlighted with a blue border).
 - Buttons: "Add tag" and "+ Add tag group".
- A checkbox for "On-premises instances" which is unchecked.

Screenshot of the AWS EC2 Instances page showing deployment group configuration.

Amazon EC2 instances
1 unique matched instance 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
Environment	Development

Add tag Remove tag

+ Add tag group

On-premises instances

Screenshot of the AWS CodeDeploy Deployment Groups creation page.

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.AllAtOnce or Create deployment configuration

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

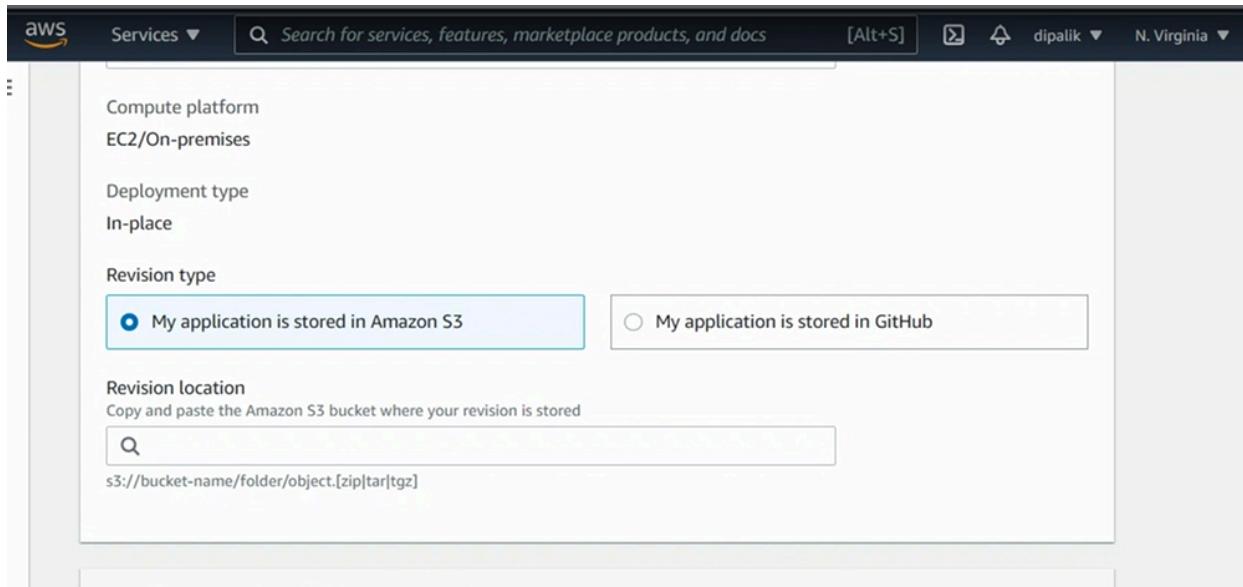
Create deployment group

The screenshot shows the AWS CodeDeploy console with a success message: "Deployment group created". The navigation bar includes links for AWS services like IAM Management Console and S3 Management Console. The left sidebar shows the CodeDeploy navigation path: Developer Tools > CodeDeploy > Applications > CodeDeployDemo > MyDevInstances. The main content area displays "MyDevInstances" with buttons for Edit, Delete, and Create deployment. A table titled "Deployment group details" provides the following information:

Deployment group name	Application name	Compute platform
MyDevInstances	CodeDeployDemo	EC2/On-premises
Deployment type	Service role ARN	Deployment configuration
In-place	arn:aws:iam::167771397877:role/CodeDeployRole	CodeDeployDefault.AllAtOnce

This screenshot is identical to the one above, showing the successful creation of a deployment group named "MyDevInstances". The "Create deployment" button is highlighted with a yellow oval. The deployment group details table is as follows:

Deployment group name	Application name	Compute platform
MyDevInstances	CodeDeployDemo	EC2/On-premises
Deployment type	Service role ARN	Deployment configuration
In-place	arn:aws:iam::167771397877:role/CodeDeployRole	CodeDeployDefault.AllAtOnce



The screenshot shows a code editor interface with the following details:

- EXPLORER** sidebar: Shows the project structure with files like "CodeDeployConfig.md", "CodeDeployConfig... (closed)", "scripts", "appspec.yml", "buildspec.yml", "index.html", and "CICD-REPO".
- OPEN EDITORS** sidebar: Shows the currently open file "CodeDeployConfig.md".
- Code Editor Area**: Displays the following CloudFormation configuration (line numbers 8-20):

```
8  sudo ./install auto
9  sudo service codedeploy-agent status
10 ...
11
12
13 # create a bucket and enable versioning
14 ...
15 aws s3 mb s3://aws-devops-cicddemo --region us-east-1
16 --profile aws-devops
17 aws s3api put-bucket-versioning --bucket
18 aws-devops-cicddemo --versioning-configuration
19 Status=Enabled --region us-east-1 --profile aws-devops
20 ...
```

Screenshot of the AWS IAM Management Console showing the Access keys page for a user named "cicd-user".

The left sidebar shows the navigation menu under "Identity and Access Management (IAM)".

The main content area displays the "Access keys" section with the following text:

Use access keys to make programmatic calls to AWS from the AWS CLI, Tools for PowerShell, AWS SDKs, or direct AWS API calls. You can have a maximum of two access keys (active or inactive) at a time.

For your protection, you should never share your secret keys with anyone. As a best practice, we recommend frequent key rotation.

If you lose or forget your secret key, you cannot retrieve it. Instead, create a new access key and make the old key inactive. Learn more

A green success message box is present, stating: "Access key AKIASOD7UL32UBP2RPKG deactivated".

A "Create access key" button is available.

A table lists the access keys:

Access key ID	Created	Last used	Status
AKIASOD7UL32UBP2RPKG	2021-05-01 23:52 UTC+0530	2021-0...	Inactive Make active

Feedback English (US) ▾ Privacy Policy Terms of Use Cookie preferences

Screenshot of the AWS IAM Management Console showing the Access keys page for a user named "cicd-user".

The left sidebar shows the navigation menu under "Identity and Access Management (IAM)".

The main content area displays the "Access keys" section with the following text:

Use access keys to make programmatic calls to AWS from the AWS CLI, Tools for PowerShell, AWS SDKs, or direct AWS API calls. You can have a maximum of two access keys (active or inactive) at a time.

