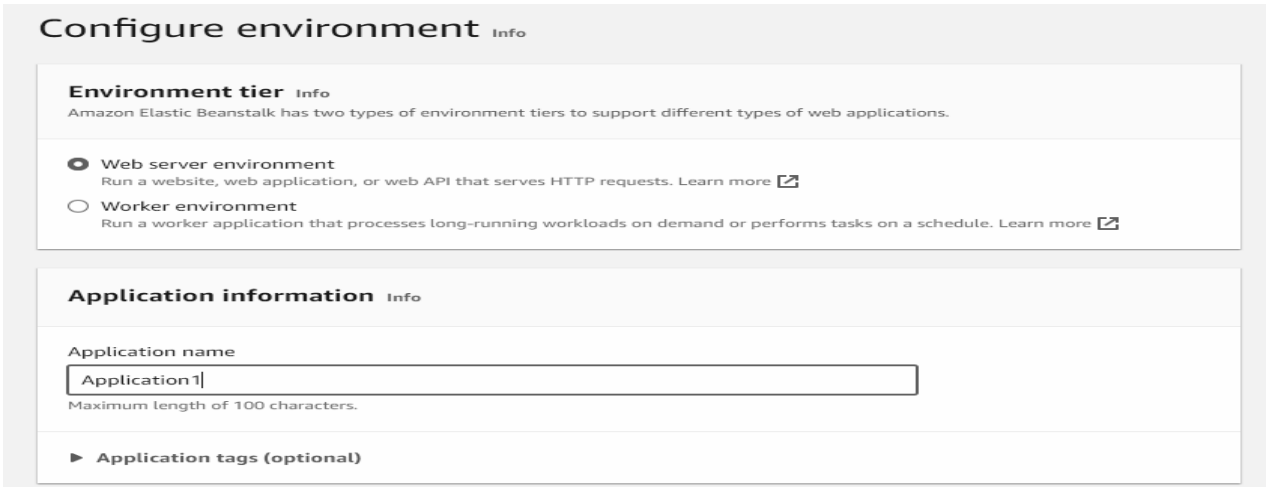
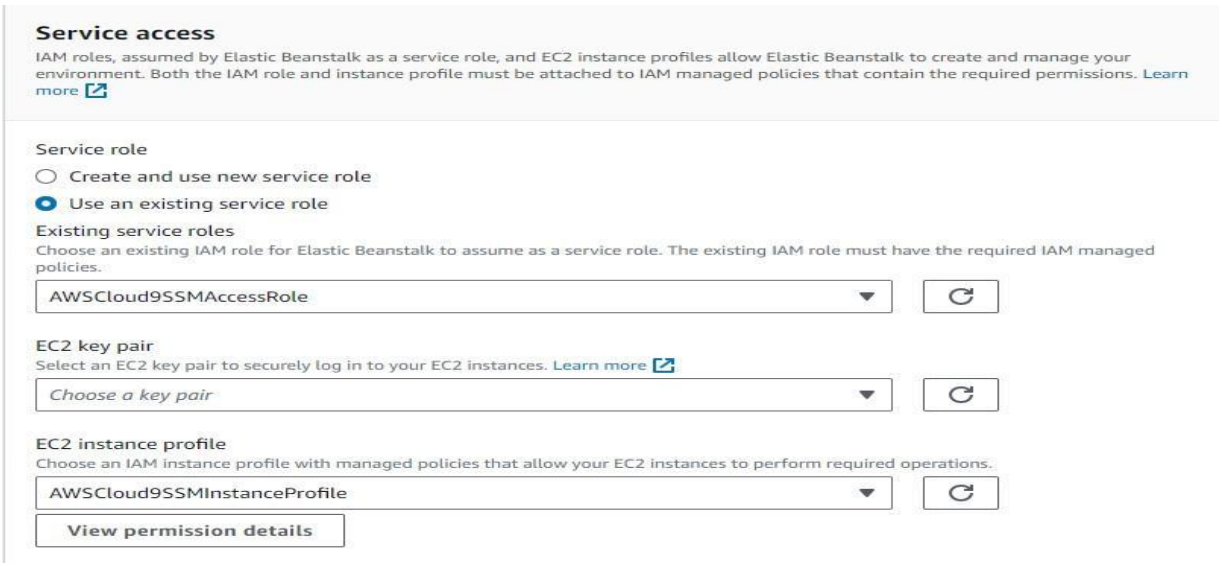
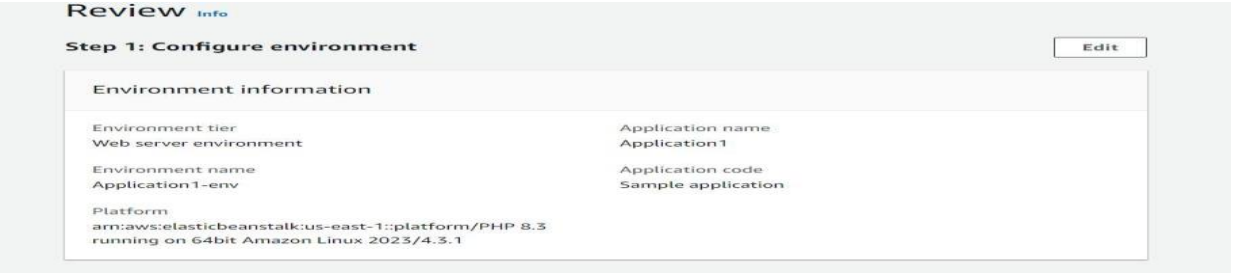
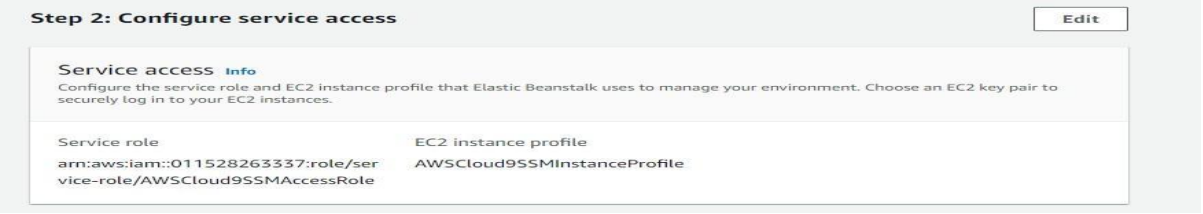

