Experiment No. 1

Aim: To understand the benefits of Cloud Infrastructure and Setup AWS Cloud9 IDE, Launch AWS Cloud9 IDE and Perform Collaboration Demonstration.

Theory:

What is cloud infrastructure?

Cloud infrastructure is a term used to describe the components needed for cloud computing, which includes hardware, abstracted resources, storage, and network resources. Think of cloud infrastructure as the tools needed to build a cloud. In order to host services and applications in the cloud, you need cloud infrastructure.

Benefits of Cloud Infrastructure:

1. Cost Savings

If you are worried about the price tag that would come with making the switch to cloud computing, you aren't alone 20% of organisations are concerned about the initial cost of implementing a cloud-based server. But those who are attempting to weigh the advantages and disadvantages of using the cloud need to consider more factors than just initial price they need to consider ROI.

2. Security

Many organisations have security concerns when it comes to adopting a cloud-computing solution. After all, when files, programs, and other data aren't kept securely onsite, how can you know that they are being protected? If you can remotely access your data, then what's stopping a cybercriminal from doing the same thing? Well, quite a bit, actually.

3. Flexibility

Your business has only a finite amount of focus to divide between all of its responsibilities. If your current IT solutions are forcing you to commit too much of your attention to computer and data-storage issues, then you aren't going to be able to concentrate on reaching business goals and satisfying customers. On the other hand, by relying on an outside organisation to take care of all IT hosting and infrastructure, you'll have more time to devote toward the aspects of your business that directly affect your bottom line.

4. Mobility

Cloud computing allows mobile access to corporate data via smartphones and devices, which, considering over 2.6 billion smartphones are being used globally today, is a great way to ensure that no one is ever left out of the loop. Staff with busy schedules, or who live a long way away from the corporate office, can use this feature to keep instantly up to date with clients and co-worker.

5. Insight

As we move ever further into the digital age, it's becoming clearer and clearer that the old adage “knowledge is power” has taken on the more modern and accurate form: “Data is money.” Hidden within the millions of bits of data that surround your customer transactions and business process are nuggets of invaluable, actionable information just waiting to be identified and acted upon. Of course, sifting through that data to find these kernels can be very difficult, unless you have access to the right cloud-computing solution.

Setting up AWS Cloud9:

To use AWS Cloud9 as the only individual in your AWS account, create an AWS account if you don't already have one, and then sign in to the AWS Cloud9 console.

Step 1: Create an AWS account

To create an AWS account

1. Go to https://aws.amazon.com/.
2. Choose Sign in to the Console.
3. Choose Create a new AWS account.
4. Complete the process by following the on-screen directions. This includes giving AWS your email address and credit card information. You must also use your phone to enter a code that AWS gives you.

After you finish creating the account, AWS will send you a confirmation email. Do not go to the next step until you get this confirmation.

Step 2: Sign in to the AWS Cloud9 console with the AWS account root user

After you complete the previous step, you're ready to sign in to the AWS Cloud9 console with an AWS account root user and start using AWS Cloud9.

1. Open the AWS Cloud9 console, at https://console.aws.amazon.com/cloud9/.
2. Enter the email address for your AWS account, and then choose Next.

Note

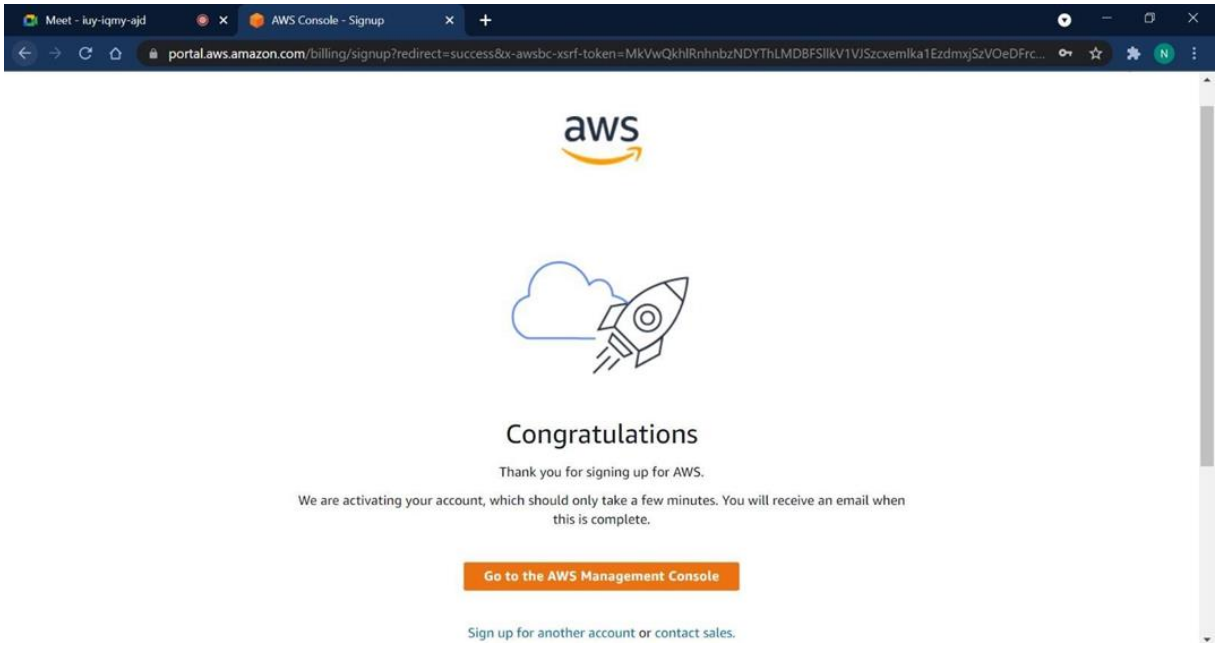
If an email address is already displayed and it's the wrong one, choose Sign in to a different account. Enter the correct email address, and then choose Next.

3. Enter the password for your AWS account, and then choose Sign In.

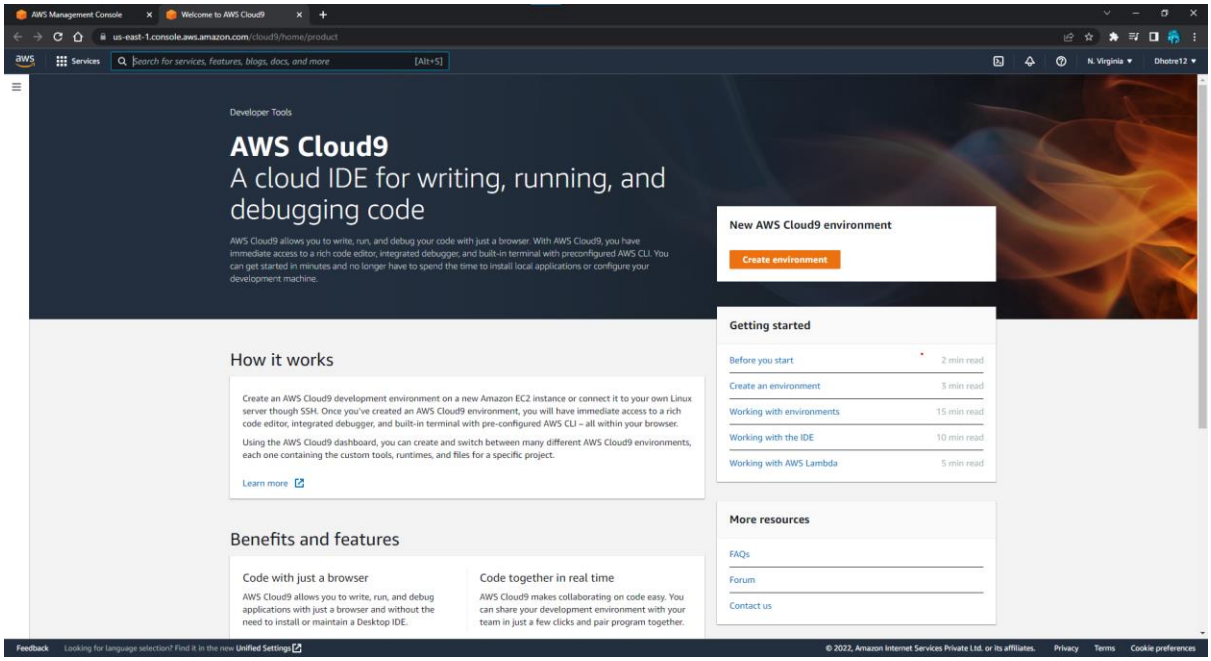
The AWS Cloud9 console is displayed, and you can now start using AWS Cloud9.

Results:

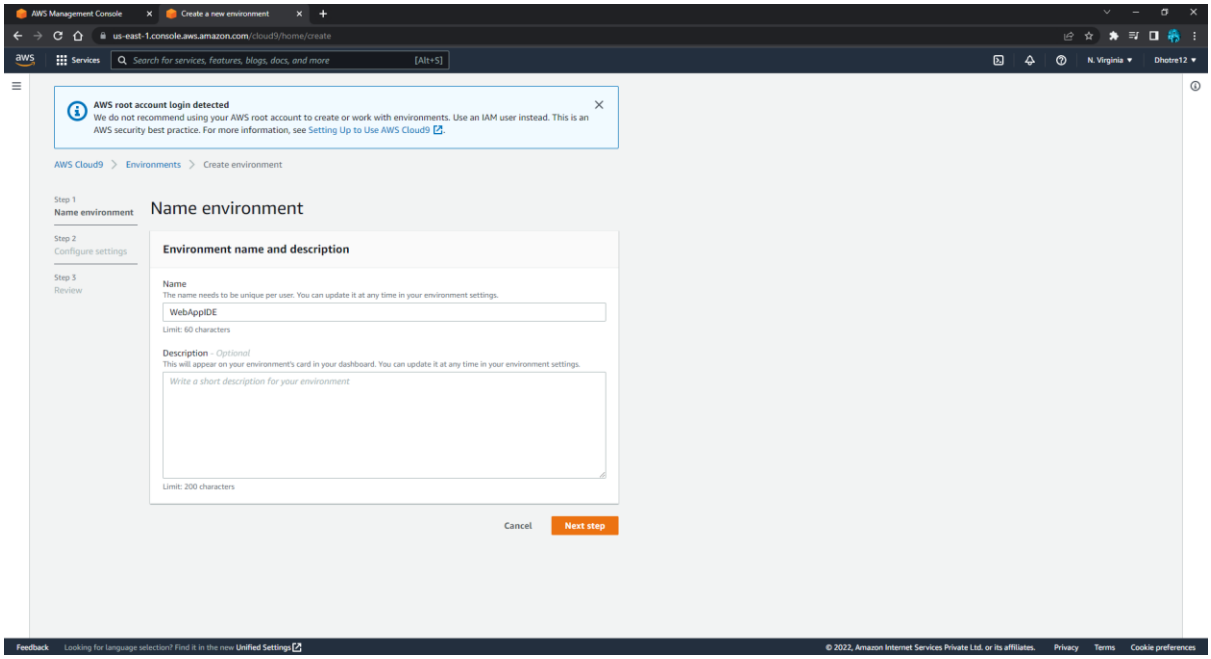
1. Account created

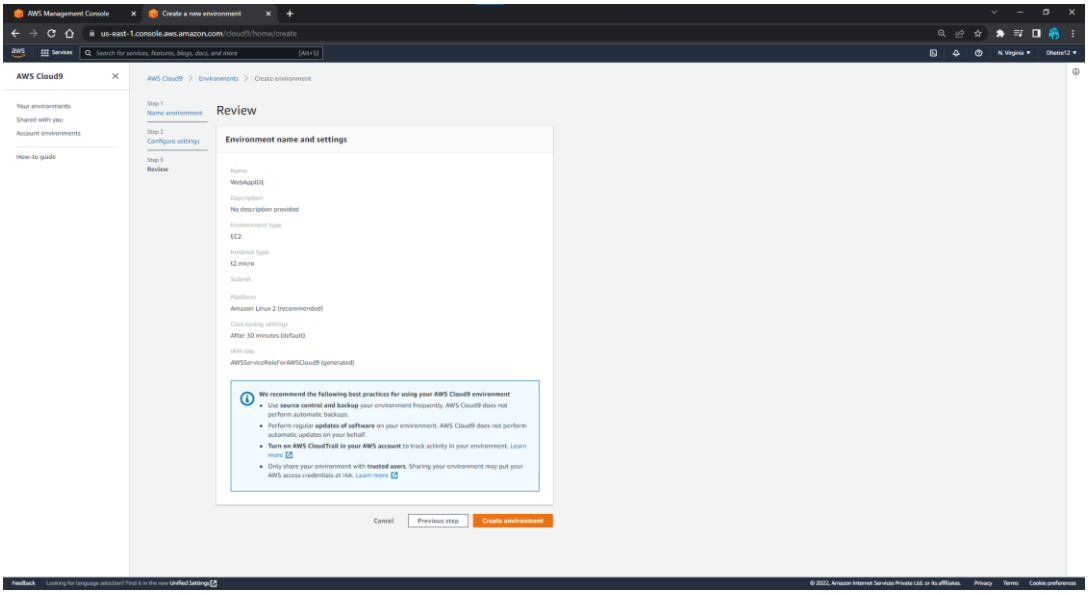


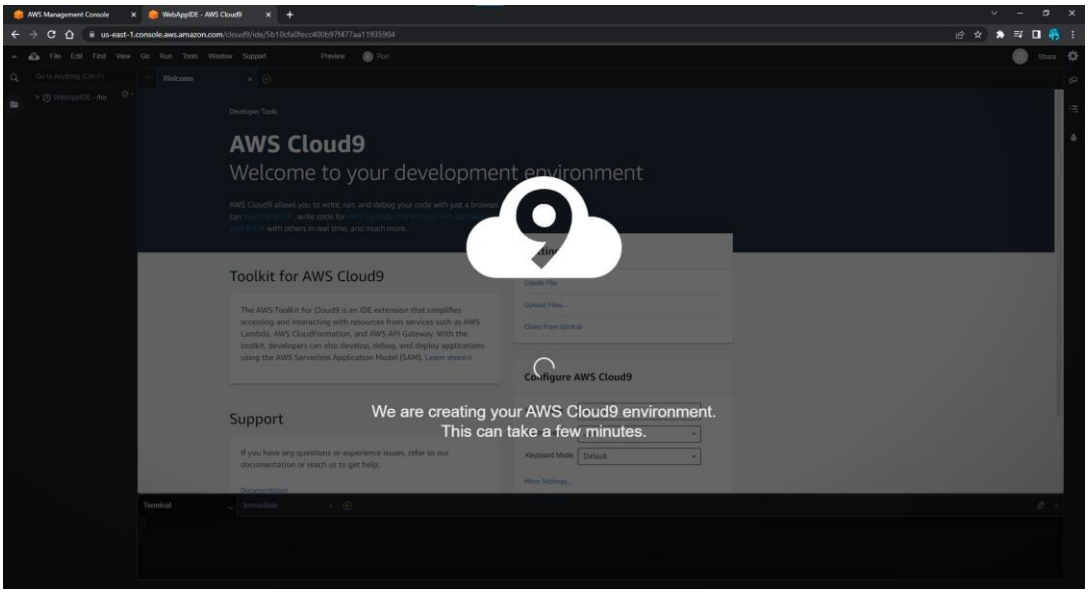
2. AWS Cloud9

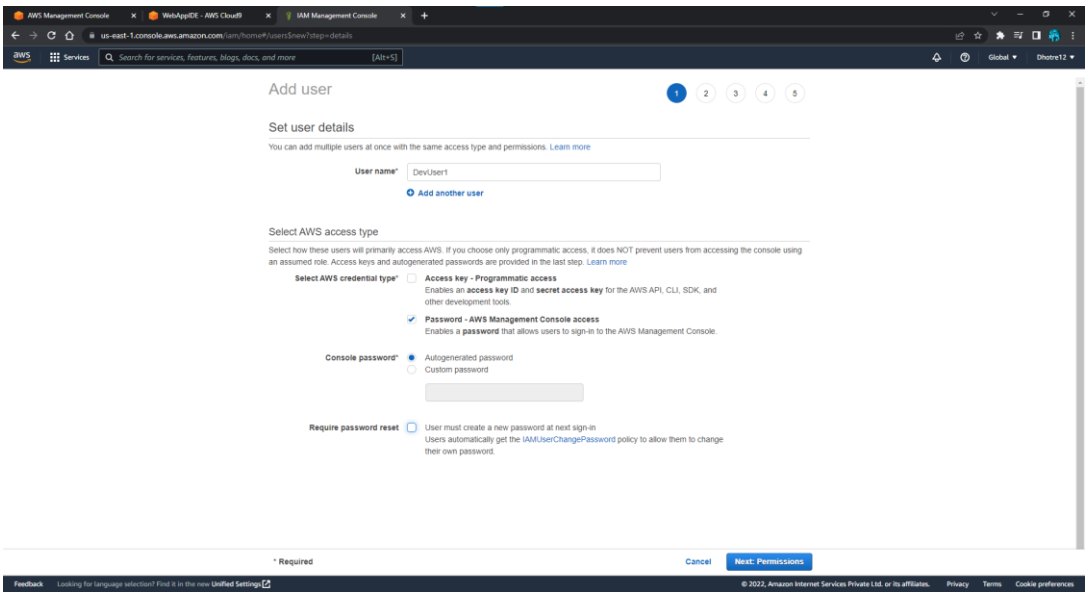


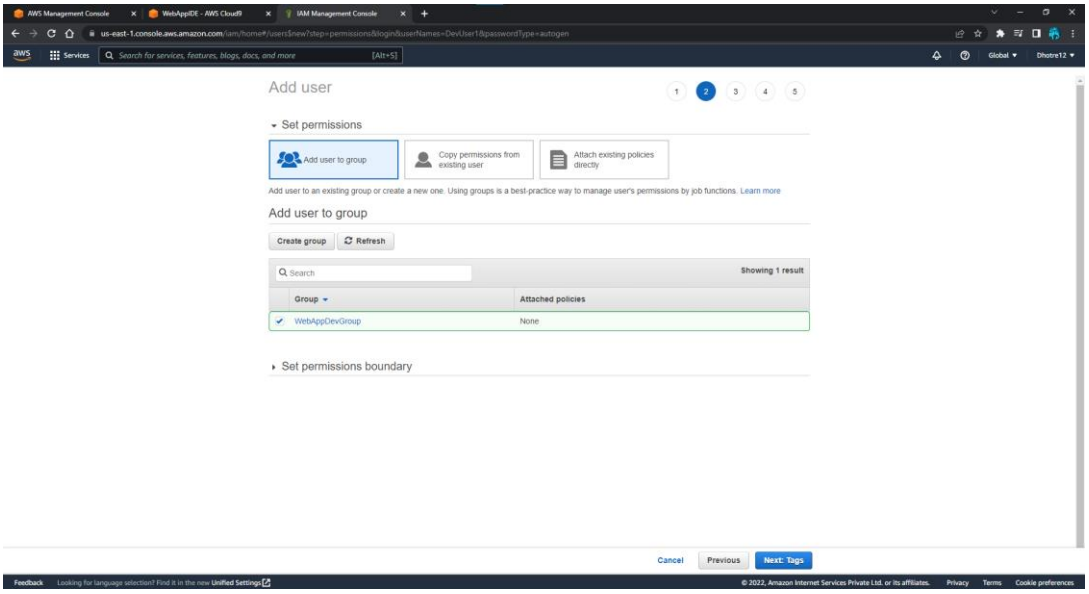
3. Creating Environment

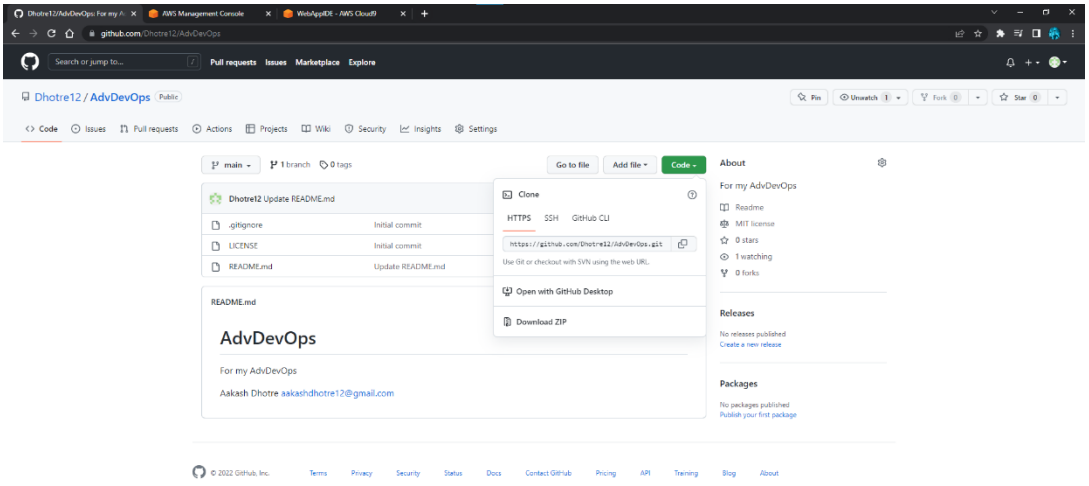


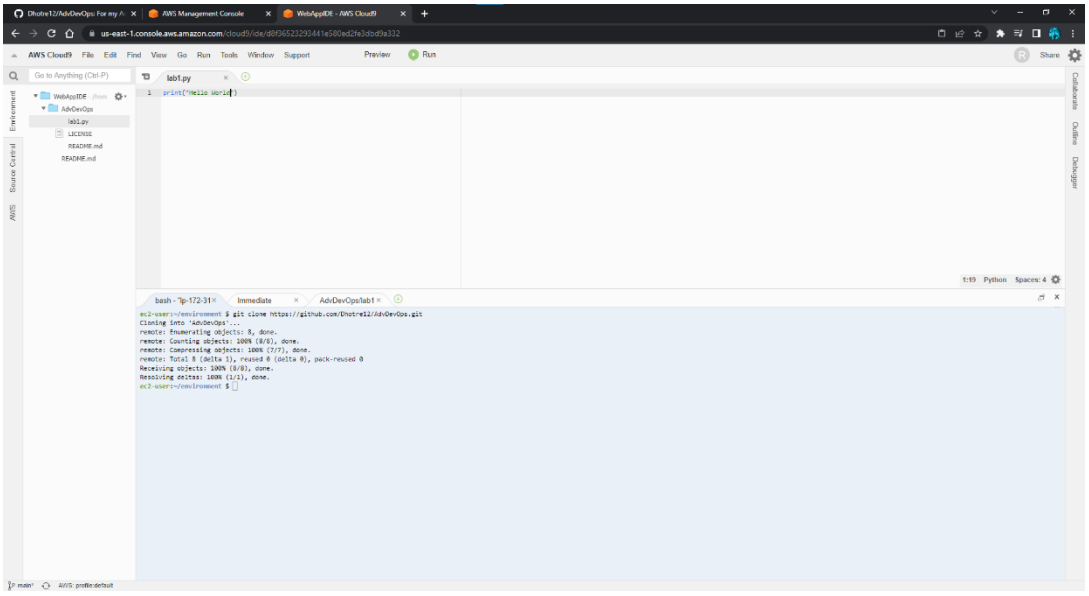

AWS Cloud9
Review
Environment name and settings
Name
WebAppIDE
Description
No description provided
Environment type
EC2
Instance type
t2.micro
Platform
Amazon Linux 2 (recommended)
Cost-saving settings
After 30 minutes (default)
IAM role
AWSServiceRoleForAWSCloud9 (generated)
We recommend the following best practices for using your AWS Cloud9 environment
Cancel
Previous step
Create environment

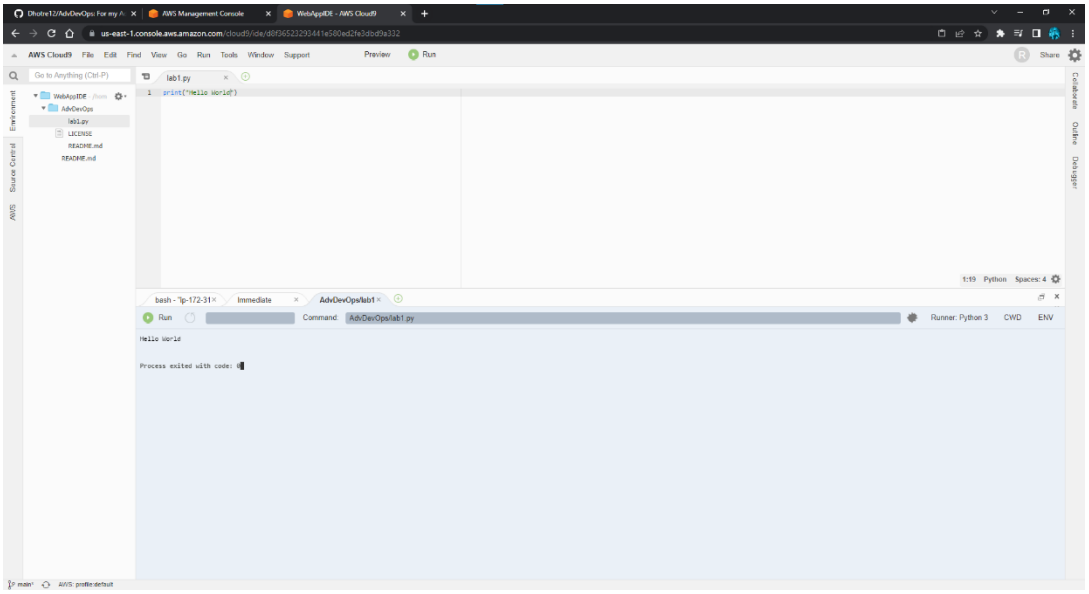

AWS Cloud9
Welcome to your development environment
Toolkit for AWS Cloud9
Configure AWS Cloud9
Support
We are creating your AWS Cloud9 environment.
This can take a few minutes.

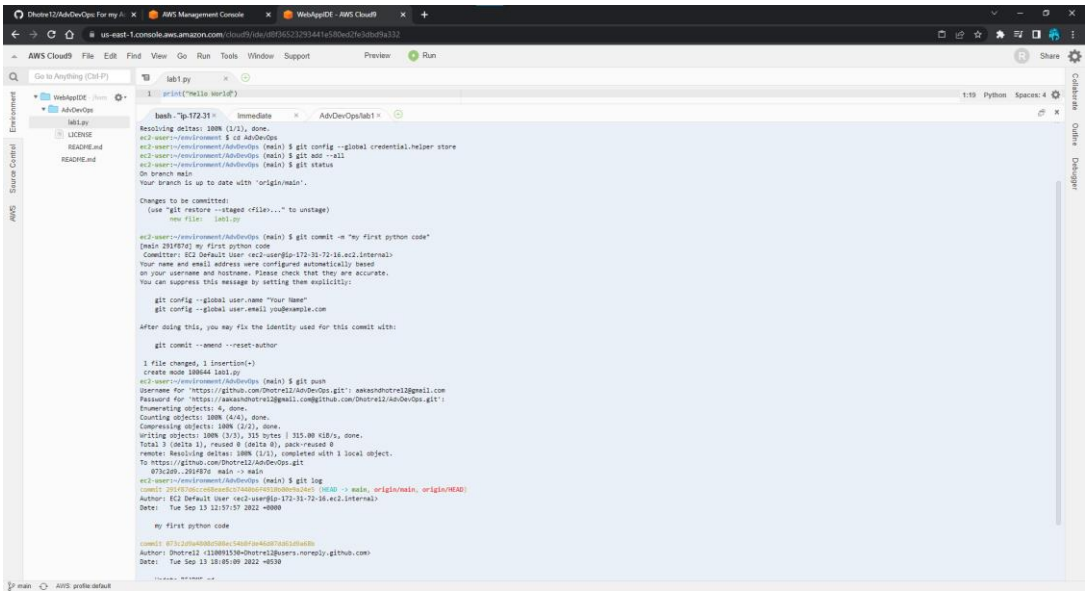

Add user
Set user details
You can add multiple users at once with the same access type and permissions. Learn more
User name*
DevUser1
Add another user
Select AWS access type
Select how these users will primarily access AWS. If you choose only programmatic access, it does NOT prevent users from accessing the console using an assumed role. Access keys and autogenerated passwords are provided in the last step. Learn more
Select AWS credential type*
Access key - Programmatic access
Enables an access key ID and secret access key for the AWS API, CLI, SDK, and other development tools.
Password - AWS Management Console access
Enables a password that allows users to sign-in to the AWS Management Console.
Console password*
Autogenerated password
Custom password
Require password reset
User must create a new password at next sign-in
Users automatically get the IAMUserChangePassword policy to allow them to change their own password.
* Required
Cancel
Next: Permissions

