

# AWS CodeDeploy FAQs | Amazon Web Services

[aws.amazon.com](https://aws.amazon.com/codedeploy/faqs/) (<https://aws.amazon.com/codedeploy/faqs/>)

## General

Q: What is AWS CodeDeploy?

AWS CodeDeploy is a service that automates code deployments to any instance, including Amazon EC2 instances and instances running on-premises. AWS CodeDeploy makes it easier for you to rapidly release new features, helps you avoid downtime during deployment, and handles the complexity of updating your applications. You can use AWS CodeDeploy to automate deployments, eliminating the need for error-prone manual operations, and the service scales with your infrastructure so you can easily deploy to one instance or thousands.

Q: Who should use AWS CodeDeploy?

AWS CodeDeploy is designed for developers and administrators who need to deploy applications to any instance, including Amazon EC2 instances and instances running on-premises. It is flexible and can also be used by anyone wanting to update software or run scripts on their instances.

Q: What types of applications can be deployed with AWS CodeDeploy?

AWS CodeDeploy can be used for deploying any type of application. To use AWS CodeDeploy, you specify the files to copy and the scripts to run on each instance during the deployment. AWS CodeDeploy is programming language and architecture agnostic, so you can use scripts for any custom deployment logic.

Q: What operating systems does AWS CodeDeploy support?

AWS CodeDeploy supports a wide variety of operating systems. AWS CodeDeploy provides agents that have been tested on Amazon Linux, Red Hat Enterprise Linux, Ubuntu Server, and Microsoft Windows Server. If you want to use other operating systems, the AWS CodeDeploy agent is available as open source software here (<https://github.com/aws/aws-codedeploy-agent>). For more information on operating system support, see AWS CodeDeploy Documentation (<http://docs.aws.amazon.com/codedeploy/latest/userguide/host-cleanup.html#how-to-run-agent-supported-os>).

Q: Will AWS CodeDeploy work with my existing tool chain?

Yes. AWS CodeDeploy works with a variety of configuration management systems, continuous integration and deployment systems, and source control systems. For more information, see product integrations page.

Q: How is AWS CodeDeploy different from other AWS deployment and management services such as AWS Elastic Beanstalk and AWS OpsWorks?

AWS CodeDeploy is a building block service focused on helping developers deploy and update software on any instance, including Amazon EC2 instances and instances running on-premises. AWS Elastic Beanstalk and AWS OpsWorks are end-to-end application management solutions.

Q: Does AWS CodeDeploy support on-premises instances?

Yes. AWS CodeDeploy supports any instance that can install the CodeDeploy agent and connect to AWS public endpoints ([http://docs.aws.amazon.com/general/latest/gr/rande.html#codedeploy\\_region](http://docs.aws.amazon.com/general/latest/gr/rande.html#codedeploy_region)).

## Concepts

Q: What is an application?

An application is a collection of software and configuration to be deployed to a group of instances. Typically, the instances in the group run the same software. For example, if you have a large distributed system, the web tier will likely constitute one application and the data tier another application.

Q: What is a revision?

A revision is a specific version of deployable content, such as source code, post-build artifacts, web pages, executable files, and deployment scripts, along with an AppSpec (<http://docs.aws.amazon.com/codedeploy/latest/userguide/writing-app-spec.html>) file. The AWS CodeDeploy Agent can access a revision from GitHub or an Amazon S3 bucket.

Q: What is a deployment group?

A deployment group is the AWS CodeDeploy entity for grouping EC2 instances or AWS Lambda functions in a CodeDeploy deployment. For EC2 deployments, it is a set of instances associated with an application that you target for a deployment. You can add instances to a deployment group by specifying a tag, an Auto Scaling group name, or both. In an AWS Lambda deployment, a deployment group defines a set of AWS CodeDeploy configurations for future serverless Lambda deployment to the group, like alarms and rollbacks.