Configure environment Info
Environment tier Info
Amazon Elastic Beanstalk has two types of environment tiers to support different types of web applications.
Web server environment
Run a website, web application, or web API that serves HTTP requests. Learn more
Worker environment
Run a worker application that processes long-running workloads on demand or performs tasks on a schedule. Learn more
Application information Info
Application name
Application1
Maximum length of 100 characters.
Application tags (optional)


Service access
IAM roles, assumed by Elastic Beanstalk as a service role, and EC2 instance profiles allow Elastic Beanstalk to create and manage your environment. Both the IAM role and instance profile must be attached to IAM managed policies that contain the required permissions. Learn more
Service role
Create and use new service role
Use an existing service role
Existing service roles
Choose an existing IAM role for Elastic Beanstalk to assume as a service role. The existing IAM role must have the required IAM managed policies.
AWSCloud9SSMAccessRole
EC2 key pair
Select an EC2 key pair to securely log in to your EC2 instances. Learn more
Choose a key pair
EC2 instance profile
Choose an IAM instance profile with managed policies that allow your EC2 instances to perform required operations.
AWSCloud9SSMInstanceProfile
View permission details


Set up networking, database, and tags – optional Info
Virtual Private Cloud (VPC)
VPC
Launch your environment in a custom VPC instead of the default VPC. You can create a VPC and subnets in the VPC management console.
Learn more
vpc-0da5a3d9b481e2d7b | (172.31.0.0/16)
Create custom VPC