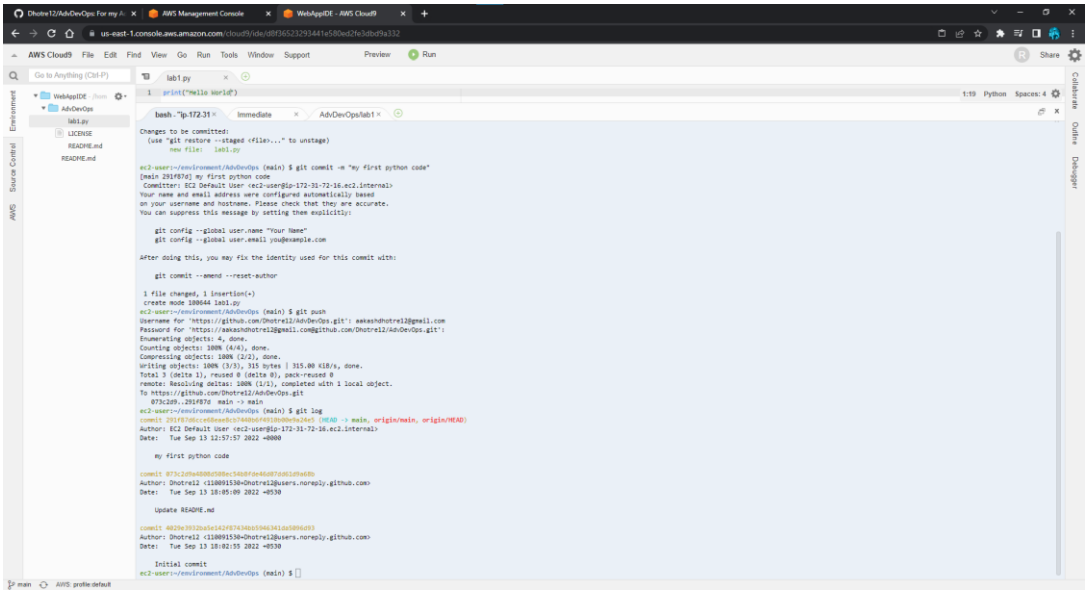

Add user
Set permissions
Add user to group
Copy permissions from existing user
Attach existing policies directly
Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. Learn more
Add user to group
Create group
Refresh
Search
Showing 1 result
Group
Attached policies
WebAppDevGroup
None
Set permissions boundary
Cancel
Previous
Next: Tags

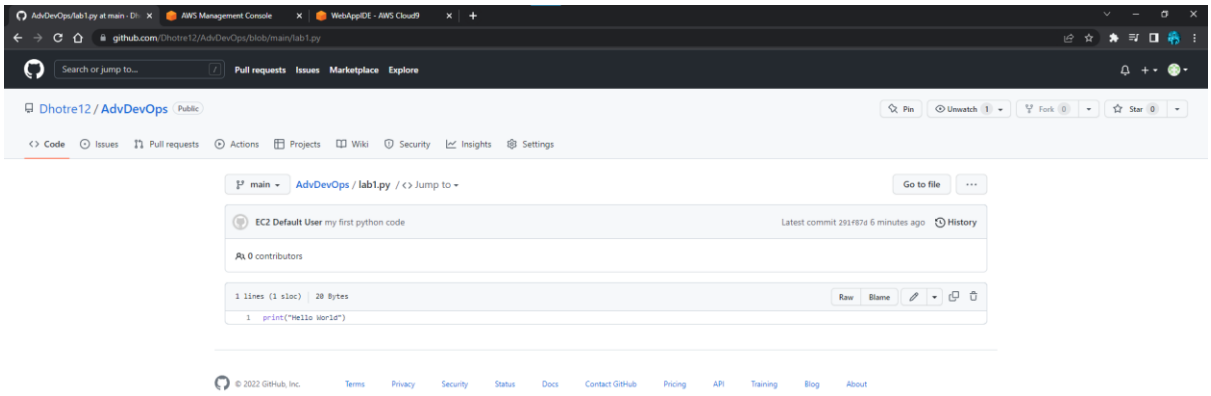

Dhotre12 / AdvDevOps
Public
Code
Issues
Pull requests
Actions
Projects
Wiki
Security
Insights
Settings
main
1 branch
0 tags
Go to file
Add file
Code
Dhotre12 Update README.md
.gitignore
Initial commit
LICENSE
Initial commit
README.md
Update README.md
Clone
HTTPS
SSH
GitHub CLI
https://github.com/Dhotre12/AdvDevOps.git
Use Git or checkout with SVN using the web URL.
Open with GitHub Desktop
Download ZIP
About
For my AdvDevOps
Readme
MIT license
0 stars
1 watching
0 forks
Releases
No releases published
Create a new release
Packages
No packages published
Publish your first package
README.md
AdvDevOps
For my AdvDevOps
Aakash Dhotre aakashdhotre12@gmail.com

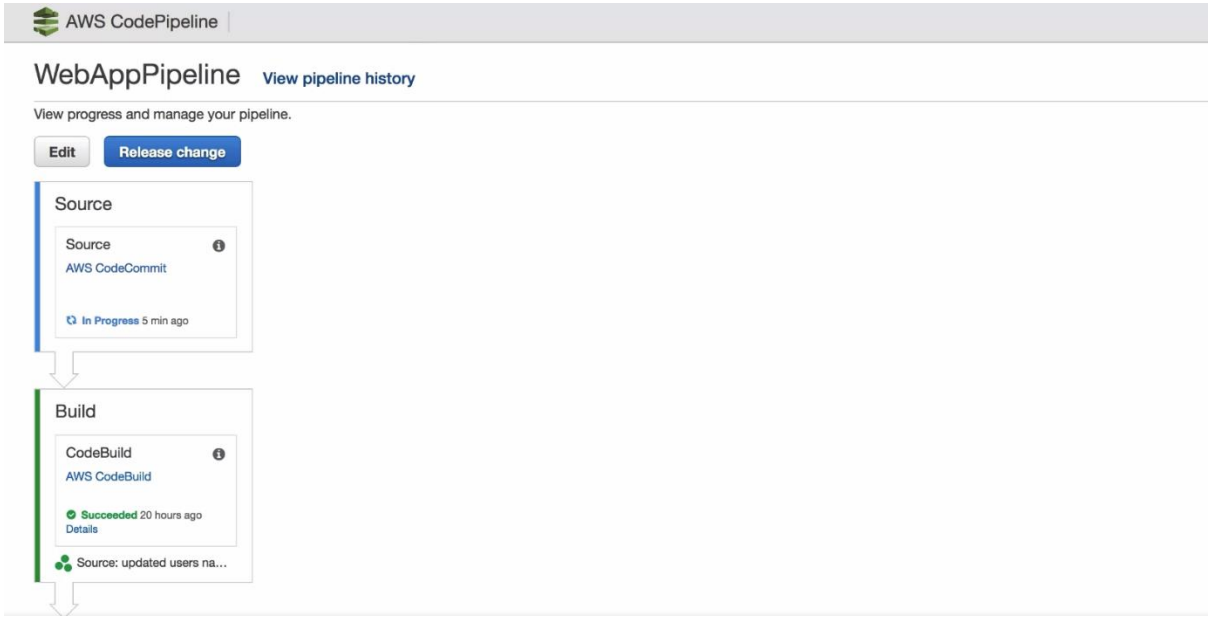

AWS Cloud9
File
Edit
Find
View
Go
Run
Tools
Window
Support
Preview
Run
Share
lab1.py
print("Hello World")
WebAppIDE
AdvDevOps
lab1.py
LICENSE
README.md
README.md
1:19
Python
Spaces: 4
bash - "ip-172-31
Immediate
AdvDevOps/lab1
ec2-user:~/environment $ git clone https://github.com/Dhotre12/AdvDevOps.git
Cloning into 'AdvDevOps'...
remote: Enumerating objects: 8, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (7/7), done.
remote: Total 8 (delta 1), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (8/8), done.
Resolving deltas: 100% (1/1), done.
ec2-user:~/environment $


lab1.py
print("Hello World")
Hello World
Process exited with code: 0


lab1.py
print("Hello World")
Resolving deltas: 100% (1/1), done.
ec2-user:~/environment $ cd AdvDevOps
ec2-user:~/environment/AdvDevOps (main) $ git config --global credential.helper store
ec2-user:~/environment/AdvDevOps (main) $ git add --all
ec2-user:~/environment/AdvDevOps (main) $ git status
On branch main
Your branch is up to date with 'origin/main'.
Changes to be committed:
(use "git restore --staged <file>..." to unstage)
new file: lab1.py
ec2-user:~/environment/AdvDevOps (main) $ git commit -m "my first python code"
[main 291f07d] my first python code
Committer: EC2 Default User <ec2-user@ip-172-31-72-16.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly:
git config --global user.name "Your Name"
git config --global user.email you@example.com
After doing this, you may fix the identity used for this commit with:
git commit --amend --reset-author
1 file changed, 1 insertion(+)
create mode 100644 lab1.py
ec2-user:~/environment/AdvDevOps (main) $ git push
Username for 'https://github.com/Dhotre12/AdvDevOps.git': aakashdhotre12@gmail.com
Password for 'https://aakashdhotre12@gmail.com@github.com/Dhotre12/AdvDevOps.git':
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 315 bytes | 315.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/Dhotre12/AdvDevOps.git
073c2d9..291f07d main -> main
ec2-user:~/environment/AdvDevOps (main) $ git log
commit 291f07dcce88eae8c5740bb5f4510e00e9a24e3 (HEAD -> main, origin/main, origin/HEAD)
Author: EC2 Default User <ec2-user@ip-172-31-72-16.ec2.internal>
Date: Tue Sep 13 12:57:57 2022 +0000
my first python code
commit 073c2d9e4808fd588ec5468fde464f874d61d9a63b
Author: Dhotre12 <110091530+Dhotre12@users.noreply.github.com>
Date: Tue Sep 13 18:05:09 2022 +0530


lab1.py
print("Hello World")
Changes to be committed:
(use "git restore --staged <file>..." to unstage)
new file: lab1.py
ec2-user:~/environment/AdvDevOps (main) $ git commit -m "my first python code"
[main 291f07d] my first python code
Committer: EC2 Default User <ec2-user@ip-172-31-72-16.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly:
git config --global user.name "Your Name"
git config --global user.email you@example.com
After doing this, you may fix the identity used for this commit with:
git commit --amend --reset-author
1 file changed, 1 insertion(+)
create mode 100644 lab1.py
ec2-user:~/environment/AdvDevOps (main) $ git push
Username for 'https://github.com/Dhotre12/AdvDevOps.git': aakashdhotre12@gmail.com
Password for 'https://aakashdhotre12@gmail.com@github.com/Dhotre12/AdvDevOps.git':
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 315 bytes | 315.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/Dhotre12/AdvDevOps.git
073c2d9..291f07d main -> main
ec2-user:~/environment/AdvDevOps (main) $ git log
commit 291f07dcce88eae8c5740bb5f4510e00e9a24e3 (HEAD -> main, origin/main, origin/HEAD)
Author: EC2 Default User <ec2-user@ip-172-31-72-16.ec2.internal>
Date: Tue Sep 13 12:57:57 2022 +0000
my first python code
commit 073c2d9e4808fd588ec5468fde464f874d61d9a63b
Author: Dhotre12 <110091530+Dhotre12@users.noreply.github.com>
Date: Tue Sep 13 18:05:09 2022 +0530
Update README.md
commit 4820e3932ba51e1429f874340b559603410a5096d93
Author: Dhotre12 <110091530+Dhotre12@users.noreply.github.com>
Date: Tue Sep 13 18:02:55 2022 +0530
Initial commit
ec2-user:~/environment/AdvDevOps (main) $





Conclusion: Hence, we have studied to understand the benefits of Cloud Infrastructure and Setup AWS Cloud9 IDE, Launch AWS Cloud9 IDE and Perform Collaboration Demonstration and learned to implement them.

Questionnaire:

1) How can I begin using AWS Cloud9?

Ans. You can choose AWS Cloud9 from the AWS Management Console after logging in. The console will walk you through the options for choosing a Linux server to connect to Cloud9. In a few simple steps, you may either create a new Amazon EC2 instance (AWS Cloud9 EC2 environment) or connect your current Linux server (AWS Cloud9 SSH environment). You may access your IDE and write code in a fully configured development environment once you've created a Cloud9 environment.

2) Who should utilise Amazon Web Services Cloud9?

Ans. AWS Cloud9 is available to everyone who writes code. Cloud9 provides instant access to a fully configured development environment in their browsers with preloaded runtimes, package managers, and debugging tools for those developing apps in Node.js (JavaScript), Python, PHP, Ruby, Go, and C++. Cloud9 allows you to access your work environment from any internet-connected computer, removing the need for a dedicated development machine.

AWS Cloud9 provides convenient access to their AWS resources via a pre-set AWS Command Line Interface (AWS CLI), ready to perform commands against AWS services, for AWS developers and those assessing new AWS services. Cloud9 has built-in tools for creating, editing, running, debugging, and deploying Lambda functions for those developing serverless apps on AWS Lambda using Node.js or Python.

3) What kinds of AWS Cloud9 development environments are there?

Ans. You can use one of two types of AWS Cloud9 setups.

AWS Cloud9 EC2 environment — Allows you to create a new Amazon EC2 instance with Cloud9. These instances are set to terminate 30 minutes after you close the IDE and start immediately when you open it.

SSH environment on AWS Cloud9 — Allows you to join an existing Linux server to Cloud9. On the Linux server that you intend to use with Cloud9 SSH environments, certain requirements are required.

4) What is AWS Cloud9, exactly?

Ans. AWS Cloud9 is a browser-based integrated development environment (IDE) that allows you to write, run, and debug code. It combines code completion, hinting, and step-through debugging with access to a full Linux server for running and storing code.

5) How can I execute my code?

Ans. The AWS Cloud9 IDE has a run button in the toolbar as well as built-in runners for over 10 languages that will immediately start your application with the most recent code modifications. You can also alter current runners, develop your own runners, or run your code from the terminal if you want complete control over how your product is run.

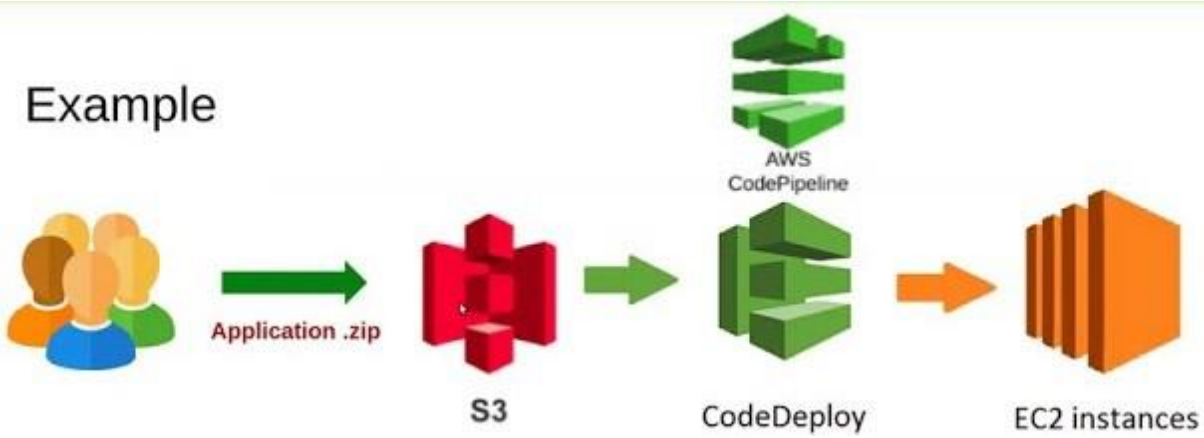
Experiment 2

Aim: To Build Your Application using AWS CodeBuild and Deploy on S3 / SEBS using AWS CodePipeline, deploy Sample Application on EC2 instance using AWS CodeDeploy.