You can define multiple deployment groups for an application such as staging and production. For information on tags, see *Working with Amazon EC2 Tags in the Console* ([http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/Using\\_Tags.html#Using\\_Tags\\_Console](http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/Using_Tags.html#Using_Tags_Console)). For more information on deploying to Auto Scaling groups, see *Auto Scaling Integration* (<http://docs.aws.amazon.com/codedeploy/latest/userguide/auto-scaling-integ.html>).

Q: What is a deployment configuration?

A deployment configuration specifies how the behavior for how deployment should proceed, including how to handle deployment failure, through for a deployment group. You can use a deployment configuration to perform zero-downtime deployments to multi-instance deployment groups. For example, if your application needs at least 50% of the instances in a deployment group to be up and serving traffic, you can specify that in your deployment configuration so that a deployment does not cause downtime. If no deployment configuration is associated with either the deployment or the deployment group, then by default AWS CodeDeploy will deploy to one instance at a time. For more information on deployment configuration, see *Instance Health* (<http://docs.aws.amazon.com/codedeploy/latest/userguide/host-health.html>).

Q: What are the parameters that I need to specify for a deployment?

There are three parameters you specify for a deployment:

1. Revision - Specifies what to deploy.
2. Deployment group - Specifies where to deploy.
3. Deployment configuration - An optional parameter that specifies how to deploy.

Q: What is an AppSpec file?

An AppSpec file is a configuration file that specifies the files to be copied and scripts to be executed. The AppSpec file uses the YAML (<http://www.yaml.org/>) format, and you include it in the root directory of your revision. The AppSpec file is used by the AWS CodeDeploy Agent and consists of two sections. The files section specifies the

source files in your revision to be copied and the destination folder on each instance. The hooks section specifies the location (as relative paths starting from the root of the revision bundle) of the scripts to run during each phase of the deployment. Each phase of a deployment is called a deployment lifecycle event. The following is a sample AppSpec file. For more information on an AppSpec file, including all the options that can be specified, see AppSpec File Reference (<http://docs.aws.amazon.com/codedeploy/latest/userguide/app-spec-ref.html>).

```
version: 0.0
os: linux
files:
# You can specify one or more mappings in the files section.
- source: /
  destination: /var/www/html/WordPress
hooks:
# The lifecycle hooks sections allows you to specify deployment scripts.
ApplicationStop:
# Step 1: Stop Apache and MySQL if running.
- location: helper_scripts/stop_server.sh
BeforeInstall:
# Step 2: Install Apache and MySQL.
# You can specify one or more scripts per deployment lifecycle event.
- location: deploy_hooks/puppet-apply-apache.sh
- location: deploy_hooks/puppet-apply-mysql.sh
AfterInstall:
# Step 3: Set permissions.
- location: deploy_hooks /change_permissions.sh
  timeout: 30
  runas: root
# Step 4: Start the server.
- location: helper_scripts/start_server.sh
  timeout: 30
  runas: root
```

Q: What are deployment lifecycle events?

A deployment goes through a set of predefined phases called deployment lifecycle events. A deployment lifecycle event gives you an opportunity to run code as part of the deployment. The following table lists the different deployment lifecycle events currently supported, in their order of execution, along with examples of when you may want to use them.

#### Deployment Lifecycle Event Description ApplicationStop

This is the first deployment lifecycle event that occurs even before the revision gets downloaded. The AppSpec file and scripts used for this deployment lifecycle event are from the last successfully deployed revision.

You can use the ApplicationStop deployment lifecycle event if you want to gracefully stop the application or remove currently installed packages in preparation of a deployment.

**DownloadBundle** During this deployment lifecycle event, the agent copies the revision files to a temporary location on the instance. This deployment lifecycle event is reserved for the agent and cannot be used to run user scripts.

**BeforeInstall** You can use the BeforeInstall deployment lifecycle event for preinstall tasks such as decrypting files and creating a backup of the current version.

**Install** During this deployment lifecycle event, the agent copies the revision files from the temporary location to the final destination folder. This deployment lifecycle event is reserved for the agent and cannot be used to run user scripts.