Review Info
Step 1: Configure environment
Edit
Environment information
Environment tier
Web server environment
Application name
Application1
Environment name
Application1-env
Application code
Sample application
Platform
arn:aws:elasticbeanstalk:us-east-1::platform/PHP 8.3 running on 64bit Amazon Linux 2023/4.3.1


Step 2: Configure service access
Edit
Service access Info
Configure the service role and EC2 instance profile that Elastic Beanstalk uses to manage your environment. Choose an EC2 key pair to securely log in to your EC2 instances.
Service role
arn:aws:iam::011528263337:role/service-role/AWSCloud9SSMAccessRole
EC2 instance profile
AWSCloud9SSMInstanceProfile

Ignore health check
false

Instance replacement
false

Platform software

Lifecycle
false

Log streaming
Deactivated

Allow URL fopen
On

Display errors
Off

Document root
–

Max execution time
60

Memory limit
256M

Zlib output compression
Off

Proxy server
nginx

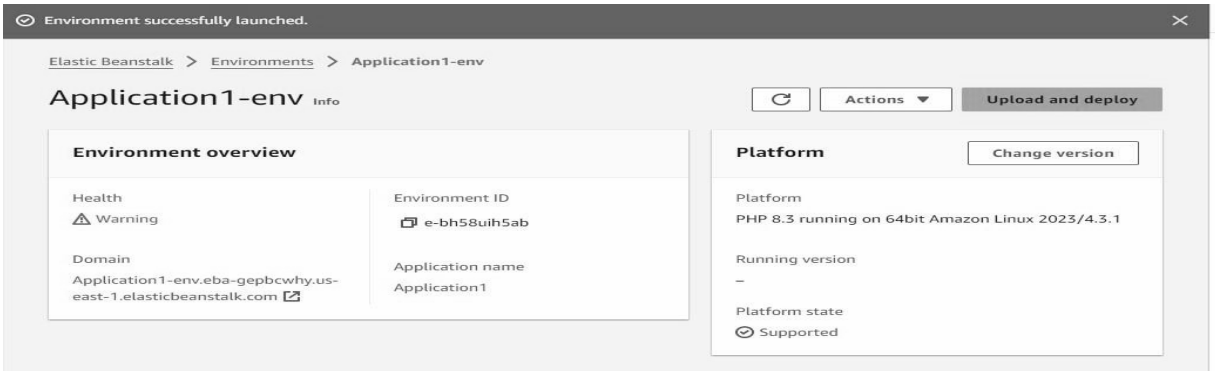
Logs retention
7

Rotate logs
Deactivated

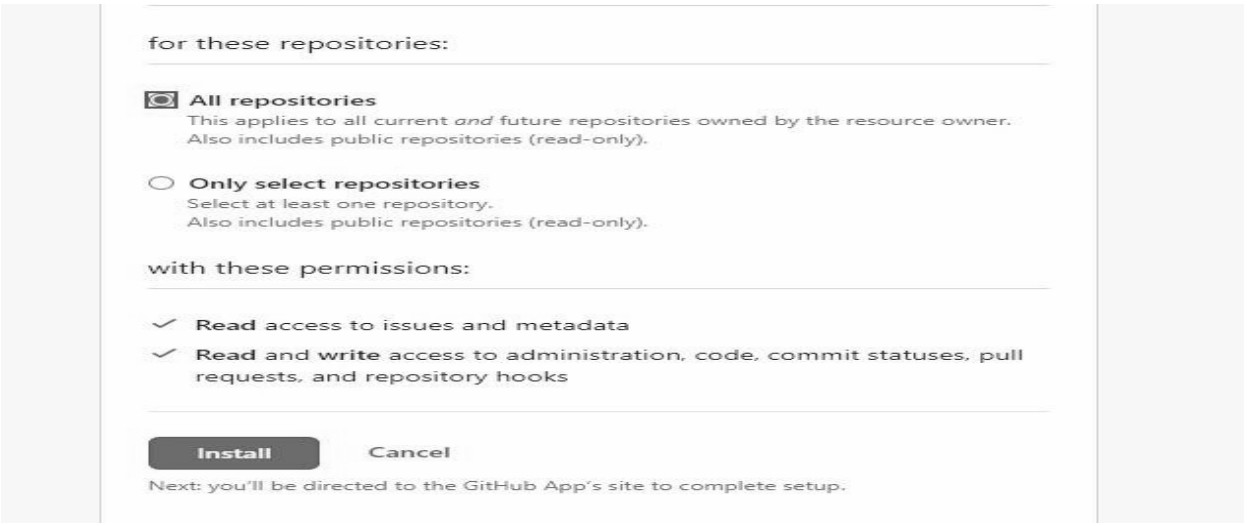
Update level
minor

X-Ray enabled
Deactivated

Environment properties




Choose pipeline settings Info
Step 1 of 5
Pipeline settings
Pipeline name
Enter the pipeline name. You cannot edit the pipeline name after it is created.
pipeline1
No more than 100 characters
Pipeline type
You can no longer create V1 pipelines through the console. We recommend you use the V2 pipeline type with improved release safety, pipeline triggers, parameterized pipelines, and a new billing model.
Execution mode