Theory:

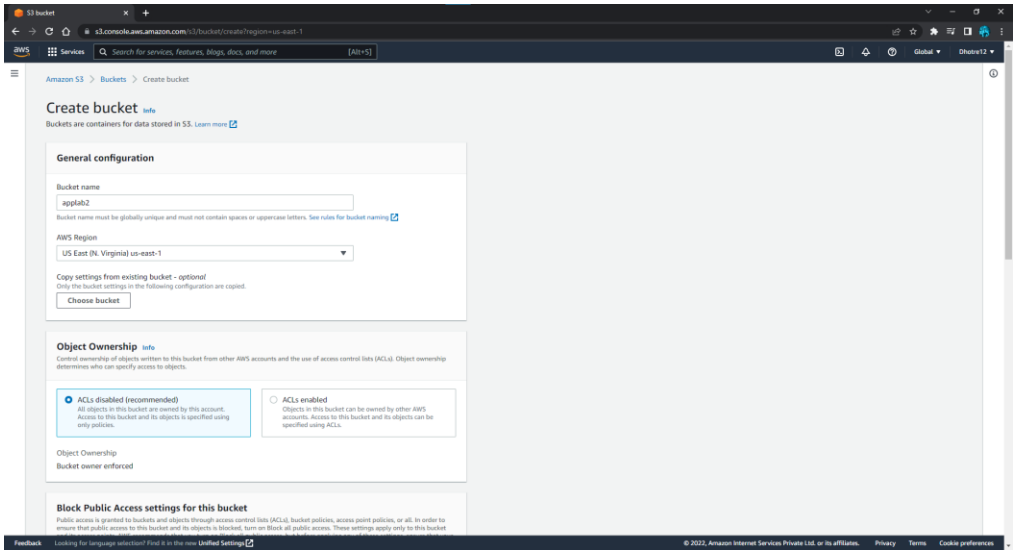
AWS CodeDeploy

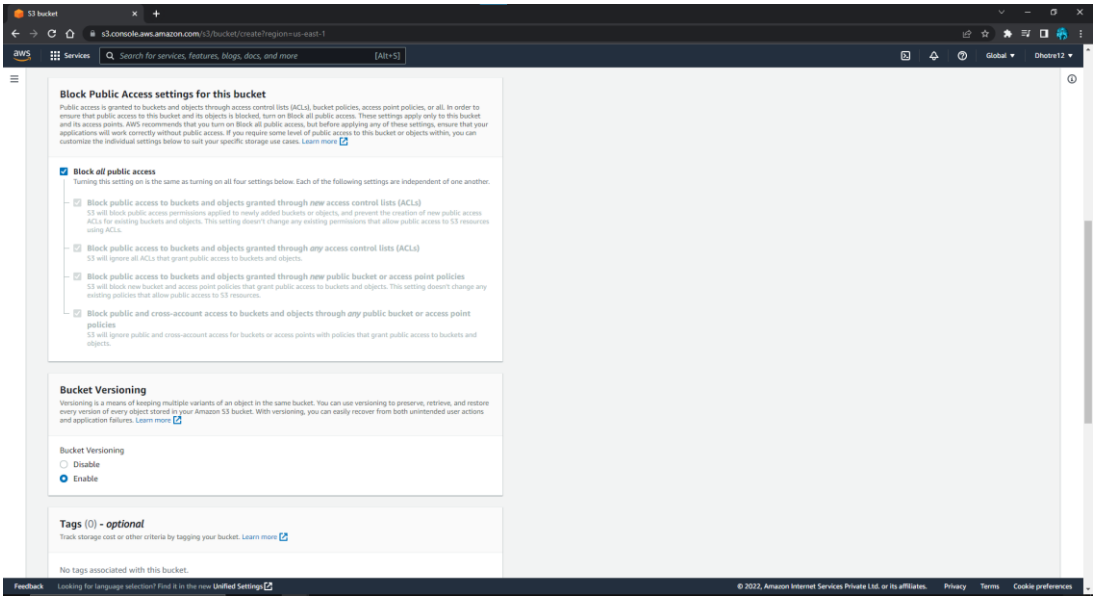
- AWS CodeDeploy is a fully managed deployment service that automates software deployments to a variety of compute services such as Amazon EC2, AWS Fargate, AWS Lambda, and you're on-premises servers.
- Code Deploy makes it easier for you to rapidly release new features, helps you avoid downtime during application deployment, and handles the complexity of updating your applications.
- **Centralized control** -AWS CodeDeploy allows you to easily launch and track the status of your application deployments through AWS management console.
- **Minimize downtime** - AWS CodeDeploy helps maximize your application availability during the software deployment process.

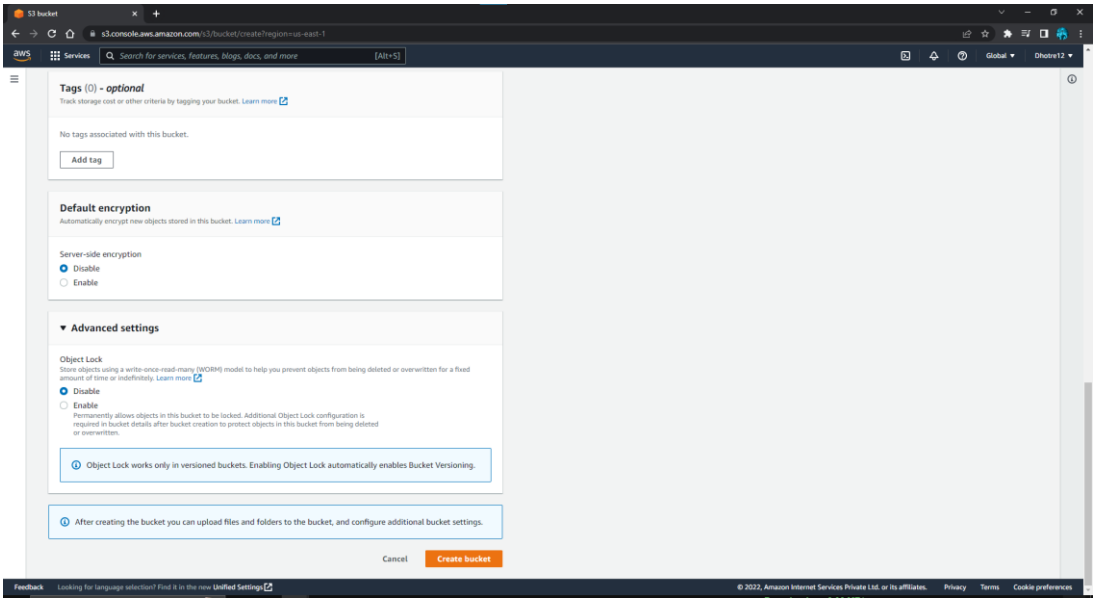


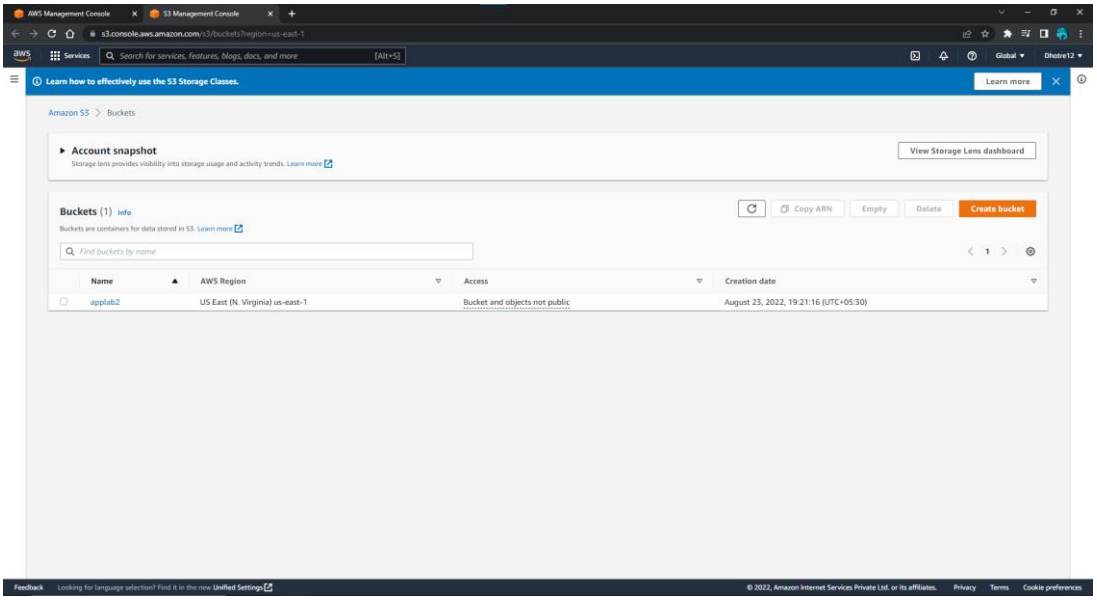
Result:

1. Go to S3 → Create Bucket →

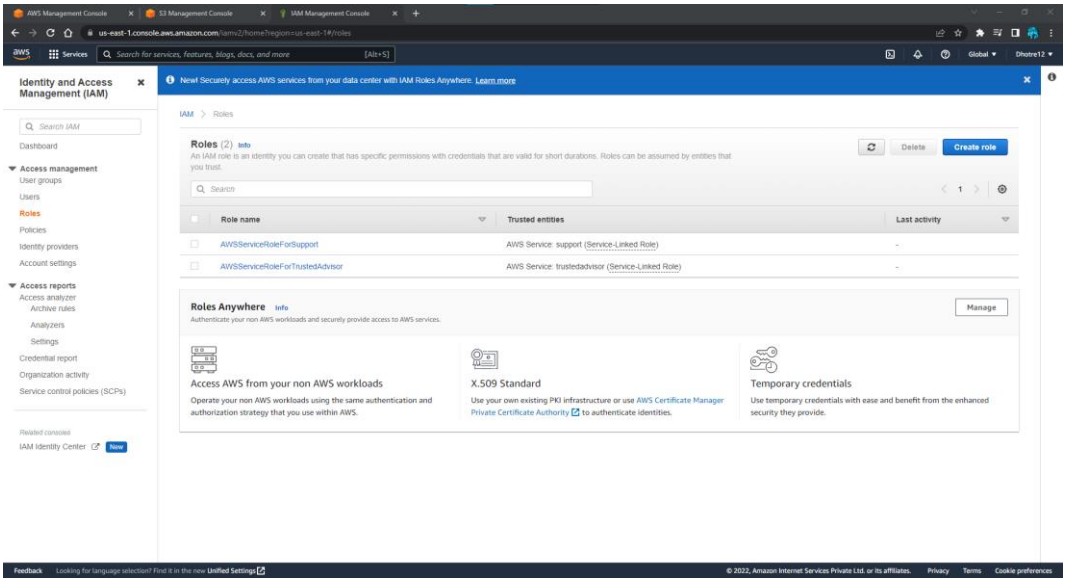


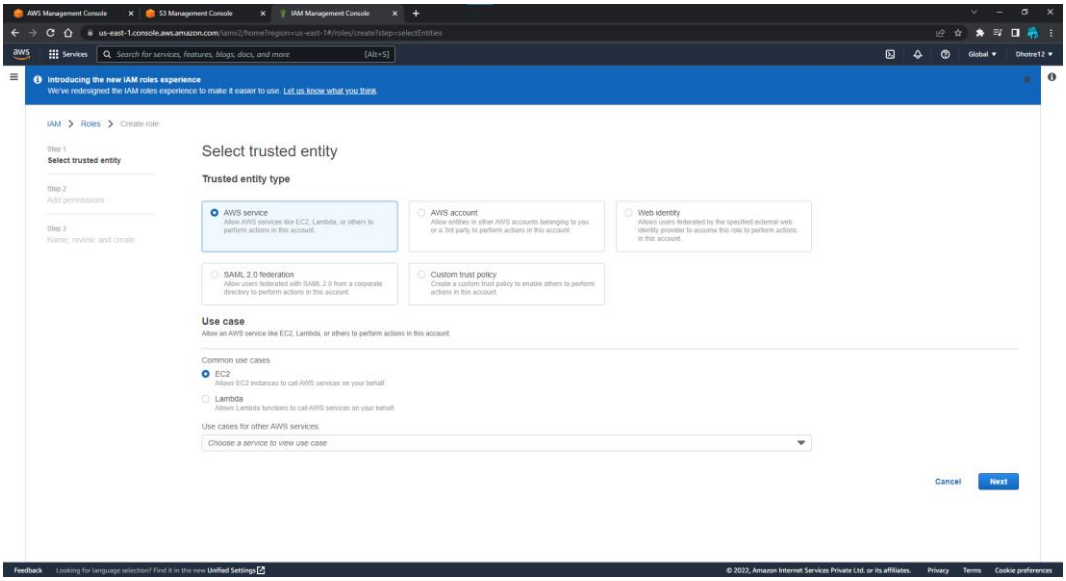

S3 bucket
s3.console.aws.amazon.com/s3/bucket/create?region=us-east-1
Services
Search for services, features, blogs, docs, and more
[Alt+S]
Global
Dhotre12
Block Public Access settings for this bucket
Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more
Block all public access
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.
Block public access to buckets and objects granted through new access control lists (ACLs)
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
Block public access to buckets and objects granted through any access control lists (ACLs)
S3 will ignore all ACLs that grant public access to buckets and objects.
Block public access to buckets and objects granted through new public bucket or access point policies
S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
Block public and cross-account access to buckets and objects through any public bucket or access point policies
S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.
Bucket Versioning
Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. Learn more
Bucket Versioning
Disable
Enable
Tags (0) - optional
Track storage cost or other criteria by tagging your bucket. Learn more
No tags associated with this bucket.
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Internet Services Private Ltd. or its affiliates.
Privacy
Terms
Cookie preferences