**AfterInstall** You can use the AfterInstall deployment lifecycle event for tasks such as configuring your application or changing file permissions. **ApplicationStart** You typically use the ApplicationStart deployment lifecycle event to restart services that were stopped during ApplicationStop. **ValidateService** ValidateService is the last deployment lifecycle event and is an opportunity to verify that the deployment completed successfully.

## Getting started

Q: How do I get started with AWS CodeDeploy?

You can sign in to the AWS Management Console (<https://console.aws.amazon.com/codedeploy>) and start using AWS CodeDeploy. If you are looking for a quick overview of the service, see **Getting Started** (<http://docs.aws.amazon.com/codedeploy/latest/userguide/getting-started.html>), which includes a step-by-step tutorial.

## Using AWS CodeDeploy

Q: Are there any prerequisites for using an existing Amazon EC2 instance with AWS CodeDeploy?

The Amazon EC2 instance must be associated with an IAM instance profile (<http://docs.aws.amazon.com/IAM/latest/UserGuide/instance-profiles.html>) and should be running a supported operating system. For more information, see **Use an Existing Amazon EC2 Instance** (<http://docs.aws.amazon.com/codedeploy/latest/userguide/how-to-configure-existing-instance.html>).

Q: What are the typical steps to go through for deploying an application using AWS CodeDeploy?

The following diagram shows the typical steps during a deployment. Creating an application and deployment group (see the **Concepts** section for an explanation of these terms) are typically one-time setup tasks per application. The recurring actions are uploading a revision and deploying it. For a detailed explanation, including step-by-step

instructions for each of these tasks, see Deployments (<http://docs.aws.amazon.com/codedeploy/latest/userguide/deployment-steps.html>).

Q: How can I access AWS CodeDeploy?

You can access AWS CodeDeploy using the AWS Management Console (<https://console.aws.amazon.com/codedeploy>), the AWS Command Line Interface (AWS CLI) (<http://docs.aws.amazon.com/cli/latest/reference/codedeploy>), the AWS SDKs (<https://aws.amazon.com/tools/>), and the AWS CodeDeploy APIs (<http://docs.aws.amazon.com/codedeploy/latest/APIReference>).

Q: What changes do I need to make to my code to deploy using AWS CodeDeploy?

You don't need to make any changes to your code. You simply add a configuration file (called an AppSpec (<http://docs.aws.amazon.com/codedeploy/latest/userguide/writing-app-spec.html>) file) in the root directory of your revision bundle that specifies the files to be copied and scripts to be executed.

Q: How can I deploy an application from my source control system using AWS CodeDeploy?

If you are using GitHub, you can deploy a revision in a .zip, .tar, or .tar.gz format from your repository directly to instances. For other source control systems, you can bundle and upload the revision to an Amazon S3 bucket in a .zip, .tar, or .tar.gz format and specify the Amazon S3 location when doing a deployment. If your application needs a build step, make sure that the GitHub repository or the Amazon S3 bucket contains the post-build artifacts. For more information on using GitHub with AWS CodeDeploy, see our product integrations page. For more information on using Amazon S3 for storing revisions, see Push a Revision (<http://docs.aws.amazon.com/codedeploy/latest/userguide/how-to-push-revision.html>).

Q: How will AWS CodeDeploy work with my configuration management tool?

You can invoke your configuration management tool from any deployment lifecycle event hook in the AppSpec file. For example, if you have a Chef recipe that you want to run as part of a deployment, you can do so by specifying it in the appropriate deployment lifecycle event hook in the AppSpec file. In addition, you can leverage your configuration management system to install the AWS CodeDeploy agent on instances. For samples that illustrate using AWS CodeDeploy with configuration management systems such as Chef, Puppet, Ansible, and Saltstack, see our product integrations page.

Q: Can I use AWS CodeDeploy with continuous integration and deployment systems?

Yes. You can integrate AWS CodeDeploy with your continuous integration and deployment systems by calling the public APIs using the AWS CLI or AWS SDKs. You can find prebuilt integrations and samples on our product integrations page.

Q: How do I get my application on the instances that I just added to the deployment group?