Choose the execution mode for your pipeline. This determines how the pipeline is run.
Superseded
A more recent execution can overtake an older one. This is the default.
Queued (Pipeline type V2 required)


for these repositories:
All repositories
This applies to all current and future repositories owned by the resource owner.
Also includes public repositories (read-only).
Only select repositories
Select at least one repository.
Also includes public repositories (read-only).
with these permissions:
Read access to issues and metadata
Read and write access to administration, code, commit statuses, pull requests, and repository hooks
Install
Cancel
Next: you'll be directed to the GitHub App's site to complete setup.

Deploy

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

AWS Elastic Beanstalk

Region

US East (N. Virginia)

Input artifacts
Choose an input artifact for this action. Learn more

SourceArtifact

No more than 100 characters

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

Application1

Environment name
Choose an environment that you have already created in the AWS Elastic Beanstalk console. Or create an environment in the AWS Elastic Beanstalk console and then return to this task.

Application1-env

Success
Congratulations! The pipeline pipeline1 has been created.

Create a notification rule for this pipeline

Developer Tools > CodePipeline > Pipelines > pipeline1

pipeline1

Notify | Edit | Stop execution | Clone pipeline | Release change

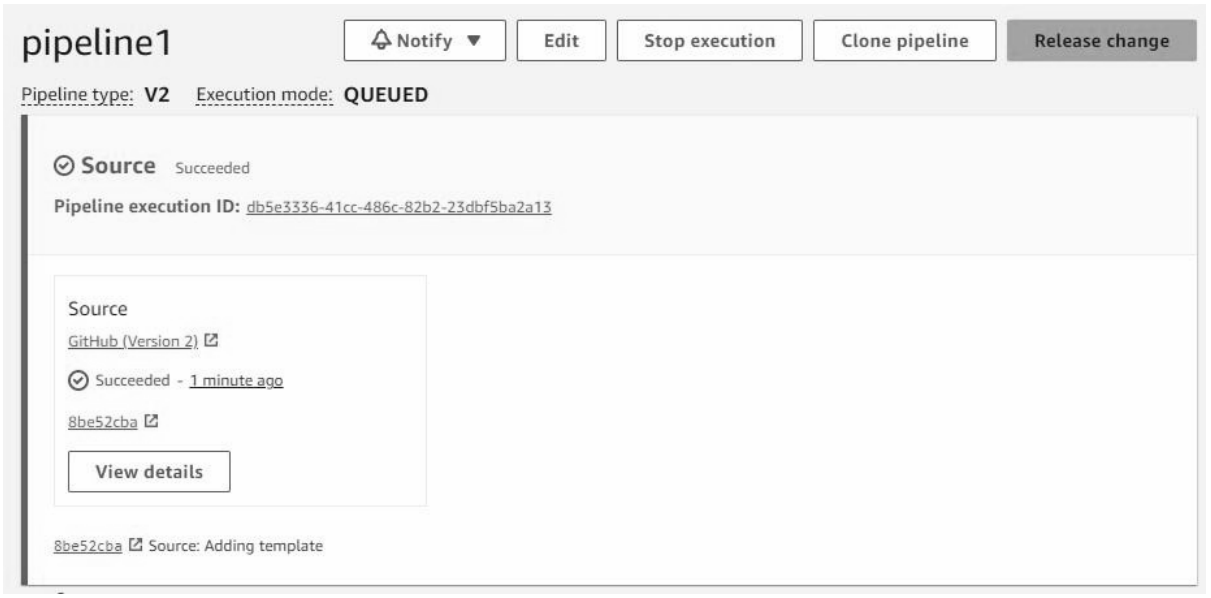
Pipeline type: **V2** Execution mode: **QUEUED**