For your protection, you should never share your secret keys with anyone. As a best practice, we recommend frequent key rotation.

If you lose or forget your secret key, you cannot retrieve it. Instead, create a new access key and make the old key inactive. Learn more

A green success message box is present, stating: "Access key AKIASOD7UL32UBP2RPKG deactivated".

A "Create access key" button is highlighted with a yellow circle.

A table lists the access keys:

Access key ID	Created	Last used	Status
AKIASOD7UL32UBP2RPKG	2021-05-01 23:52 UTC+0530	2021-0...	Inactive Make active

SSH keys for AWS CodeCommit

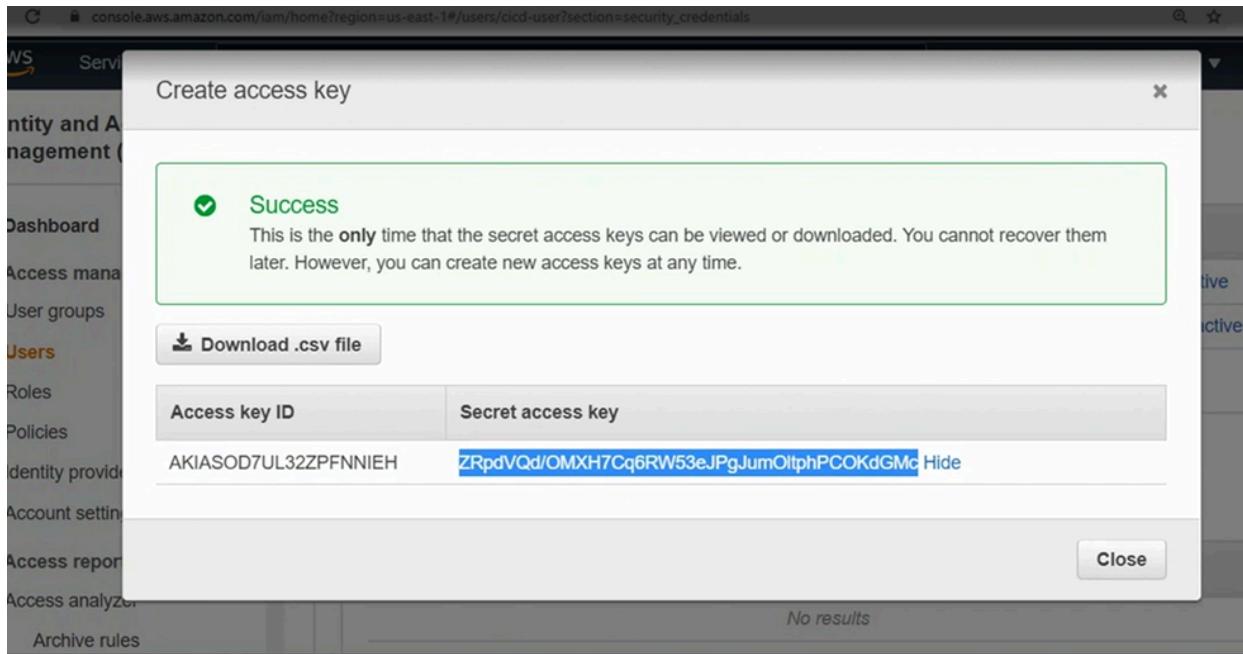
Use SSH public keys to authenticate access to AWS CodeCommit repositories. Learn more

An "Upload SSH public key" button is present.

A table lists the SSH keys:

SSH key ID	Uploaded	Status

Feedback English (US) ▾ Privacy Policy Terms of Use Cookie preferences



```
acer@LAPTOP-KLLSIHQ0 MINGW64 ~/cicd-demo/cicd-repo (master)
$ aws configure --profile aws-devops
AWS Access Key ID [*****RPKG]: AKIASOD7UL32ZPFNNIEH
AWS Secret Access Key [*****c0iZ]: ZRpdVQd/OMXH7Cq6RW53eJPgJumOlphPC
OKdGMc
Default region name [us-east-1]: us-east-1
Default output format [aws configure --list]: json
acer@LAPTOP-KLLSIHQ0 MINGW64 ~/cicd-demo/cicd-repo (master)
$
```

The screenshot shows the VS Code interface with the Explorer sidebar on the left and a terminal window on the right. The terminal window displays the following AWS CLI commands:

```
CodeDeployConfig.md X
CodeDeployConfig.md > # create a bucket and enable versioning
8 sudo ./install auto
9 sudo service codedeploy-agent status
10 ...
11
12
13 # create a bucket and enable versioning
14 ...
15 aws s3 mb s3://aws-devops-cicddemo --region us-east-1
--profile aws-devops
16 aws s3api put-bucket-versioning --bucket
aws-devops-cicddemo --versioning-configuration
Status=Enabled --region us-east-1 --profile aws-devops
17 ...
18
19 # deploy the files into S3
20 ...
```

The screenshot shows a terminal window with the following AWS CLI session:

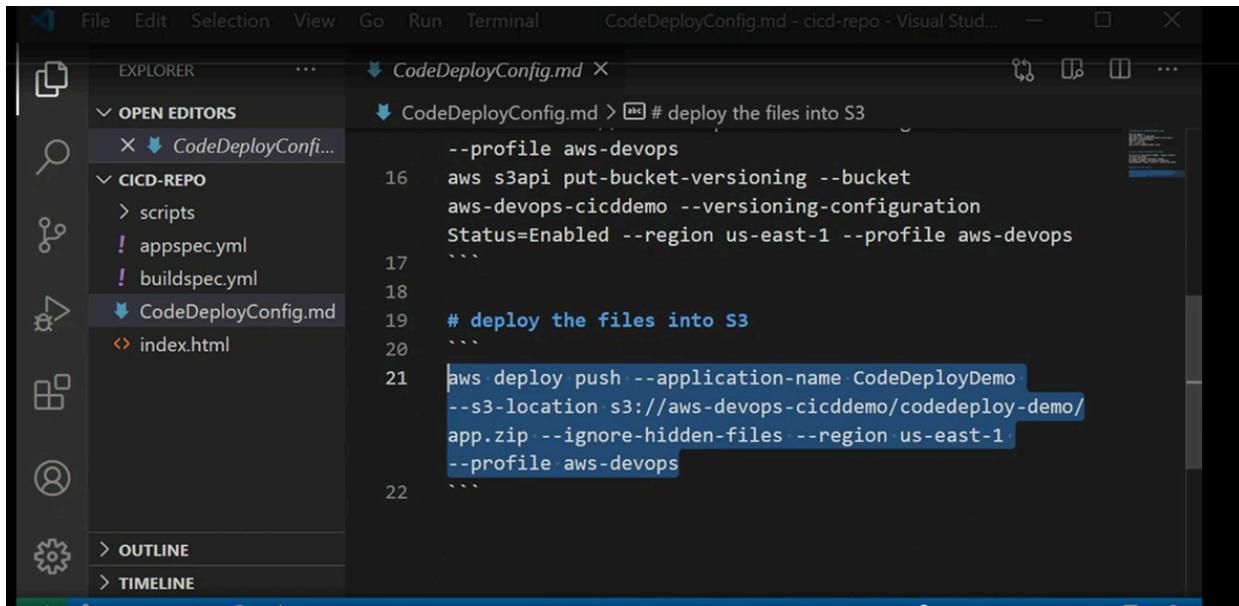
```
acer@LAPTOP-KLLSIHQ0 MINGW64 ~/cicd-demo/cicd-repo (master)
$ aws configure --profile aws-devops
AWS Access Key ID [*****RPKG]: AKIASOD7UL32ZPFNNIEH
AWS Secret Access Key [*****coiZ]: ZRpDVQd/OMXH7Cq6RW53eJPgJum0ltphPC
OKdGMC
Default region name [us-east-1]: us-east-1
Default output format [aws configure --list]: json
acer@LAPTOP-KLLSIHQ0 MINGW64 ~/cicd-demo/cicd-repo (master)
$ aws s3 mb s3://aws-devops-cicddemo --region us-east-1 --profile aws-devops
make_bucket: aws-devops-cicddemo
acer@LAPTOP-KLLSIHQ0 MINGW64 ~/cicd-demo/cicd-repo (master)
$ aws s3api put-bucket-versioning --bucket aws-devops-cicddemo --versioning-configuration
Status=Enabled --region us-east-1 --profile aws-devops
```

The screenshot shows the AWS S3 Management Console interface. On the left, there's a sidebar with options like 'Buckets', 'Access Points', 'Object Lambda Access Points', 'Batch Operations', 'Access analyzer for S3', 'Block Public Access settings for this account', 'Storage Lens', 'Dashboards', 'AWS Organizations settings', 'Feature spotlight', and 'AWS Marketplace for S3'. The main area is titled 'Amazon S3' and 'Buckets (11)'. It includes a search bar ('Find buckets by name') and a table with columns: Name, AWS Region, Access, and Creation date. The first row, 'aws-devops-cicddemo', is highlighted with a yellow border.

Name	AWS Region	Access	Creation date
aws-devops-cicddemo	US East (N. Virginia) us-east-1	Objects can be public	May 2, 2021, 00:06:25 (UTC+05:30)
aws-swaggers-files	US East (N. Virginia) us-east-1	Public	December 7, 2018, 22:30:22 (UTC+05:30)
awssomedomain.tk	US West (N. California) us-west-1	Public	February 2, 2021, 17:04:43 (UTC+05:30)
awssatraining	US East (N. Virginia) us-east-1	Public	June 15, 2020, 21:25:57 (UTC+05:30)
awsswaggers	US East (N. Virginia) us-east-1	Bucket and objects not public	December 7, 2018, 10:06:32 (UTC+05:30)

The screenshot shows a Windows File Explorer window with the path 'acer > cicd-demo > cicd-repo'. The left sidebar shows 'Quick access' with various folders like Desktop, Downloads, Documents, Pictures, and OneDrive. The main area displays a list of files and folders: '.git', 'scripts', 'appspec.yml', 'buildspec.yml', 'Co...', 'in...', and 'ind...'. The file 'appspec.yml' is selected and highlighted with a yellow border.

`appspec.yml` is required by CodeDeploy to deploy our application.

