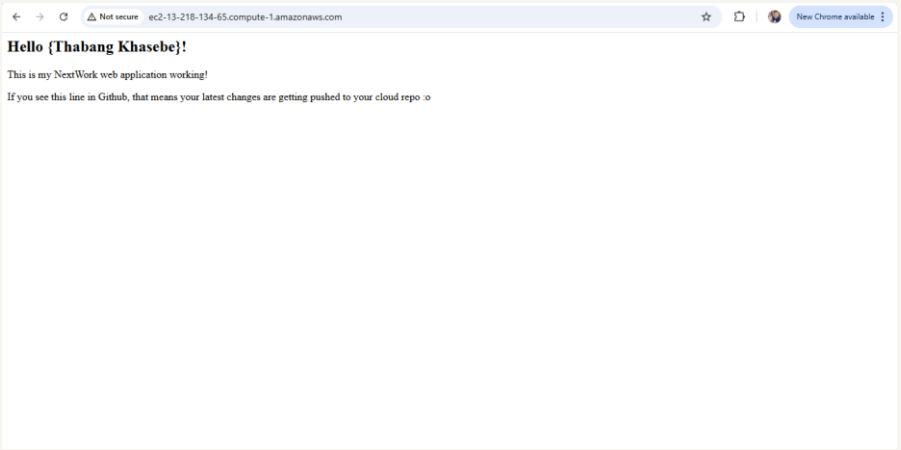
Deploy a Web App with CodeDeploy

Thabang Khasebe



Thabang Khasebe

Introducing Today's Project!

In this project, I will demonstrate how to automate the deployment of a web app using AWS CodeDeploy. I'm doing this project to learn how to streamline deployments, reduce errors, and maintain application availability during updates.

Key tools and concepts

Services used: CodeDeploy, EC2, IAM, S3. Key concepts: automated deployments, deployment groups, IAM roles, and managing rollouts to EC2 instances.

Project reflection

This project took me approximately 3 hours. The most challenging part was configuring the deployment scripts and appspec.yml correctly. It was most rewarding to see the app deploy automatically without manual steps.

This project is part five of a series of DevOps projects where I'm building a CI/CD pipeline! I'll be working on the next project tomorrow

Thabang Khasebe

Deployment Environment

To set up for CodeDeploy, I launched an EC2 instance and VPC because the EC2 instance hosts the web app to be deployed, and the VPC provides a secure, isolated network environment for the instance to operate safely.

Instead of launching these resources manually, I used AWS CloudFormation. When I need to delete these resources, I can simply delete the CloudFormation stack, which automatically cleans up all associated resources.

Other resources created include a security group, IAM role, and key pair. They're in the template to enable secure access, manage permissions, and allow SSH connections to the EC2 instance, ensuring safe and proper setup.

Thabang Khasebe

Resources (11)

Search resources

1

Logical ID	Physical ID	Type	Status	Module
DeployRoleProfile	NextWorkCodeDeployEC2Stack-DeployRoleProfile-3PbVGyIq94i9	AWS::IAM::InstanceProfile	DELETE_COMPLETE	-
InternetGateway	igw-086d4118ae9754db6	AWS::EC2::InternetGateway	DELETE_COMPLETE	-
PublicInternetRoute	rtb-05dbccb783cb331f5|0.0.0.0/0	AWS::EC2::Route	DELETE_COMPLETE	-
PublicRouteTable	rtb-05dbccb783cb331f5	AWS::EC2::RouteTable	DELETE_COMPLETE	-
PublicSecurityGroup	sg-038c9a981ad14284e	AWS::EC2::SecurityGroup	DELETE_COMPLETE	-
PublicSubnetA	subnet-04d5a69d502aaf92e	AWS::EC2::Subnet	DELETE_COMPLETE	-

Thabang Khasebe

Deployment Scripts

Scripts are small programs or sets of commands used to automate tasks. To set up CodeDeploy, I also wrote scripts to install dependencies, start the server, and stop the server during the deployment process.

install_dependencies will install and configure the necessary software, like Tomcat and Apache, to run the web application and ensure they work together to make the site accessible online.

start_server.sh will start the Tomcat server so that the deployed web application can run and be accessed by users through the internet.