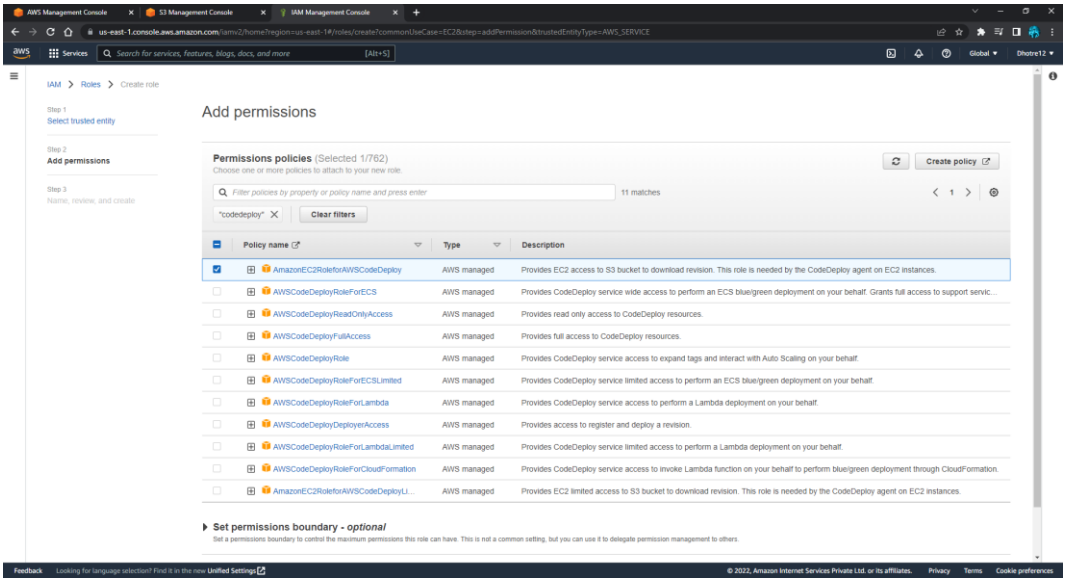

S3 bucket
s3.console.aws.amazon.com/s3/bucket/create?region=us-east-1
Services
Search for services, features, blogs, docs, and more
[Alt+S]
Global
Dhotre12
Tags (0) - optional
Track storage cost or other criteria by tagging your bucket. Learn more
No tags associated with this bucket.
Add tag
Default encryption
Automatically encrypt new objects stored in this bucket. Learn more
Server-side encryption
Disable
Enable
Advanced settings
Object Lock
Store objects using a write-once-read-many (WORM) model to help you prevent objects from being deleted or overwritten for a fixed amount of time or indefinitely. Learn more
Disable
Enable
Permanently allows objects in this bucket to be locked. Additional Object Lock configuration is required in bucket details after bucket creation to protect objects in this bucket from being deleted or overwritten.
Object Lock works only in versioned buckets. Enabling Object Lock automatically enables Bucket Versioning.
After creating the bucket you can upload files and folders to the bucket, and configure additional bucket settings.
Cancel
Create bucket
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Internet Services Private Ltd. or its affiliates.
Privacy
Terms
Cookie preferences

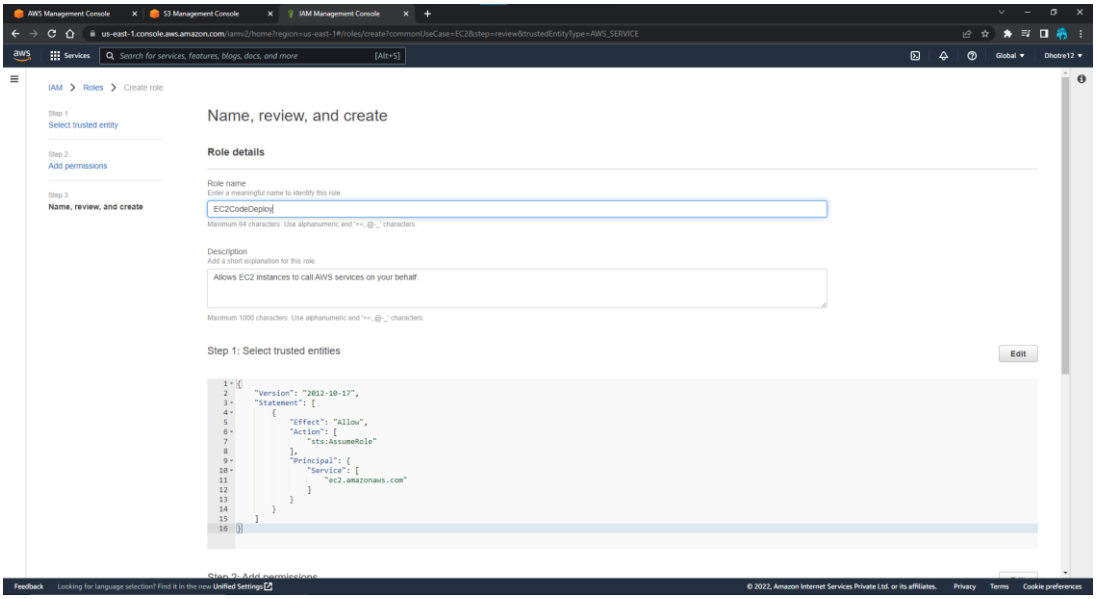

AWS Management Console
S3 Management Console
s3.console.aws.amazon.com/s3/buckets?region=us-east-1
Services
Search for services, features, blogs, docs, and more
[Alt+S]
Global
Dhotre12
Learn how to effectively use the S3 Storage Classes.
Learn more
Amazon S3
Buckets
Account snapshot
Storage lens provides visibility into storage usage and activity trends. Learn more
View Storage Lens dashboard
Buckets (1) Info
Buckets are containers for data stored in S3. Learn more
Copy ARN
Empty
Delete
Create bucket
Find buckets by name
Name
AWS Region
Access
Creation date
applab2
US East (N. Virginia) us-east-1
Bucket and objects not public
August 23, 2022, 19:21:16 (UTC+05:30)
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Internet Services Private Ltd. or its affiliates.
Privacy
Terms
Cookie preferences