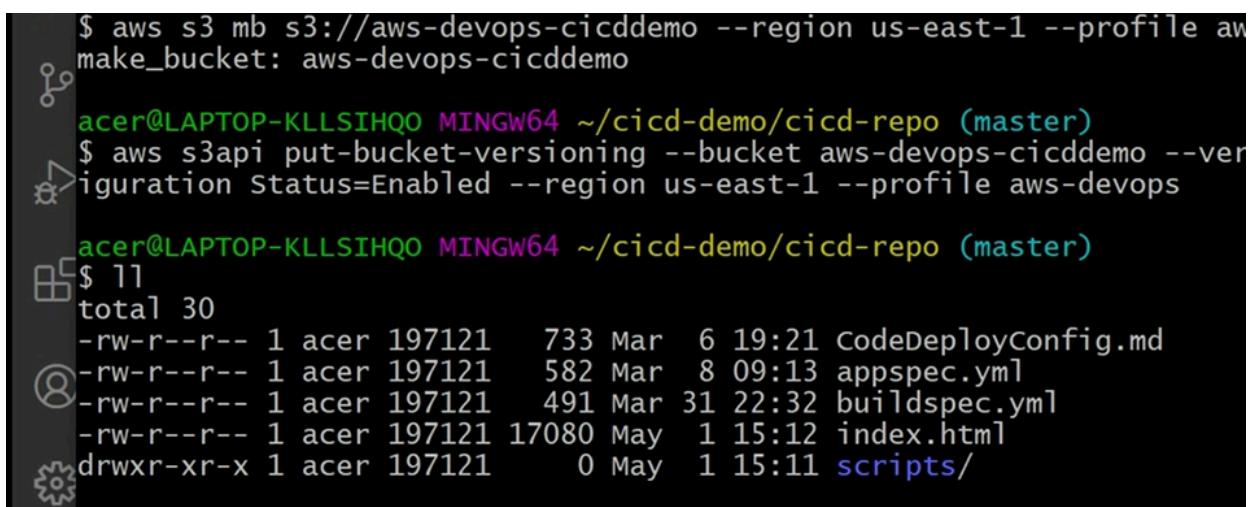


```
CodeDeployConfig.md - cicd-repo - Visual Studio Code
File Edit Selection View Go Run Terminal CodeDeployConfig.md - cicd-repo - Visual Studio Code

EXPLORER ... CodeDeployConfig.md ×
OPEN EDITORS ...
CICD-REPO ...
scripts
appspec.yml
buildspec.yml
CodeDeployConfig.md
index.html

CodeDeployConfig.md > # deploy the files into S3
--profile aws-devops
16 aws s3api put-bucket-versioning --bucket
aws-devops-cicddemo --versioning-configuration
Status=Enabled --region us-east-1 --profile aws-devops
```
17
18
19 # deploy the files into S3
```
19 # deploy the files into S3
```
21 aws deploy push --application-name CodeDeployDemo
--s3-location s3://aws-devops-cicddemo/codedeploy-demo/
app.zip --ignore-hidden-files --region us-east-1
--profile aws-devops
```
22

> OUTLINE
> TIMELINE
```



```
$ aws s3 mb s3://aws-devops-cicddemo --region us-east-1 --profile aws-devops
make_bucket: aws-devops-cicddemo

acer@LAPTOP-KLLSIHQ0 MINGW64 ~/cicd-demo/cicd-repo (master)
$ aws s3api put-bucket-versioning --bucket aws-devops-cicddemo --versioning-configuration Status=Enabled --region us-east-1 --profile aws-devops

acer@LAPTOP-KLLSIHQ0 MINGW64 ~/cicd-demo/cicd-repo (master)
$ ll
total 30
-rw-r--r-- 1 acer 197121    733 Mar  6 19:21 CodeDeployConfig.md
-rw-r--r-- 1 acer 197121    582 Mar  8 09:13 appspec.yml
-rw-r--r-- 1 acer 197121    491 Mar 31 22:32 buildspec.yml
-rw-r--r-- 1 acer 197121 17080 May  1 15:12 index.html
drwxr-xr-x 1 acer 197121      0 May  1 15:11 scripts/
```

```

acer@LAPTOP-KLLSIHQ0 MINGW64 ~/cicd-demo/cicd-repo (master)
$ ll
total 30
-rw-r--r-- 1 acer 197121 733 Mar  6 19:21 CodeDeployConfig.md
-rw-r--r-- 1 acer 197121 582 Mar  8 09:13 appspec.yml
-rw-r--r-- 1 acer 197121 491 Mar 31 22:32 buildspec.yml
-rw-r--r-- 1 acer 197121 17080 May  1 15:12 index.html
drwxr-xr-x 1 acer 197121 0 May  1 15:11 scripts/

>acer@LAPTOP-KLLSIHQ0 MINGW64 ~/cicd-demo/cicd-repo (master)
$ aws deploy push --application-name CodeDeployDemo --s3-location s3://aws-devops-cicddemo/codedeploy-demo/app.zip --ignore-hidden-files --region us-east-1 --profile aws-devops
To deploy with this revision, run:
aws deploy create-deployment --application-name CodeDeployDemo --s3-location bucket=aws-devops-cicddemo,key=codedeploy-demo/app.zip,bundleType=zip,eTag=d65ed4711aae062a5559f90e78c6dd5f,version=E0ei.fzx7bg9vtTnF9_bv8AzDt11EUIp --deployment-group-name <deployment-group-name> --deployment-config-name <deployment-config-name> --description <description>

```

The screenshot shows the AWS S3 Management Console interface. On the left, there's a sidebar with navigation links like 'Buckets', 'Access Points', 'Storage Lens', and 'AWS Marketplace for S3'. The main area shows the 'aws-devops-cicddemo' bucket. At the top, there are tabs for 'Objects', 'Properties', 'Permissions', 'Metrics', 'Management', and 'Access Points'. The 'Objects' tab is selected. Below it, there's a table with one item:

Name	Type	Last modified	Size	Storage class
codedeploy-demo/	Folder	-	-	-

The screenshot shows the AWS CodeDeploy console with a deployment group named 'MyDevInstances'. The deployment type is set to 'In-place'. The revision type is selected as 'My application is stored in Amazon S3'. The revision location is specified as 's3://aws-devops-cicddemo/codedeploy-demo/app.zip?versionId=EOei.fzx7bg9vtTnF9_bv8AzDtl1EUIp&eTag=d65ed4711aae062a5559...'. The browser's address bar shows the URL: `console.aws.amazon.com/codesuite/codedeploy/application/CodeDeployDemo/deployments/new?deploymentGroupName=MyDevInstances®ion=us-east-1`.

This screenshot shows the continuation of the deployment configuration. The deployment type is still 'In-place'. The revision type is selected as 'My application is stored in Amazon S3'. The revision location is now explicitly typed as 's3://aws-devops-cicddemo/codedeploy-demo/app.zip?versionId=EOei.fzx7bg9vtTnF9_bv8AzDtl1EUIp&eTag=d65ed4711aae062a5559...'. The browser's address bar shows the URL: `console.aws.amazon.com/codesuite/codedeploy/application/CodeDeployDemo/deployments/new?deploymentGroupName=MyDevInstances®ion=us-east-1`.

The screenshot shows the AWS CodeDeploy console interface. At the top, there are several tabs: CodeDeploy - AWS Developer Tools, Instances | EC2 Management Console, IAM Management Console, and S3 Management Console. Below the tabs, the URL is console.aws.amazon.com/codesuite/codedeploy/application/CodeDeployDemo/deployments/new?deploymentGroupName=MyDevInstances®ion=us-east-1. The main content area has a sidebar on the left with sections like 'Source', 'Artifacts', 'Build', 'Deploy', 'Applications', 'Deployment configurations', and 'On-premises instances'. The 'Deploy' section is expanded, showing 'Getting started', 'Deployments', 'Deployment' (which is selected and highlighted in orange), 'Applications', 'Deployment configurations', and 'On-premises instances'. The main panel displays deployment options for handling file conflicts: 'Fail the deployment' (an error is reported and the deployment status is changed to Failed), 'Overwrite the content' (the file in the application revision is copied to the target location on the instance, replacing the previous file), and 'Retain the content' (the file in the application revision is not copied to the instance. The existing file is kept at the target location and treated as part of the new deployment). Below these options are two expandable sections: 'Deployment group overrides' and 'Rollback configuration overrides'. At the bottom right, there are 'Cancel' and 'Create deployment' buttons, with the 'Create deployment' button being highlighted by a yellow oval.