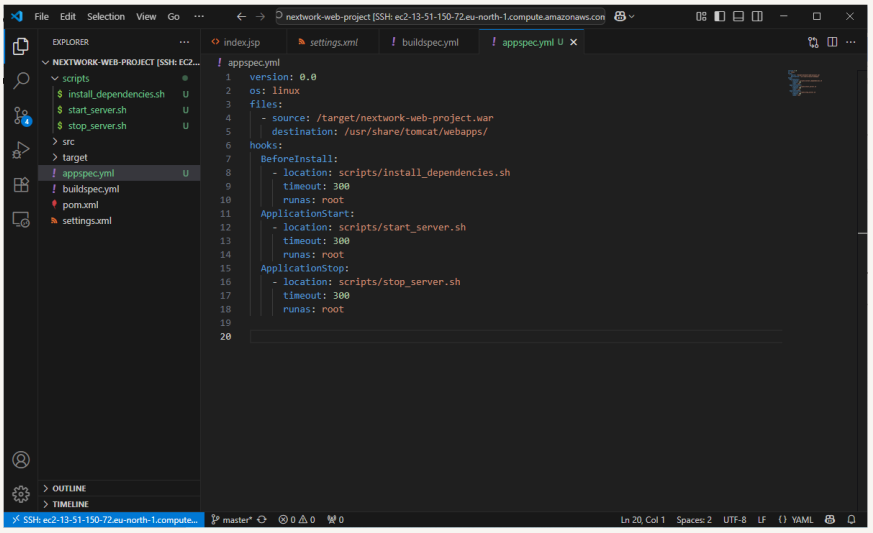
stop_server.sh will stop the Tomcat server to safely shut down the web application before updating or redeploying it, preventing issues during deployment.

Thabang Khasebe

appspec.yml

Then, I wrote an appspec.yml file to tell CodeDeploy how to deploy my app. The `files` section maps the WAR file to the Tomcat directory, and the `hooks` section lists scripts to run before install, to start, and to stop the server.

I also updated buildspec.yml because I needed to include the appspec.yml file and deployment scripts in the build artifacts. This ensures CodeDeploy has all the instructions and files it needs to deploy the application after the build completes.



Thabang Khasebe

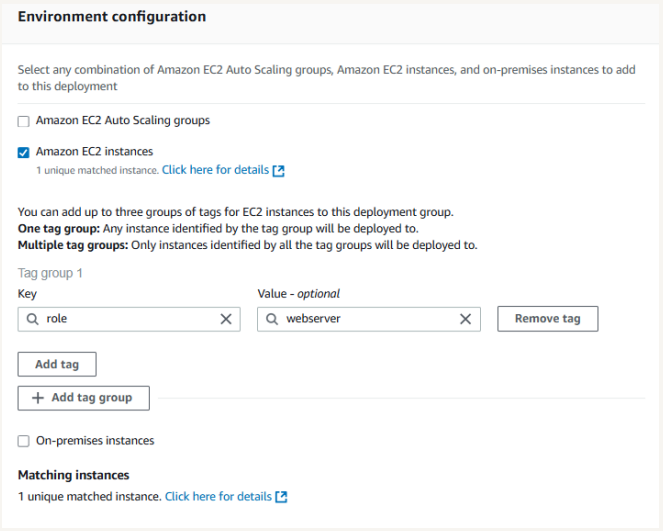
Setting Up CodeDeploy

A deployment group is a set of deployment settings and target instances where your app will be deployed. A CodeDeploy application is the logical container that represents the app you want to deploy, organizing its deployments and versions.

To set up a deployment group, you also need to create an IAM role to grant CodeDeploy the permissions it needs to access your EC2 instances and manage the deployment process securely and efficiently.