Deploy the latest revision to the deployment group for the newly added instances to get your application. Except for Amazon EC2 instances that are launched as part of an Auto Scaling group, AWS CodeDeploy doesn't automatically deploy the latest revision to newly added instances.

Q: How does AWS CodeDeploy work with Auto Scaling?

You can associate an Auto Scaling group with a deployment group to make sure that newly launched instances always get the latest version of your application. Every time a new Amazon EC2 instance is launched for that Auto Scaling group, it will be first put in a Pending state and a deployment of the last successful revision for that deployment group triggered on that Amazon EC2 instance. If the deployment completes successfully, the state of the Amazon EC2 instance is changed to InService. If that deployment fails, the Amazon EC2 instance is terminated, a new Amazon EC2 instance is launched in Pending state, and a deployment triggered for the newly launched EC2

instance. For more information on Auto Scaling group instance lifecycle events, see Auto Scaling Group Lifecycle (<http://docs.aws.amazon.com/AutoScaling/latest/DeveloperGuide/AutoScalingGroupLifecycle.html>).

Q: How do I track the status of a deployment?

You can track the status of a deployment using the AWS Management Console (<https://console.aws.amazon.com/codedeploy>), the AWS Command Line Interface (AWS CLI) (<http://docs.aws.amazon.com/cli/latest/reference/codedeploy>), the AWS SDKs (<https://aws.amazon.com/tools/>), and the AWS CodeDeploy APIs (<http://docs.aws.amazon.com/codedeploy/latest/APIReference>). You can see the overall status of a deployment and drill down further to see the status of each instance and the status of each deployment lifecycle event for the instance. You can also see the log entries corresponding to any failure, making it easy to debug deployment issues without having to log into the instance.

Q: Can I stop an in-flight deployment?

Yes. When you stop an in-flight deployment, the AWS CodeDeploy service will instruct the agent on each instance to stop executing additional scripts. To get your application back to a consistent state, you can either redeploy the revision, or deploy another revision.

Q: How do I roll back an application to the previous revision?

To roll back an application to a previous revision, you just need to deploy that revision. AWS CodeDeploy keeps track of the files that were copied for the current revision and removes them before starting a new deployment, so there is no difference between redeploy and roll back. However, you need to make sure that the previous revisions are available for roll back.

Q: Can I use a versioned Amazon S3 bucket to store revisions?

Yes. You can use a versioned Amazon S3 bucket and specify the version ID to uniquely identify a revision.

Q: What are the service limits when using AWS CodeDeploy?

For information on the service limits, see Limits (<http://docs.aws.amazon.com/codedeploy/latest/userguide/limits.html>). To increase your service limits, submit a request through the AWS Support Center. (<https://console.aws.amazon.com/support/home?region=us-east-1#/case/create>)

Q: Can I get a history of AWS CodeDeploy API calls made on my account for security analysis and operational troubleshooting purposes?

Yes. To receive a history of AWS CodeDeploy API calls made on your account, you simply turn on AWS CloudTrail in the AWS Management Console.

## Security

Q: Can I use AWS CodeDeploy to deploy an application to Amazon EC2 instances running within an Amazon Virtual Private Cloud (VPC)?

Yes, but the AWS CodeDeploy agent installed on the Amazon EC2 instances must be able to access the public AWS CodeDeploy and Amazon S3 service endpoints. For more information, see AWS CodeDeploy Endpoints ([http://docs.aws.amazon.com/general/latest/gr/rande.html#codedeploy\\_region](http://docs.aws.amazon.com/general/latest/gr/rande.html#codedeploy_region)) and Amazon S3 Endpoints ([http://docs.aws.amazon.com/general/latest/gr/rande.html#s3\\_region](http://docs.aws.amazon.com/general/latest/gr/rande.html#s3_region)).

Q: Can I use AWS Identity and Access Management (IAM) to manage access to AWS CodeDeploy?