The screenshot shows the AWS CodeDeploy console after a deployment has been created. The URL is now console.aws.amazon.com/codesuite/codedeploy/deployments/d-5LIOI2EOA?region=us-east-1. The sidebar remains the same, with the 'Deployment' section still selected. The main panel shows a green success message: 'Success Deployment created'. Below this, the 'Deployment lifecycle events' table is displayed. The table has columns: Instance ID, Duration, Status, Most recent event, Events, and Start time. There is one row of data: Instance ID i-040f4df2bda9c1205, Duration 7 seconds, Status Succeeded (indicated by a green circle with a checkmark), Most recent event ValidateService, Events View events, and Start time May 12 12:11 (UTC). The table also includes navigation arrows and a search bar.

Check if index.html is placed onto EC2 instance,

```
ec2-user@ip-172-31-62-123:~$ cd /var/www/html
login as: ec2-user
Authenticating with public key "virginiakeypair"
Last login: Sat May  1 17:47:36 2021 from 150.129.238.17
[ec2-user@ip-172-31-62-123 ~]$ ls
index.html
[ec2-user@ip-172-31-62-123 html]$
```

Configurations

The screenshot shows the AWS CodeDeploy console interface. On the left, a navigation sidebar lists 'Build' (CodeBuild), 'Deploy' (CodeDeploy), 'Pipeline' (CodePipeline), and 'Settings'. Under 'Deploy', 'Application' is selected, and 'Getting started', 'Deployments', 'Applications', and 'Deployment configurations' are listed. The main content area is titled 'CodeDeployDemo' and shows 'Application details' for the application 'CodeDeployDemo' running on 'EC2/On-premises'. Below this, the 'Deployment groups' tab is active, showing a table with one row and a 'Create deployment group' button highlighted with a yellow oval.

CodeDeploy - AWS Developer Tools Instances | EC2 Management Console

console.aws.amazon.com/codesuite/codedeploy/applications/CodeDeployDemo/deployment-groups/new?region=us-east-1

aws Services ▾ ec2 Application

Application
CodeDeployDemo
Compute type
EC2/On-premises

Deployment group name

Enter a deployment group name
MyProdInstances
100 character limit

Service role

Feedback English (US) Privacy Policy Terms of Use Support

CodeDeploy - AWS Developer Tools Launch instance wizard | EC2 Management Console

console.aws.amazon.com/ec2/v2/home?region=us-east-1#LaunchInstanceWizard:

aws Services ▾ Search for services, features, marketplace products, and docs [Alt+S] dipalik N. Virginia Support

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

Step 2: Choose an Instance Type

Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
<input type="checkbox"/> t2	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="checkbox"/> t2	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/> t2	t2.small	1	2	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/> t2	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/> t2	t2.large	2	8	EBS only	-	Low to Moderate	Yes

Cancel Previous **Review and Launch** Next: Configure Instance Details

Feedback English (US) Privacy Policy Terms of Use Cookie preferences

CodeDeploy - AWS Developer Tools | Launch instance wizard | EC2 Management | +

console.aws.amazon.com/ec2/v2/home?region=us-east-1#LaunchInstanceWizard;

Services ▾ Search for services, features, marketplace products, and docs [Alt+S] dipalik N. Virginia Support

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances Launch into Auto Scaling Group

Purchasing option Request Spot Instances

Network

Subnet

Auto-assign Public IP Use subnet setting (Enable)

Placement group Add instance to placement group

CodeDeploy - AWS Developer Tools | Launch instance wizard | EC2 Management | +

console.aws.amazon.com/ec2/v2/home?region=us-east-1#LaunchInstanceWizard;

aws Services ▾ Search for services, features, marketplace products, and docs [Alt+S] dipalik N. Virginia Support

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

Step 3: Configure Instance Details

Placement group Add instance to placement group

Capacity Reservation

Domain join directory

IAM role

Shutdown behavior

Stop - Hibernate behavior Enable hibernation as an additional stop behavior

Enable termination protection Protect against accidental termination

The screenshot shows a code editor interface with the following details:

- EXPLORER** sidebar: Shows 'OPEN EDITORS' with 'CodeDeployConfig.md' selected, and 'CICD-REPO' with files: 'scripts', 'appspec.yml', 'buildspec.yml', 'CodeDeployConfig.md', and 'index.html'.
- CodeDeployConfig.md** content:

```
1 # Installing the CodeDeploy agent on EC2
2 # Installing the CodeDeploy agent on EC2
3 ...
4 sudo yum update -y
5 sudo yum install -y ruby wget
6 wget https://aws-codedeploy-eu-west-1.s3.eu-west-1.amazonaws.com/latest/install
7 chmod +x ./install
8 sudo ./install auto
9 sudo service codedeploy-agent status
10 ...
11 ...
12 ...
13 # create a bucket and enable versioning
14 ...
15 aws s3 mb s3://aws-devops-cicddemo --region us-east-1
```

The screenshot shows the 'Configure Instance Details' step of the AWS Launch Instance Wizard. The user has entered the following configuration:

- Enclave**: Enable
- Metadata accessible**: Enabled
- Metadata version**: V1 and V2 (token optional)
- Metadata token response hop limit**: 1
- User data**:
As text

```
sudo yum install -y ruby wget
wget https://aws-codedeploy-eu-west-1.s3.eu-west-1.amazonaws.com/latest/install
chmod +x ./install
sudo ./install auto
sudo service codedeploy-agent status
```

At the bottom, the 'Review and Launch' button is highlighted, and the 'Next: Add Storage' button is visible.

Screenshot of Step 6: Configure Security Group in the AWS Launch Instance Wizard.

The wizard is at Step 6: Configure Security Group. It shows a list of existing security groups:

ID	Name	Description
sg-fa723ab3	default	default VPC security group
sg-0a420a467e76203a1	launch-wizard-1	launch-wizard-1 created 2021-03-06T13:02:15.354+00:00
sg-037a8f4f7df4024fc	launch-wizard-2	launch-wizard-2 created 2021-04-28T11:49:24.615+00:00
sg-0007ffce88ef1db8a	launch-wizard-3	launch-wizard-3 created 2021-04-28T12:13:37.106+00:00
sg-0cab94f07e3f2c5d7	launch-wizard-4	launch-wizard-4 created 2021-04-29T09:53:10.164+00:00
sg-04f0c5f42d79e7c28	launch-wizard-5	launch-wizard-5 created 2021-05-01T22:12:53.588+00:00

Inbound rules for sg-04f0c5f42d79e7c28 (Selected security groups: sg-04f0c5f42d79e7c28)

Buttons: Cancel, Previous, **Review and Launch**

Screenshot of Step 7: Review Instances in the AWS Launch Instance Wizard.