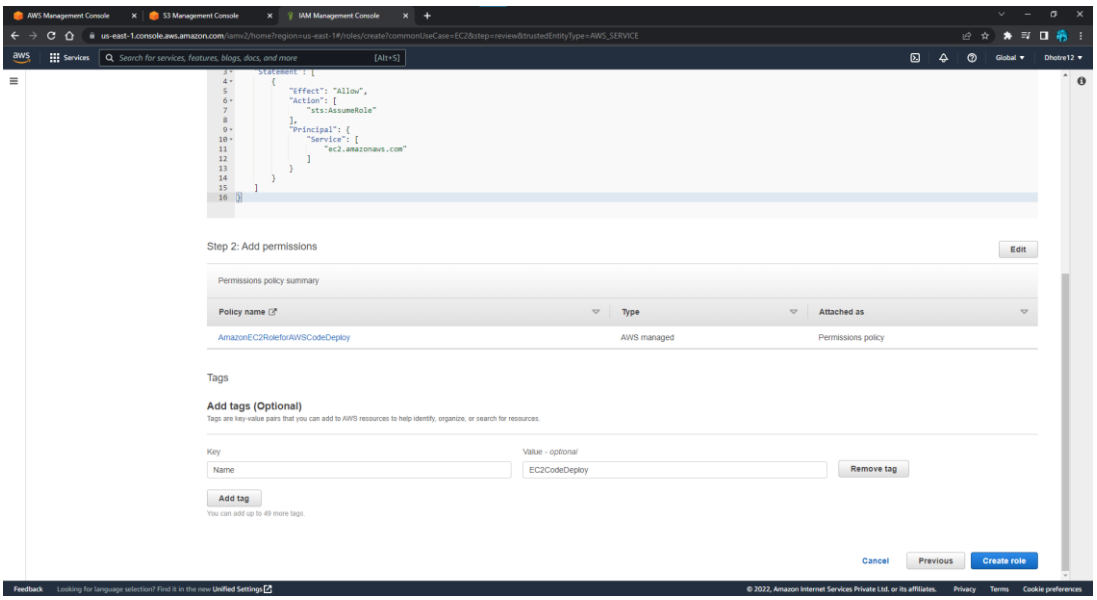
2. Setup IAM Roles

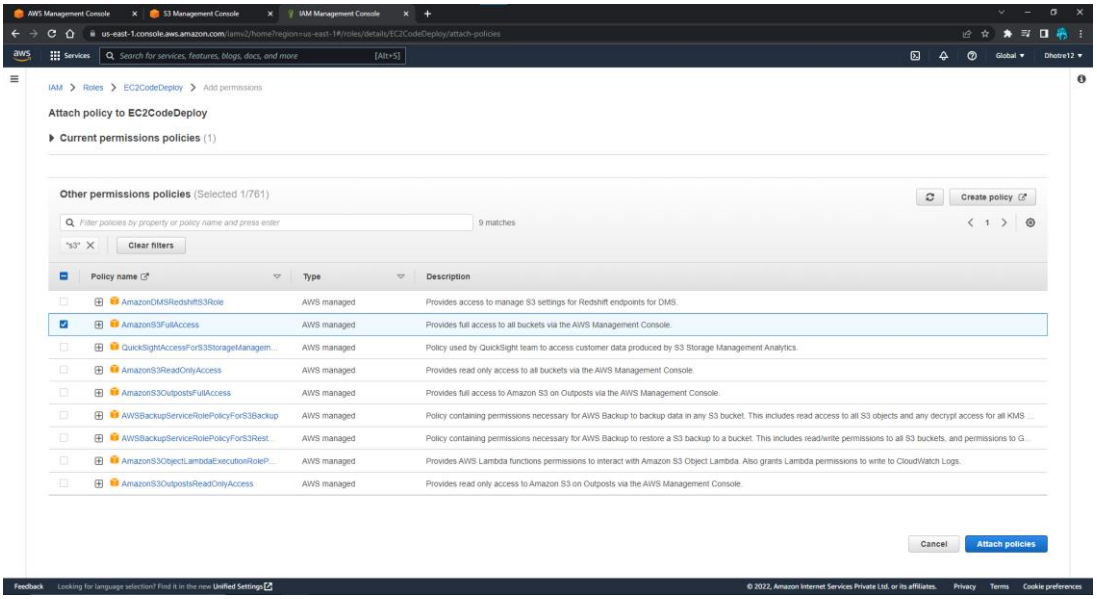


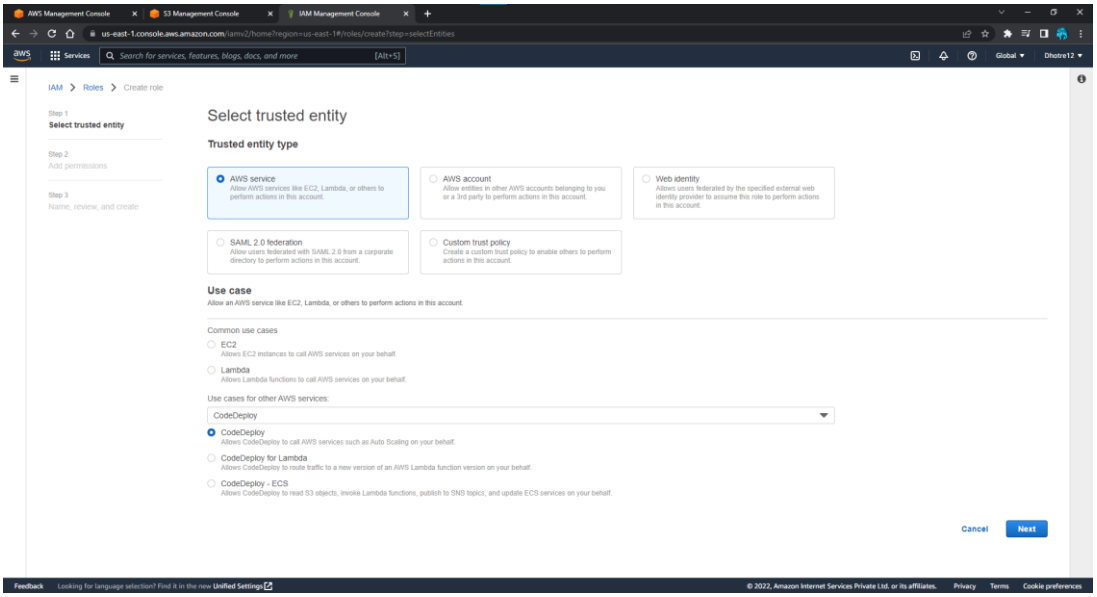


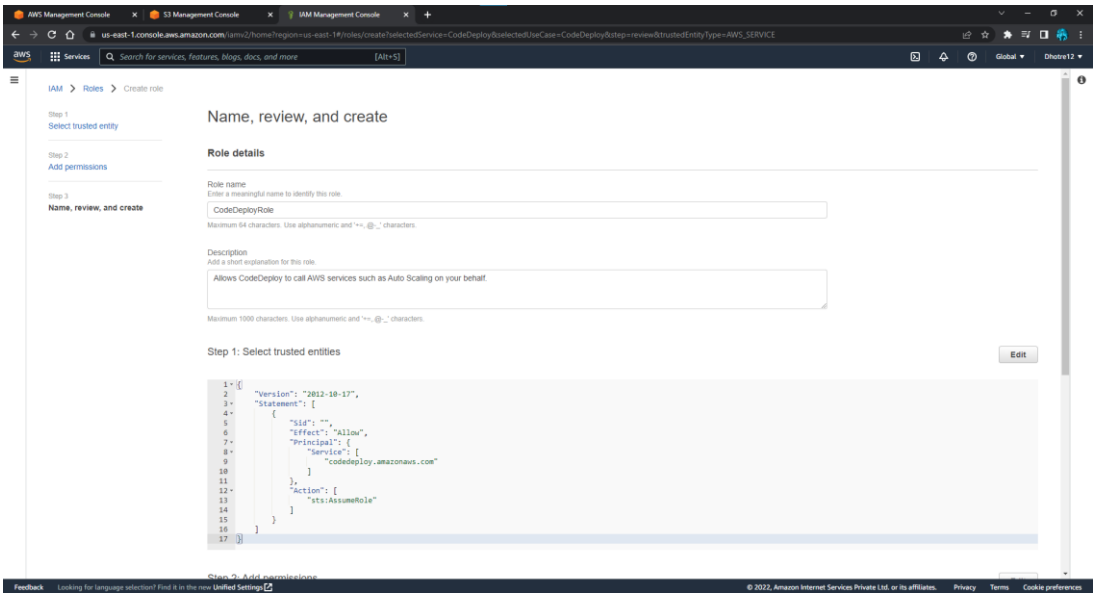


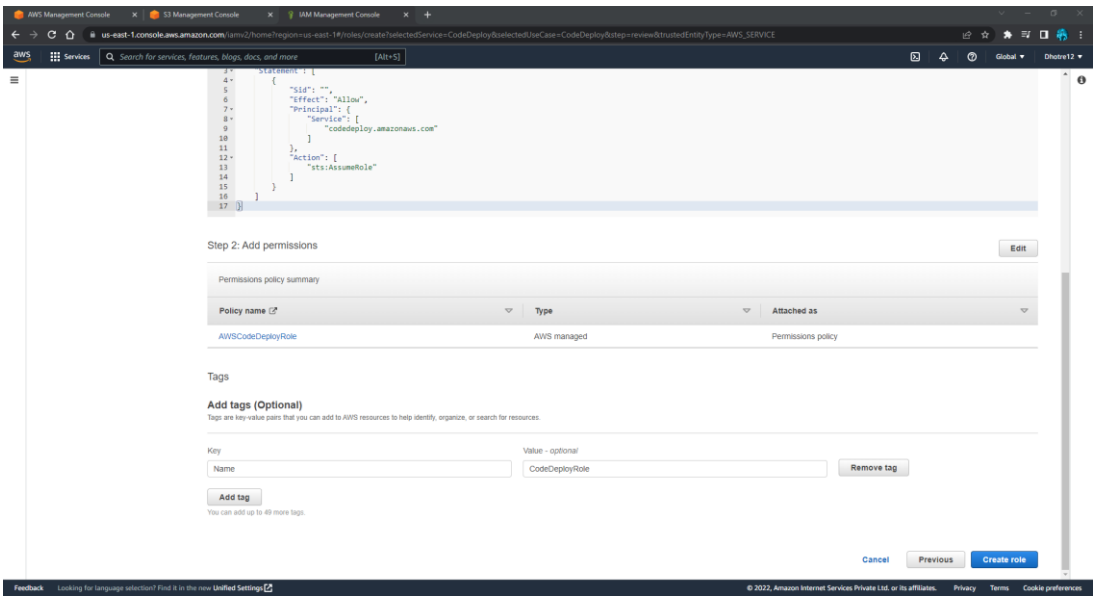

AWS Management Console
S3 Management Console
IAM Management Console
us-east-1.console.aws.amazon.com/iamv2/home?region=us-east-1#/roles/create?commonUseCase=EC2&step=review&trustedEntityType=AWS_SERVICE
Search for services, features, blogs, docs, and more [Alt+S]
Global
Dhotre12
IAM > Roles > Create role
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Name, review, and create
Role details
Role name
Enter a meaningful name to identify this role.
EC2CodeDeploy
Maximum 64 characters. Use alphanumeric and '+=,.@-_' characters.
Description
Add a short explanation for this role.
Allows EC2 instances to call AWS services on your behalf.
Maximum 1000 characters. Use alphanumeric and '+=,.@-_' characters.
Step 1: Select trusted entities
Edit
{
"Version": "2012-10-17",
"Statement": [
{
"Effect": "Allow",
"Action": [
"sts:AssumeRole"
],
"Principal": {
"Service": [
"ec2.amazonaws.com"
]
}
}
]
}
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Internet Services Private Ltd. or its affiliates.
Privacy
Terms
Cookie preferences

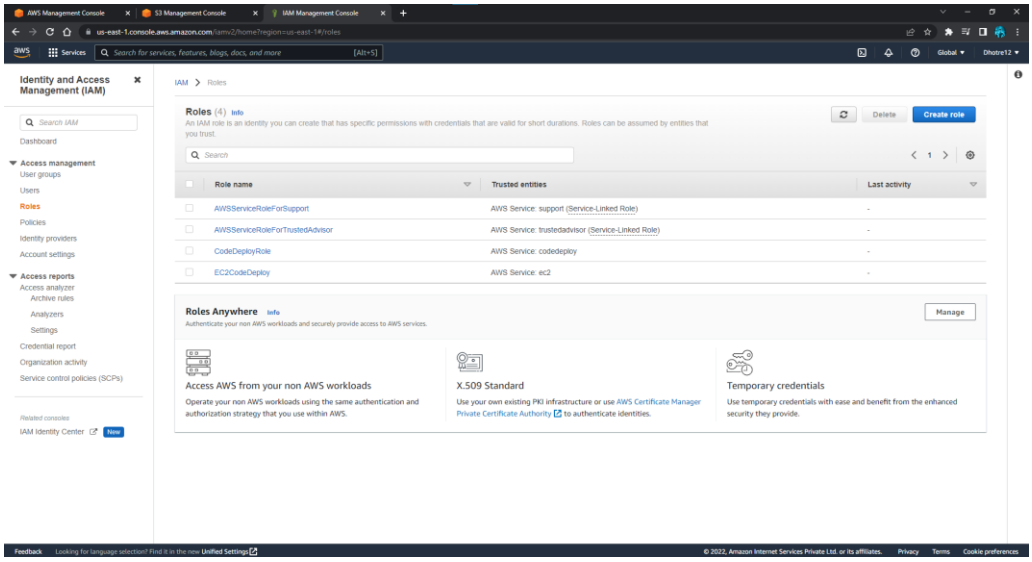

AWS Management Console
S3 Management Console
IAM Management Console
us-east-1.console.aws.amazon.com/iamv2/home?region=us-east-1#/roles/create?commonUseCase=EC2&step=review&trustedEntityType=AWS_SERVICE
Search for services, features, blogs, docs, and more [Alt+S]
Global
Dhotre12
"Statement": [
{
"Effect": "Allow",
"Action": [
"sts:AssumeRole"
],
"Principal": {
"Service": [
"ec2.amazonaws.com"
]
}
}
]
}
Step 2: Add permissions
Edit
Permissions policy summary
Policy name
Type
Attached as
AmazonEC2RoleforAWSCodeDeploy
AWS managed
Permissions policy
Tags
Add tags (Optional)
Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.
Key
Value - optional
Name
EC2CodeDeploy
Remove tag
Add tag
You can add up to 49 more tags.
Cancel
Previous
Create role
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Internet Services Private Ltd. or its affiliates.
Privacy
Terms
Cookie preferences


AWS Management Console
S3 Management Console
IAM Management Console
us-east-1.console.aws.amazon.com/iamv2/home?region=us-east-1#/roles/details/EC2CodeDeploy/attach-policies
Search for services, features, blogs, docs, and more [Alt+S]
Global
Dhotre12
IAM > Roles > EC2CodeDeploy > Add permissions
Attach policy to EC2CodeDeploy
Current permissions policies (1)
Other permissions policies (Selected 1/761)
Create policy
Filter policies by property or policy name and press enter
9 matches
"s3"
Clear filters
Policy name
Type
Description
AmazonDMSRedshiftS3Role
AWS managed
Provides access to manage S3 settings for Redshift endpoints for DMS.
AmazonS3FullAccess
AWS managed
Provides full access to all buckets via the AWS Management Console.
QuickSightAccessForS3StorageManagem...
AWS managed
Policy used by QuickSight team to access customer data produced by S3 Storage Management Analytics.
AmazonS3ReadOnlyAccess
AWS managed
Provides read only access to all buckets via the AWS Management Console.
AmazonS3OutpostsFullAccess
AWS managed
Provides full access to Amazon S3 on Outposts via the AWS Management Console.
AWSBackupServiceRolePolicyForS3Backup
AWS managed
Policy containing permissions necessary for AWS Backup to backup data in any S3 bucket. This includes read access to all S3 objects and any decrypt access for all KMS...
AWSBackupServiceRolePolicyForS3Rest...
AWS managed
Policy containing permissions necessary for AWS Backup to restore a S3 backup to a bucket. This includes read/write permissions to all S3 buckets, and permissions to G...
AmazonS3ObjectLambdaExecutionRolePol...
AWS managed
Provides AWS Lambda functions permissions to interact with Amazon S3 Object Lambda. Also grants Lambda permissions to write to CloudWatch Logs.
AmazonS3OutpostsReadOnlyAccess
AWS managed
Provides read only access to Amazon S3 on Outposts via the AWS Management Console.
Cancel
Attach policies
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Internet Services Private Ltd. or its affiliates.
Privacy
Terms
Cookie preferences


AWS Management Console
S3 Management Console
IAM Management Console
us-east-1.console.aws.amazon.com/iamv2/home?region=us-east-1#/roles/create?step=selectEntities
Search for services, features, blogs, docs, and more
IAM > Roles > Create role
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Select trusted entity
Trusted entity type
AWS service
Allow AWS services like EC2, Lambda, or others to perform actions in this account.
AWS account
Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.
Web identity
Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.
SAML 2.0 federation
Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.
Custom trust policy
Create a custom trust policy to enable others to perform actions in this account.
Use case
Allow an AWS service like EC2, Lambda, or others to perform actions in this account.
Common use cases
EC2
Allows EC2 instances to call AWS services on your behalf.
Lambda
Allows Lambda functions to call AWS services on your behalf.
Use cases for other AWS services:
CodeDeploy
CodeDeploy
Allows CodeDeploy to call AWS services such as Auto Scaling on your behalf.
CodeDeploy for Lambda
Allows CodeDeploy to route traffic to a new version of an AWS Lambda function version on your behalf.
CodeDeploy - ECS
Allows CodeDeploy to read S3 objects, invoke Lambda functions, publish to SNS topics, and update ECS services on your behalf.
Cancel
Next
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Internet Services Private Ltd. or its affiliates.
Privacy
Terms
Cookie preferences


AWS Management Console
S3 Management Console
IAM Management Console
us-east-1.console.aws.amazon.com/iamv2/home?region=us-east-1#/roles/create?selectedService=CodeDeploy&selectedUseCase=CodeDeploy&step=review&trustedEntityType=AWS_SERVICE
Search for services, features, blogs, docs, and more
IAM > Roles > Create role
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Name, review, and create
Role details
Role name
Enter a meaningful name to identify this role.
CodeDeployRole
Maximum 64 characters. Use alphanumeric and '+=,.@-_' characters.
Description
Add a short explanation for this role.
Allows CodeDeploy to call AWS services such as Auto Scaling on your behalf.
Maximum 1000 characters. Use alphanumeric and '+=,.@-_' characters.
Step 1: Select trusted entities
Edit
{
"Version": "2012-10-17",
"Statement": [
{
"Sid": "",
"Effect": "Allow",
"Principal": {
"Service": [
"codedeploy.amazonaws.com"
]
},
"Action": [
"sts:AssumeRole"
]
}
]
}
Step 2: Add permissions
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Internet Services Private Ltd. or its affiliates.
Privacy
Terms
Cookie preferences


AWS Management Console
S3 Management Console
IAM Management Console
us-east-1.console.aws.amazon.com/iamv2/home?region=us-east-1#/roles/create?selectedService=CodeDeploy&selectedUseCase=CodeDeploy&step=review&trustedEntityType=AWS_SERVICE
Search for services, features, blogs, docs, and more
"Statement": [
{
"Sid": "",
"Effect": "Allow",
"Principal": {
"Service": [
"codedeploy.amazonaws.com"
]
},
"Action": [
"sts:AssumeRole"
]
}
]
}
Step 2: Add permissions
Edit
Permissions policy summary
Policy name
Type
Attached as
AWSCodeDeployRole
AWS managed
Permissions policy
Tags
Add tags (Optional)
Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.
Key
Value - optional
Name
CodeDeployRole
Remove tag
Add tag
You can add up to 49 more tags.
Cancel
Previous
Create role
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Internet Services Private Ltd. or its affiliates.
Privacy
Terms
Cookie preferences



3. Setup EC2 as deployment server

a. Create EC2 instance with the following software packages should install

b. Choose AMI: Amazon Linux 2

c. Choose AMI role as EC2CodeDeploy

d. Choose User Data: for installing required packages.

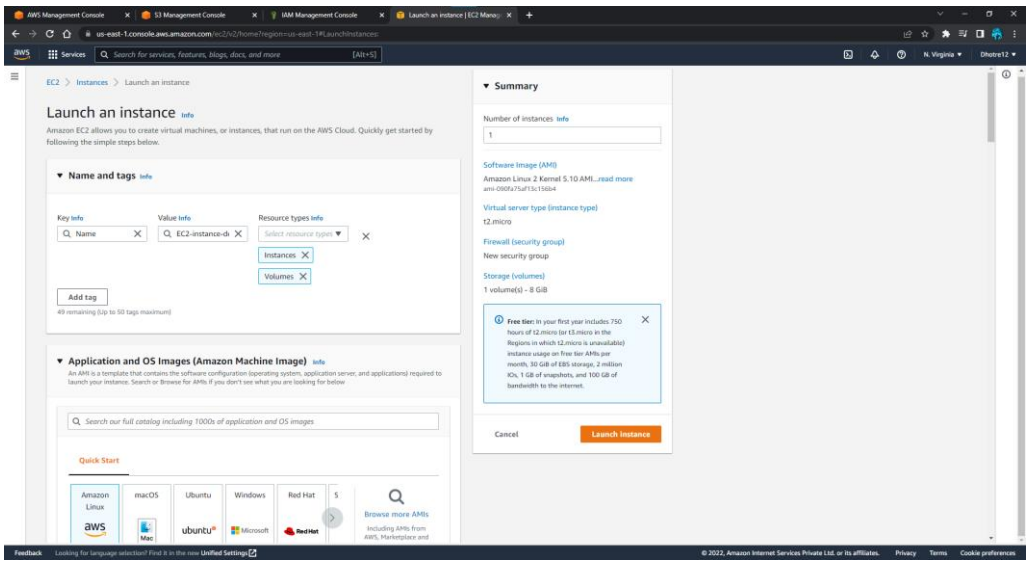
```
#!/bin/bash
sudo yum -y update
sudo yum -y install ruby
sudo yum -y install wget
cd /home/ec2-user
wget https://aws-codedeploy-ap-south-1.s3.ap-south-1.amazonaws.com/latest/install
sudo chmod +x ./install
sudo ./install auto
sudo yum install -y python-pip
sudo pip install awscli
```

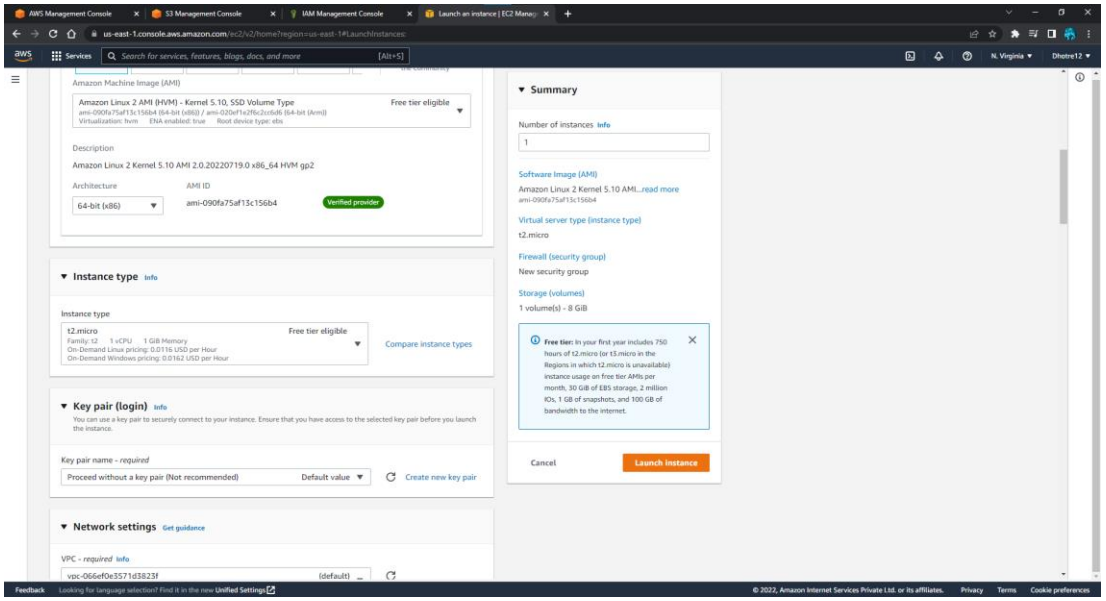
e. Security groups: which enable port SSH port 22 and HTTP 80 for application