Yes. AWS CodeDeploy supports resource-level permissions (<http://docs.aws.amazon.com/IAM/latest/UserGuide/PermissionsOverview.html>). For each AWS CodeDeploy resource, you can specify which user has access and to

which actions. For example, you can set an IAM policy to let a user deploy a particular application but only list revisions for other applications. You can therefore prevent users from inadvertently making changes to the wrong application. For more information on using IAM with AWS CodeDeploy, see [Access Permissions Reference](http://docs.aws.amazon.com/codedeploy/latest/userguide/access-permissions.html) (<http://docs.aws.amazon.com/codedeploy/latest/userguide/access-permissions.html>).

## Regions

Q: Which regions does AWS CodeDeploy support?

Please refer to [Regional Products and Services](https://aws.amazon.com/about-aws/global-infrastructure/regional-product-services/) (<https://aws.amazon.com/about-aws/global-infrastructure/regional-product-services/>) for details of CodeDeploy availability by region.

Q: How do I deploy an AWS CodeDeploy application to multiple regions?

AWS CodeDeploy performs deployments with AWS resources located in the same region. To deploy an application to multiple regions, define the application in your target regions, copy the application bundle to an Amazon S3 bucket in each region, and then start the deployments using either a serial or parallel rollout across the regions.

## Billing

Q: How much does AWS CodeDeploy cost?

There is no additional charge for code deployments to Amazon EC2 instances through AWS CodeDeploy. You pay \$0.02 per on-premises instance update using AWS CodeDeploy. Please see the [Pricing](#) page for more details.

Explore AWS CodeDeploy integrations

Visit the [product integrations](#) page

Ready to build?

Get started with AWS CodeDeploy (<https://console.aws.amazon.com/console/home>)

Have more questions?

Contact us

Register for re:Invent Bootcamps

Learn from AWS experts at exam prep bootcamps - space is limited!



([https://reinvent.awsevents.com/learn/bootcamps/?sc\\_icampaign=aware\\_reinvent\\_bootcamps2019\\_aws&sc\\_ichannel=ha&sc\\_icontent=awssm-2448&sc\\_iplace=2up&trk=ha\\_awssm-2448](https://reinvent.awsevents.com/learn/bootcamps/?sc_icampaign=aware_reinvent_bootcamps2019_aws&sc_ichannel=ha&sc_icontent=awssm-2448&sc_iplace=2up&trk=ha_awssm-2448))

Get AWS Certified

AWS Certifications can help you advance your career and boost your earning power



([https://pages.awscloud.com/tc\\_get-aws-certified.html?sc\\_icampaign=aware\\_getcertified\\_evergreen2019&sc\\_ichannel=ha&sc\\_icontent=awssm-2655&sc\\_iplace=2up&trk=ha\\_awssm-2655](https://pages.awscloud.com/tc_get-aws-certified.html?sc_icampaign=aware_getcertified_evergreen2019&sc_ichannel=ha&sc_icontent=awssm-2655&sc_iplace=2up&trk=ha_awssm-2655))

AWS re:Invent | December 2 – 6, 2019 | Las Vegas, Nevada

Reserved seating opens October 15th. Register now to save your spot. [View session catalog >>](#)

# AWS re:Invent

([https://reinvent.awsevents.com/learn/?sc\\_icampaign=Event\\_reInvent\\_2019\\_1up\\_DG5\\_VIP&sc\\_ichannel=ha&sc\\_icontent=awssm-3019-a&sc\\_ioutcome=Strategic\\_Events&sc\\_iplace=1up&trk=ha\\_a131L0000058G7nQAE~ha\\_awssm-3019~ha\\_awssm-3019-a&trkCampaign=AWS\\_reInvent\\_2019](https://reinvent.awsevents.com/learn/?sc_icampaign=Event_reInvent_2019_1up_DG5_VIP&sc_ichannel=ha&sc_icontent=awssm-3019-a&sc_ioutcome=Strategic_Events&sc_iplace=1up&trk=ha_a131L0000058G7nQAE~ha_awssm-3019~ha_awssm-3019-a&trkCampaign=AWS_reInvent_2019))

Page Content

General Concepts Getting started Using AWS CodeDeploy Security Regions Billing

[aws.amazon.com](https://aws.amazon.com) (<https://aws.amazon.com/codedeploy/faqs/>)