The wizard is at Step 7: Review Instances. A modal dialog is open: "Select an existing key pair or create a new key pair".

The modal contains the following text:

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about [removing existing key pairs from a public AMI](#).

Form fields:

- Choose an existing key pair: virginlakeypair
- Select a key pair: virginlakeypair

Checkboxes:

- I acknowledge that I have access to the selected private key file (virginlakeypair.pem), and that without this file, I won't be able to log into my instance. (checked)

Buttons: Cancel, **Launch Instances**, Previous, Launch

CodeDeploy - AWS Developer Tools | Launch instance wizard | EC2 Management Console | +

console.aws.amazon.com/ec2/v2/home?region=us-east-1#instances;search=i-083e1eb840d9ed40e;sort=instanceId

aws Services Search for services, features, marketplace products, and docs [Alt+S] dipalik N. Virginia Support

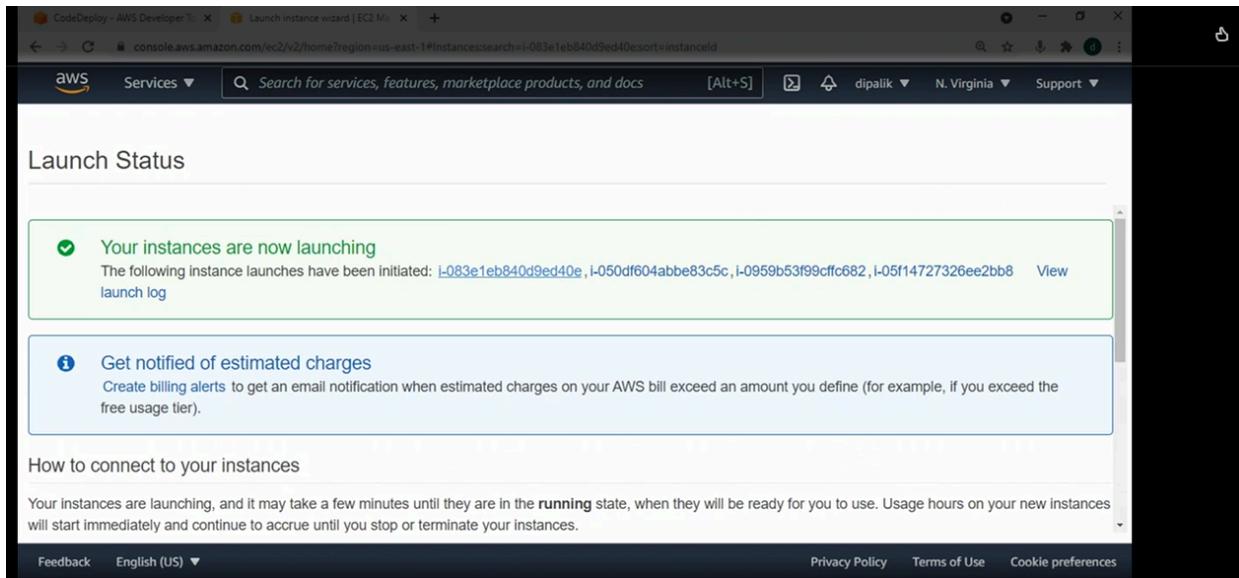
Launch Status

Your instances are now launching
The following instance launches have been initiated: [i-083e1eb840d9ed40e](#), [i-050df604abbe83c5c](#), [i-0959b53f99ccfc682](#), [i-05f14727326ee2bb8](#) View launch log

Get notified of estimated charges
Create billing alerts to get an email notification when estimated charges on your AWS bill exceed an amount you define (for example, if you exceed the free usage tier).

How to connect to your instances
Your instances are launching, and it may take a few minutes until they are in the **running** state, when they will be ready for you to use. Usage hours on your new instances will start immediately and continue to accrue until you stop or terminate your instances.

Feedback English (US) Privacy Policy Terms of Use Cookie preferences



CodeDeploy - AWS Developer Tools | Instances | EC2 Management Console | +

console.aws.amazon.com/codesuite/codedeploy/applications/CodeDeployDemo/deployment-groups/new?region=us-east-1

aws Services ec2

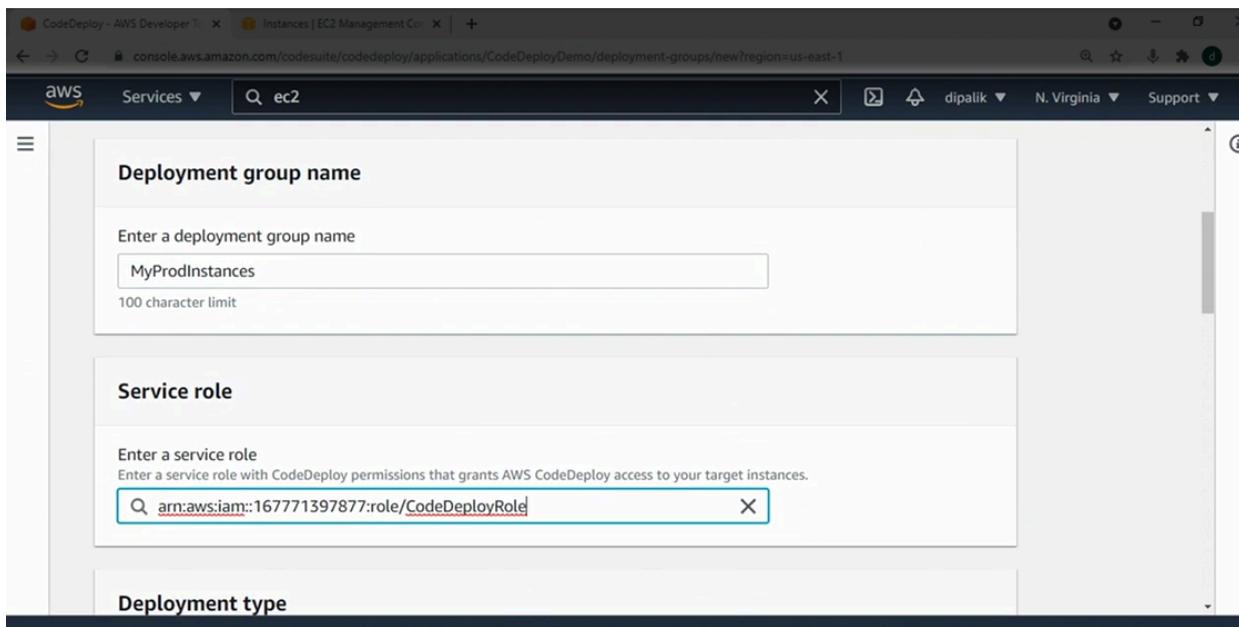
Deployment group name

Enter a deployment group name
MyProdInstances
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::167771397877:role/CodeDeployRole

Deployment type



arn:aws:iam::167771397877:role/CodeDeployRole

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

Specify the Amazon EC2 Auto Scaling groups or Amazon EC2 instances where the current application revision is deployed.