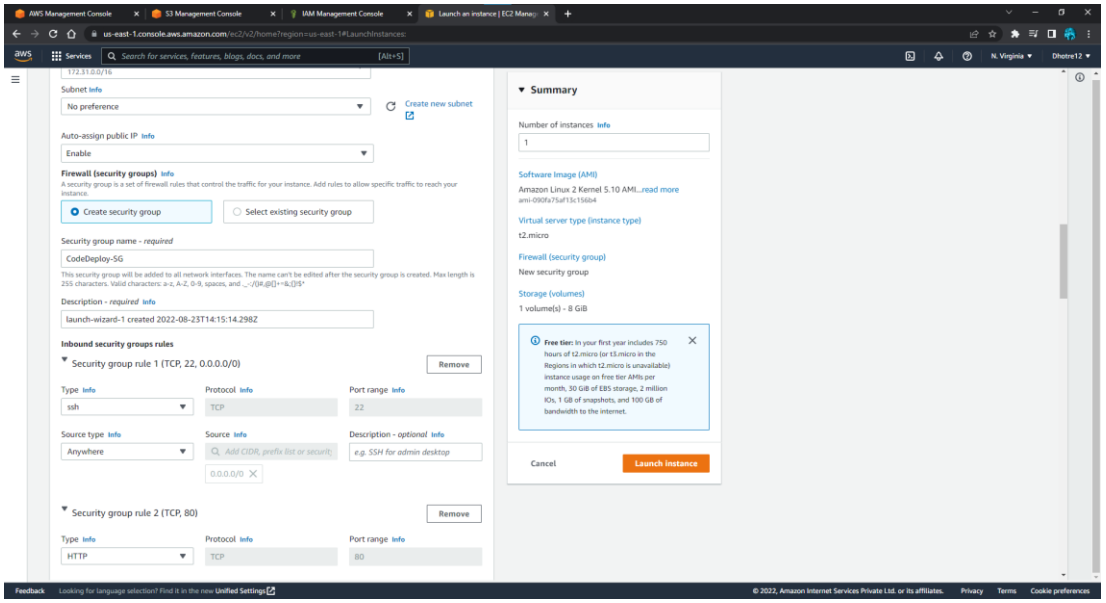
f. Add tags to your EC2 instance

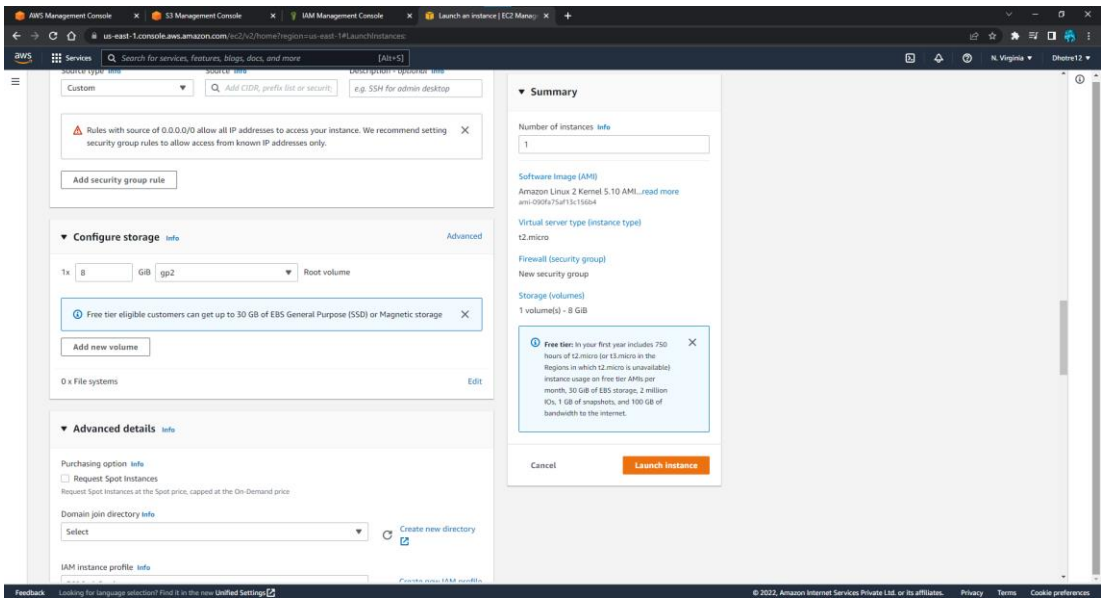
g. Launch instance

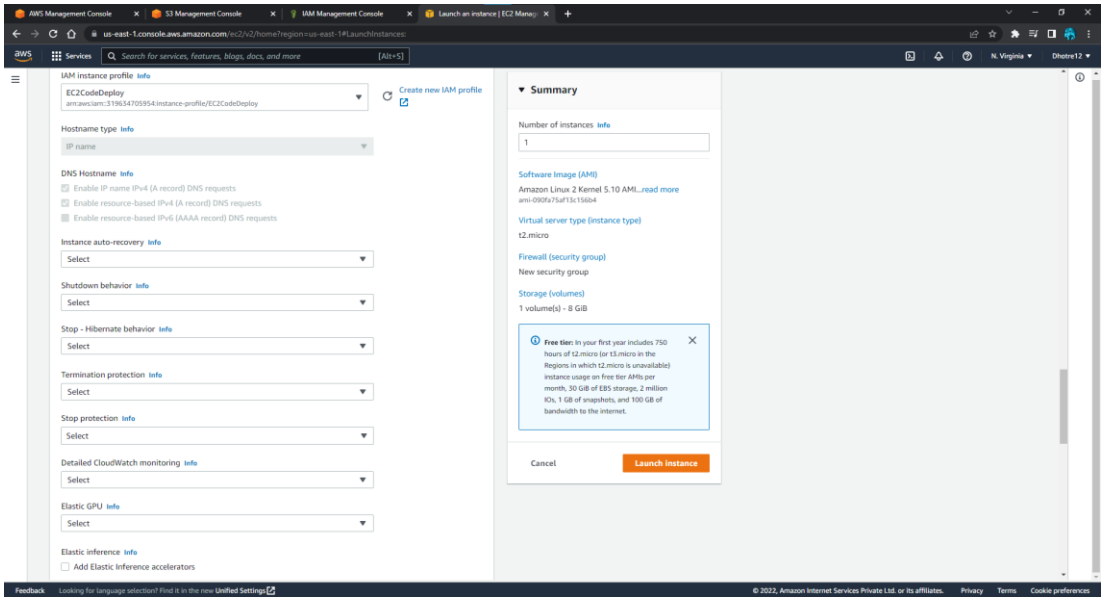
h. Make sure that your bucket should enabled version

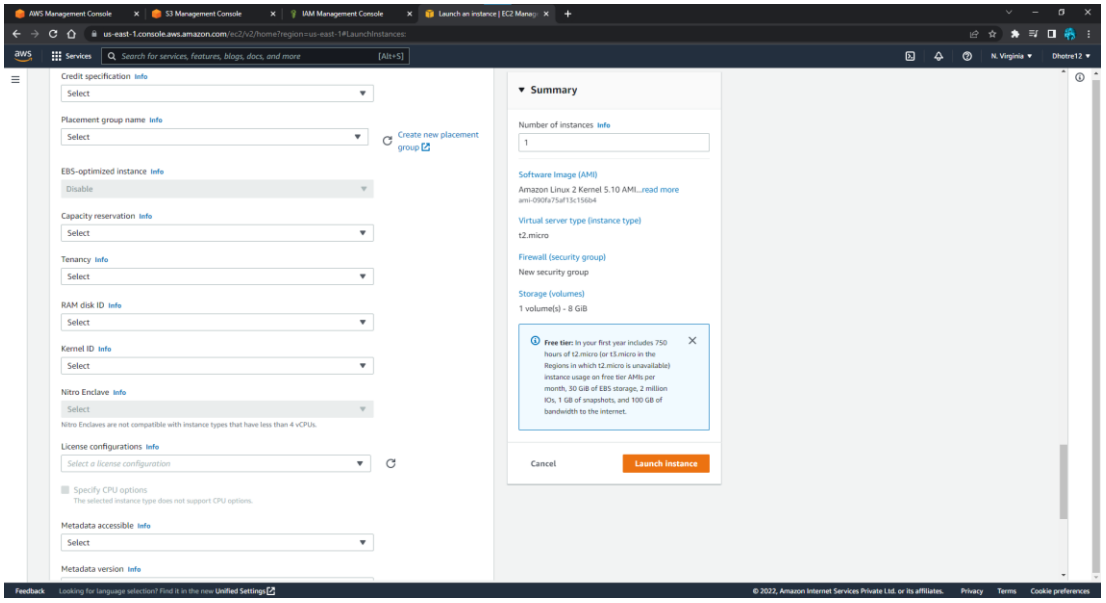


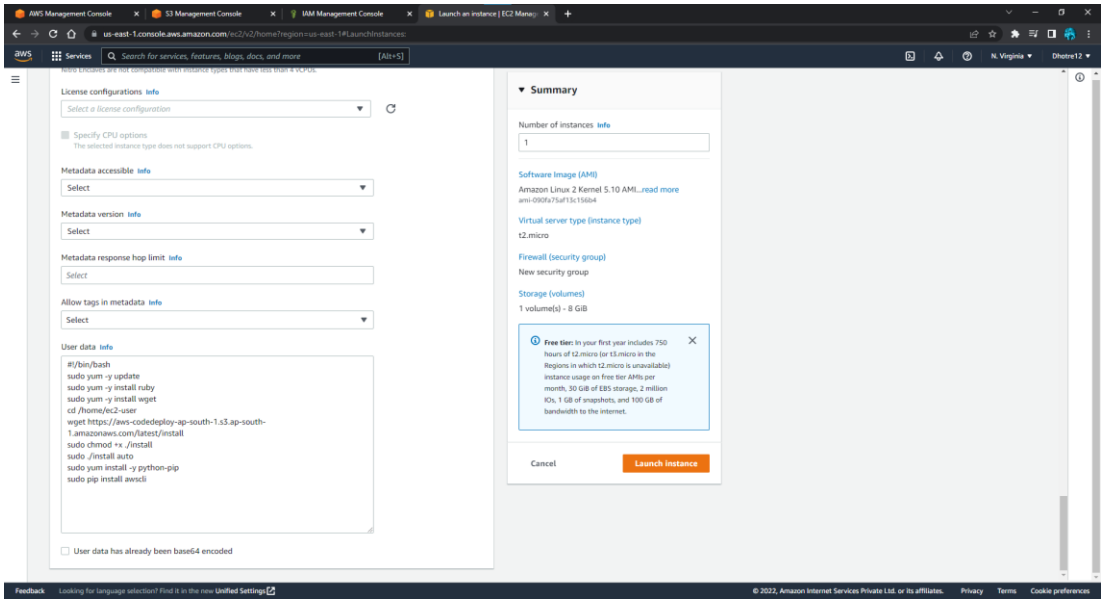

Amazon Machine Image (AMI)
Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type
Free tier eligible
Description
Amazon Linux 2 Kernel 5.10 AMI 2.0.20220719.0 x86_64 HVM gp2
Architecture
64-bit (x86)
AMI ID
ami-090fa75af13c156b4
Verified provider
Instance type
t2.micro
Free tier eligible
Compare instance types
Key pair (login)
Key pair name - required
Proceed without a key pair (Not recommended)
Default value
Create new key pair
Network settings
VPC - required
Summary
Number of instances
1
Software Image (AMI)
Amazon Linux 2 Kernel 5.10 AMI...read more
Virtual server type (instance type)
t2.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.
Cancel
Launch instance

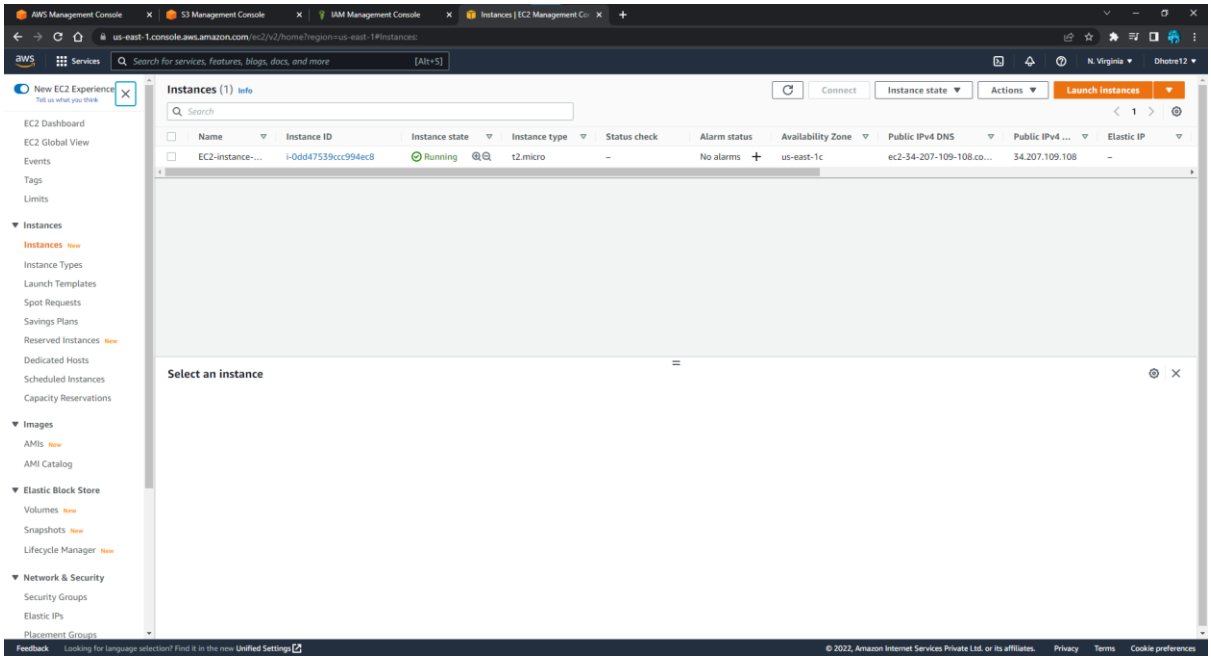

Subnet
No preference
Create new subnet
Auto-assign public IP
Enable
Firewall (security groups)
Create security group
Select existing security group
Security group name - required
CodeDeploy-SG
Description - required
launch-wizard-1 created 2022-08-23T14:15:14.298Z
Inbound security groups rules
Security group rule 1 (TCP, 22, 0.0.0.0/0)
Remove
Type
ssh
Protocol
TCP
Port range
22
Source type
Anywhere
Source
0.0.0.0/0
Description - optional
e.g. SSH for admin desktop
Security group rule 2 (TCP, 80)
Remove
Type
HTTP
Protocol
TCP
Port range
80
Summary
Number of instances
1
Software Image (AMI)
Amazon Linux 2 Kernel 5.10 AMI...read more
Virtual server type (instance type)
t2.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.
Cancel
Launch instance