Tags are helpful for organizing and targeting resources. I used the tag `webserver` to identify which EC2 instances CodeDeploy should deploy the application to, ensuring deployments only affect the right servers.

Thabang Khasebe

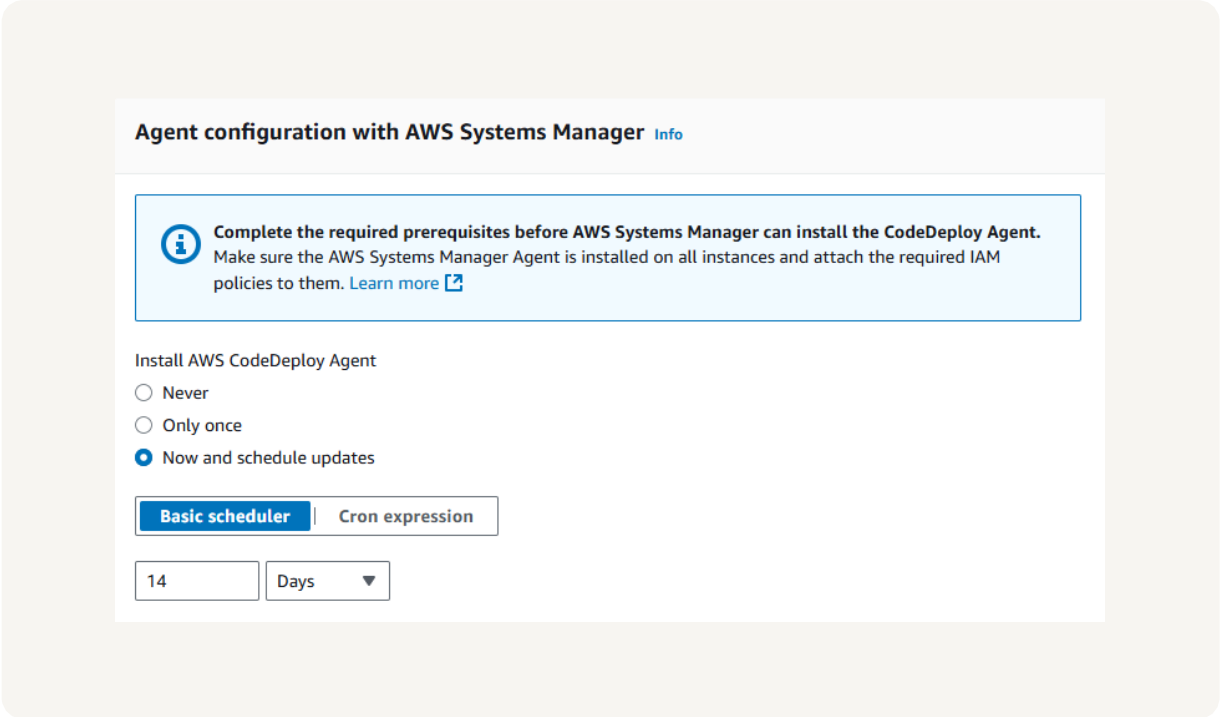


Thabang Khasebe

Deployment configurations

Another key setting is the deployment configuration, which affects how your application is deployed across instances. I used CodeDeployDefault.AllAtOnce, so all instances update simultaneously,minimizing deployment time but with some risk of downtime

In order to connect the EC2 instance with CodeDeploy, a CodeDeploy Agent is also set up to communicate deployment instructions, download the application revision, and run deployment scripts on the instance.



Thabang Khasebe

Success!

A CodeDeploy deployment is the process of releasing your app to target instances. A deployment group defines which instances get the app and how; the deployment is the actual rollout using that group's settings.

I had to configure a revision location, which means specifying where CodeDeploy finds the app files to deploy. My revision location was the S3 bucket holding the build artifact and deployment scripts packaged during the build process.

To check that the deployment was a success, I visited the public DNS of my EC2 instance in a browser. I saw my web app running, confirming CodeDeploy successfully deployed the latest version.

Thabang Khasebe