Feedback English (US) ▾ Privacy Policy Terms of Use Cookie preferences

CodeDeploy - AWS Developer Tools Instances | EC2 Management Console

console.aws.amazon.com/codesuite/codedeploy/applications/CodeDeployDemo/deployment-groups/new?region=us-east-1

AWS Services ▾ Q ec2

Deployment type

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

Specify the Amazon EC2 Auto Scaling groups or Amazon EC2 instances where the current application revision is deployed.

Automatically copy Amazon EC2 Auto Scaling group
Provision an Amazon EC2 Auto Scaling group and deploy the new application revision to it. AWS CodeDeploy will create the Auto Scaling group by copying the one you specify here.

Manually provision instances
I will specify here the instances where the current application revision is running. I will specify the instances for the replacement environment when I create a deployment.

Feedback English (US) ▾ Privacy Policy Terms of Use Cookie preferences

The screenshot shows the AWS CodeDeploy 'Deployment type' configuration screen. It features two main options: 'In-place' (selected) and 'Blue/green'. The 'In-place' option is described as updating instances in the deployment group with the latest application revisions, noting that each instance will be briefly taken offline for its update. The 'Blue/green' option is described as replacing instances with new ones and deploying the latest revision to them, mentioning that after instances in the replacement environment are registered with a load balancer, instances from the original environment are deregistered and can be terminated. Below this, there's an 'Environment configuration' section where users can select Amazon EC2 Auto Scaling groups to add to the deployment. A checkbox for 'Amazon EC2 Auto Scaling groups' is present but unchecked.

In the in-place deployment type, each instance will be taken offline for its update.

The screenshot shows the AWS EC2 Instances tag configuration screen. It displays a list of selected instances under 'Amazon EC2 instances' (4 unique matched instances). Below this, there's a note about adding up to three groups of tags for EC2 instances to a deployment group. It defines 'One tag group' as any instance identified by the tag group being deployed to, and 'Multiple tag groups' as instances identified by all the tag groups. A 'Tag group 1' section is shown with a key 'Environment' having a value 'Production'. There are buttons for 'Add tag' and 'Add tag group'. At the bottom, there's an unchecked checkbox for 'On-premises instances'.

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.AllAtOnce
CodeDeployDefault.OneAtATime
CodeDeployDefault.HalfAtATime
CodeDeployDefault.AllAtOnce

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

Application Load Balancer or Network Load Balancer Classic Load Balancer

CodeDeploy - AWS Developer Tools Instances | EC2 Management Console | +

aws Services Search for services, features, marketplace products, and docs [AWS] dipalik N. Virginia Support

Create deployment configuration

Deployment configuration name
Choose a deployment configuration name

100 character limit

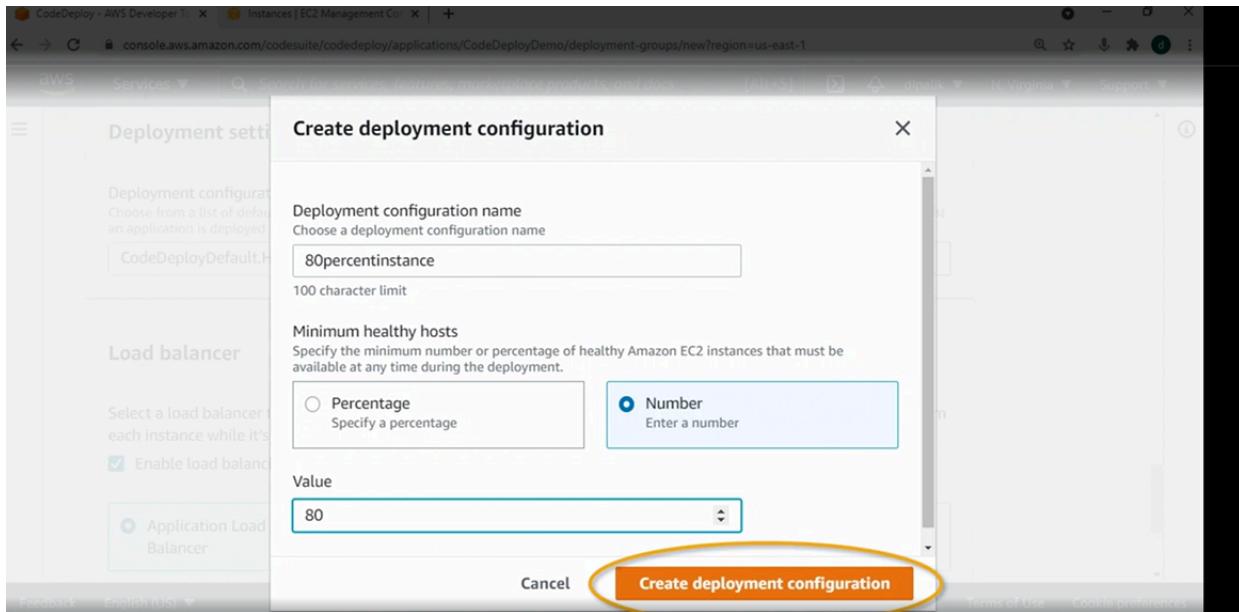
Minimum healthy hosts
Specify the minimum number or percentage of healthy Amazon EC2 instances that must be available at any time during the deployment.