Source Succeeded

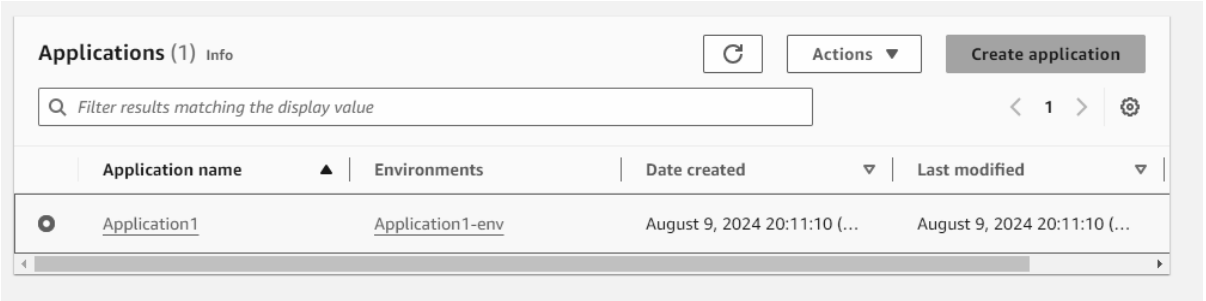
Pipeline execution ID: db5e3336-41cc-486c-82b2-23dbf5ba2a13

Source
GitHub (Version 2)
Succeeded - Just now
8be52cba
View details


Deploy
In progress
Pipeline execution ID: db5e3336-41cc-486c-82b2-23dbf5ba2a13
Deploy
AWS Elastic Beanstalk
In progress - Just now
View details
8be52cba
Source: Adding template


pipeline1
Notify
Edit
Stop execution
Clone pipeline
Release change
Pipeline type: V2
Execution mode: QUEUED
Source
Succeeded
Pipeline execution ID: db5e3336-41cc-486c-82b2-23dbf5ba2a13
Source
GitHub (Version 2)
Succeeded - 1 minute ago
8be52cba
View details
8be52cba
Source: Adding template


Deploy
Succeeded
Start rollback
Pipeline execution ID: db5e3336-41cc-486c-82b2-23dbf5ba2a13
Deploy
AWS Elastic Beanstalk
Succeeded - Just now
View details
8be52cba
Source: Adding template


Applications (1) Info
Actions
Create application
Filter results matching the display value
1
Application name
Environments
Date created
Last modified
Application1
Application1-env
August 9, 2024 20:11:10 (...
August 9, 2024 20:11:10 (...


Congratulations!
You have successfully created a pipeline that retrieved this source application from an Amazon S3 bucket and deployed it to three Amazon EC2 instances using AWS CodeDeploy.
For next steps, read the AWS CodePipeline Documentation.