Percentage Specify a percentage Number Enter a number

Value

Integer from 1 to 99

Cancel **Create deployment configuration**



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.HalfAtATime
or
Create deployment configuration

CodeDeployDefault.OneAtATime

CodeDeployDefault.HalfAtATime

CodeDeployDefault.AllAtOnce

80percentinstance

Select a load balancer to route 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

Application Load Balancer or Network Load Balancer

Classic Load Balancer

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

Create deployment configuration

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

Application Load Balancer or Network Load Balancer

Classic Load Balancer

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

Create deployment configuration

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

Cancel

Create deployment group

appspec.yml

The screenshot shows the AWS CodeCommit interface. On the left, there's a sidebar with 'Source' and 'CodeCommit' sections. Under 'CodeCommit', 'Code' is selected. The main area shows a repository named 'cicd-repo'. Inside, there's a 'scripts' folder containing several files: 'appspec.yml', 'buildspec.', 'CodeDeploy', and 'index.html'. A context menu is open over 'appspec.yml', with 'Open link in new tab' highlighted. At the bottom of the interface, there are links for 'Feedback', 'English (US)', 'Privacy Policy', 'Terms of Use', and 'Cookie preferences'.

The screenshot shows the contents of the 'appspec.yml' file in the 'scripts' folder of the 'cicd-repo' repository. The file contains the following YAML configuration:

```
1 version: 0.0
2 os: linux
3 files:
4   - source: /index.html
5     destination: /var/www/html/
6 hooks:
7   ApplicationStop:
8     - location: scripts/stop_server.sh
9       timeout: 300
10      runas: root
11
12 BeforeInstall:
13   - location: scripts/install_dependencies.sh
14     timeout: 300
15     runas: root
16
17 AfterInstall:
18   - location: scripts/after_install.sh
19     timeout: 300
20     runas: root
21
```

The screenshot shows the AWS CodeCommit console. On the left, the navigation pane is open with the following structure:

- Developer Tools
- CodeCommit
 - Source
 - Getting started
 - Repositories
 - Code**
 - Pull requests
 - Commits
 - Branches
 - Git tags
 - Settings
 - Approval rule templates
- Artifacts

The main content area displays a script configuration with line numbers:

```
9     timeout: 300
10    runas: root
11
12 BeforeInstall:
13   - location: scripts/install_dependencies.sh
14   timeout: 300
15   runas: root
16
17 AfterInstall:
18   - location: scripts/after_install.sh
19   timeout: 300
20   runas: root
21
22 ApplicationStart:
23   - location: scripts/start_server.sh
24   timeout: 300
25   runas: root
26
27 ValidateService:
28   - location: scripts/validate_service.sh
29   timeout: 300
30
31
```

The screenshot shows the AWS CodeDeploy console. On the left, the navigation pane is open with the following structure:

- Developer Tools
- CodeDeploy
 - Source
 - Artifacts
 - Build
 - Deploy
 - Getting started
 - Deployments
 - Deployment**
 - Applications
 - Deployment configurations
 - On-premises instances

The main content area is titled "Deployment lifecycle events" and displays a table of deployment events:

Instance ID	Duration	Status	Most recent event	Events	Start time
i-050df604abbe83c5c	8 seconds	✓ Succeeded	ValidateService	View events	Mar 7:5 (UT)
i-05f14727326ee2bb8	8 seconds	✓ Succeeded	ValidateService	View events	Mar 7:4 (UT)
i-083e1eb840d9ed40e	8 seconds	✓ Succeeded	ValidateService	View events	Mar 7:4 (UT)

Event	Duration	Status	Error code	Start time	End time
ApplicationStop	less than one second	Succeeded	-	May 2, 2021 7:50 PM (UTC+5:30)	May 2, 2021 7:50 PM (UTC+5:30)
DownloadBundle	less than one second	Succeeded	-	May 2, 2021 7:50 PM (UTC+5:30)	May 2, 2021 7:50 PM (UTC+5:30)
BeforeInstall	2 seconds	Succeeded	-	May 2, 2021 7:50 PM (UTC+5:30)	May 2, 2021 7:50 PM (UTC+5:30)
Install	less than one second	Succeeded	-	May 2, 2021 7:50 PM (UTC+5:30)	May 2, 2021 7:50 PM (UTC+5:30)

Rollbacks

Redeploy and roll back a deployment with CodeDeploy

CodeDeploy rolls back deployments by redeploying a previously deployed revision of an application as a new deployment. These rolled-back deployments are technically new deployments, with new deployment IDs, rather than restored versions of a previous deployment.

Deployments can be rolled back automatically or manually.

Topics

- Automatic rollbacks
- Manual rollbacks
- Rollback and redeployment workflow
- Rollback behavior with existing content

The screenshot shows a browser window with three tabs at the top: 'CodeDeploy - AWS Developer Tools', 'CloudWatch Management Console', and 'Redeploy and roll back a deployment'. The main content area displays the AWS Documentation for AWS CodeDeploy, specifically the 'User Guide' section. The left sidebar contains a navigation menu with items like 'Create a deployment', 'Redeploy and roll back a deployment', and 'Manual rollbacks'. The main content area has a heading 'Manual rollbacks' and a note about manually rolling back a deployment. A 'Note' box contains a warning about removing instances from a deployment group. On the right side, there's a 'On this page' sidebar with links to 'Automatic rollbacks', 'Manual rollbacks', 'Rollback and redeployment workflow', and 'Rollback behavior with existing content'.

This screenshot shows the same AWS CodeDeploy User Guide page, but the main content is now about 'Automatic rollbacks'. It lists three bullet points: 'Manual rollbacks', 'Rollback and redeployment workflow', and 'Rollback behavior with existing content'. Below this, there's a section titled 'Automatic rollbacks' with a note about configuring deployment groups to automatically roll back. A 'Note' box states that when creating a new deployment, you can choose to override the automatic rollback configuration. The right sidebar remains the same as in the previous screenshot.

The screenshot shows the AWS CodeDeploy console interface. The top navigation bar includes 'Services', a search bar, and user information. The main content area is titled 'Edit deployment group MyProdInstances'. On the left, there's a sidebar with checkboxes for 'Ignore alarm configuration' (checked) and 'Continue deployment even if alarm status is unavailable'. The main panel is titled 'Rollbacks' and contains a section for enabling deployment rollbacks. Under 'Rollbacks', there are three checkboxes: 'Roll back when a deployment fails' (checked), 'Roll back when alarm thresholds are met' (unchecked), and 'Disable rollbacks' (unchecked). The right side of the screen shows other deployment group details and a 'Leave' button.

Alarms

[Delete alarm](#)

[Add alarm](#)

Name

EC2Utilization

Ignore alarm configuration

Skips the step of checking Amazon CloudWatch alarms during the deployment process

Continue deployment even if alarm status is unavailable

Permits deployment to run when alarm data cannot be retrieved from Amazon Cloudwatch

Rollbacks