Custom
Add CIDR, prefix list or security
e.g. SSH for admin desktop
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.
Add security group rule
Configure storage
Advanced
1x
8
GiB
gp2
Root volume
Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage
Add new volume
0 x File systems
Edit
Advanced details
Purchasing option
Request Spot Instances
Request Spot Instances at the Spot price, capped at the On-Demand price
Domain join directory
Select
Create new directory
IAM instance profile
Summary
Number of instances
1
Software Image (AMI)
Amazon Linux 2 Kernel 5.10 AMI...read more
Virtual server type (instance type)
t2.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.
Cancel
Launch instance


IAM instance profile Info
EC2CodeDeploy
Create new IAM profile
Hostname type Info
DNS Hostname Info
Instance auto-recovery Info
Shutdown behavior Info
Stop - Hibernate behavior Info
Termination protection Info
Stop protection Info
Detailed CloudWatch monitoring Info
Elastic GPU Info
Elastic inference Info
Summary
Number of instances Info
Software Image (AMI)
Amazon Linux 2 Kernel 5.10 AMI...read more
Virtual server type (instance type)
t2.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.
Cancel
Launch instance

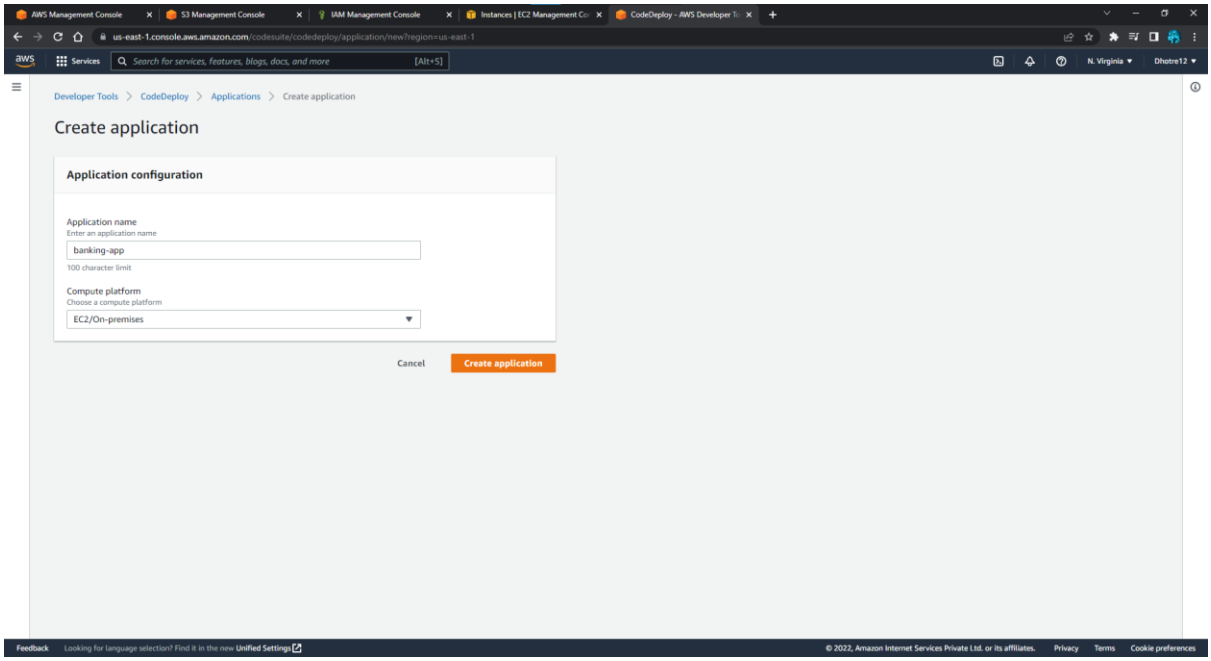

Credit specification Info
Placement group name Info
Create new placement group
EBS-optimized instance Info
Capacity reservation Info
Tenancy Info
RAM disk ID Info
Kernel ID Info
Nitro Enclave Info
Nitro Enclaves are not compatible with instance types that have less than 4 vCPUs.
License configurations Info
Specify CPU options
Metadata accessible Info
Metadata version Info
Summary
Number of instances Info
Software Image (AMI)
Amazon Linux 2 Kernel 5.10 AMI...read more
Virtual server type (instance type)
t2.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.
Cancel
Launch instance

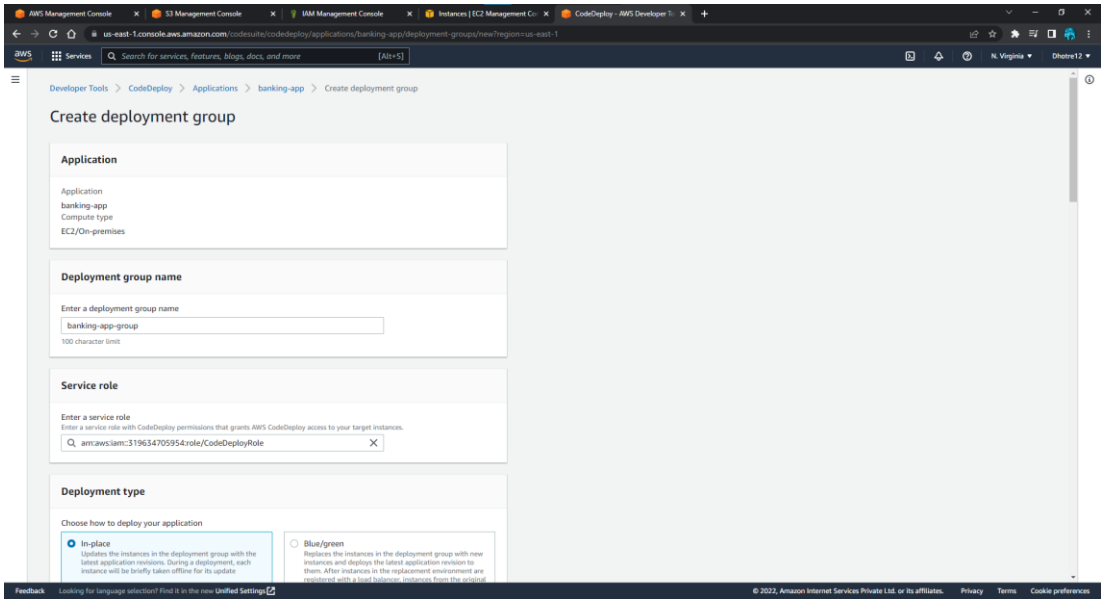

License configurations Info
Specify CPU options
Metadata accessible Info
Metadata version Info
Metadata response hop limit Info
Allow tags in metadata Info
User data Info
#!/bin/bash
sudo yum -y update
sudo yum -y install ruby
sudo yum -y install wget
cd /home/ec2-user
wget https://aws-codedeploy-ap-south-1.s3.ap-south-1.amazonaws.com/latest/install
sudo chmod +x ./install
sudo ./install auto
sudo yum install -y python-pip
sudo pip install awscli
User data has already been base64 encoded
Summary
Number of instances Info
Software Image (AMI)
Amazon Linux 2 Kernel 5.10 AMI...read more
Virtual server type (instance type)
t2.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.
Cancel
Launch instance

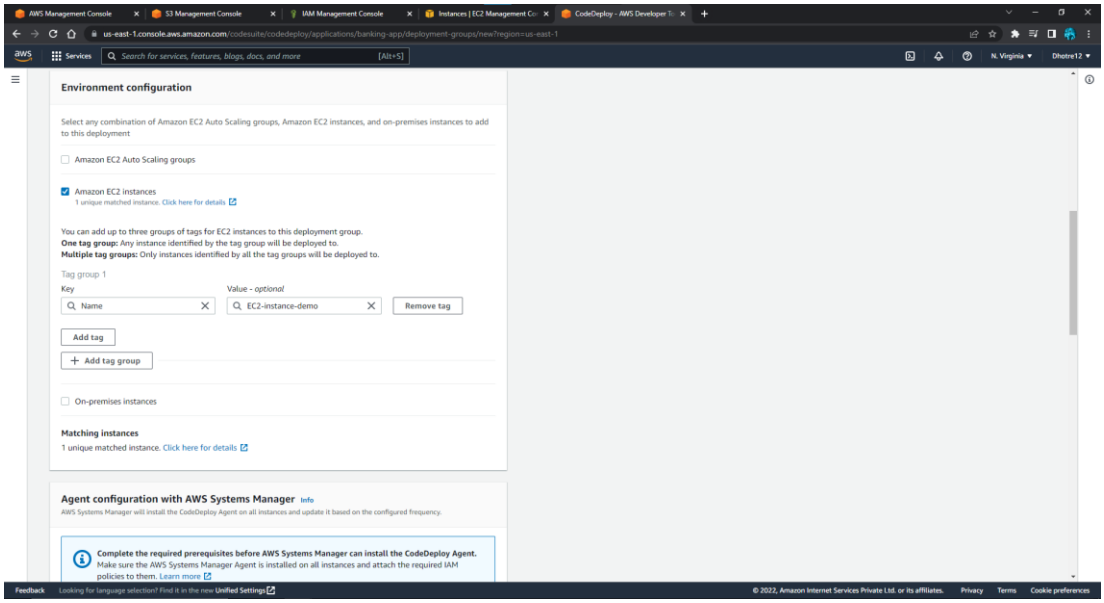


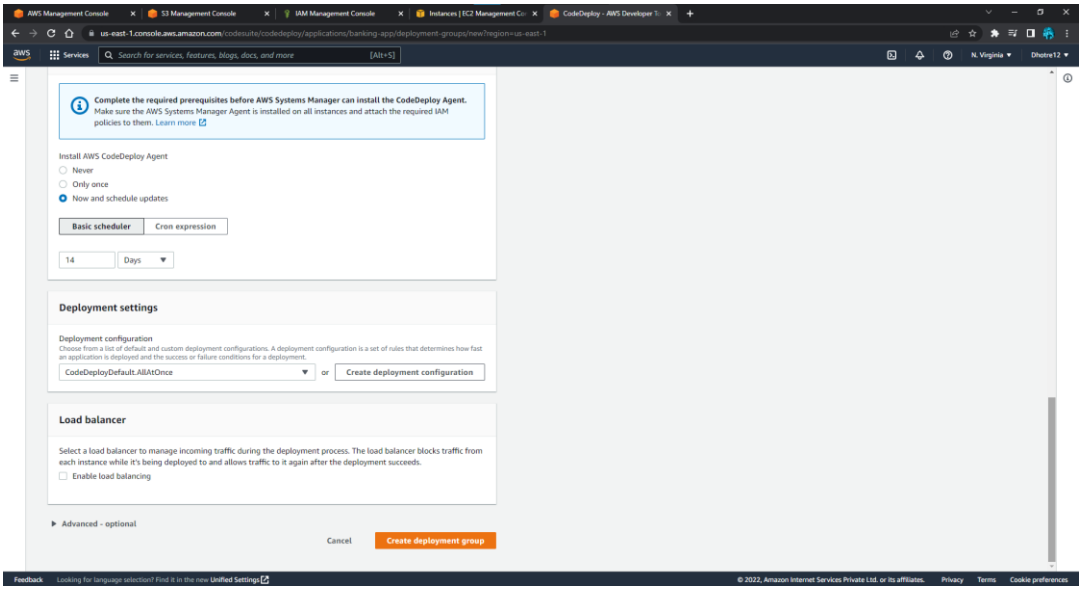
4. Setup S3 as source/repository

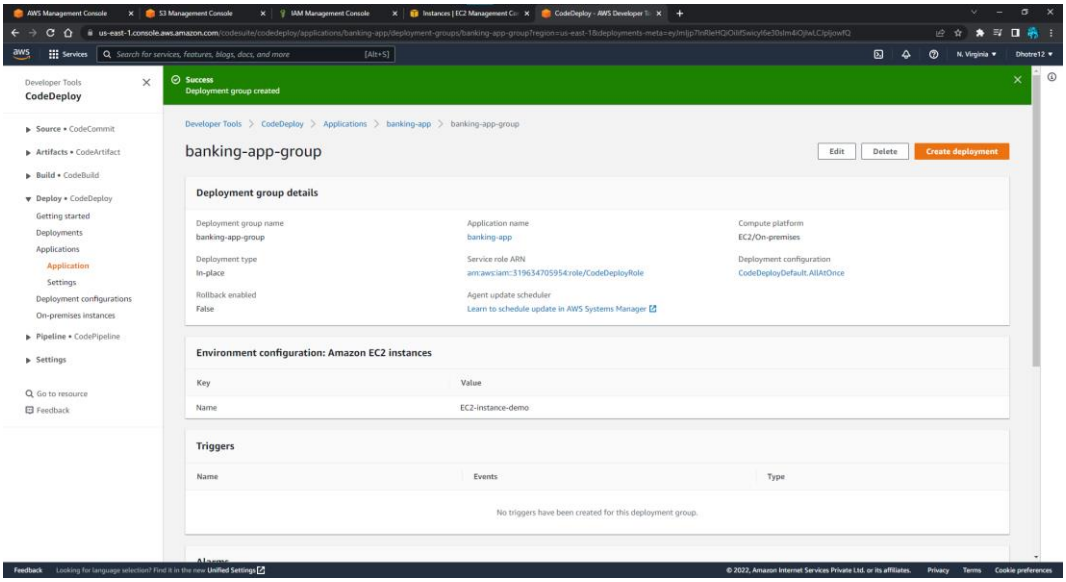
5. Setup application & Deployment Group



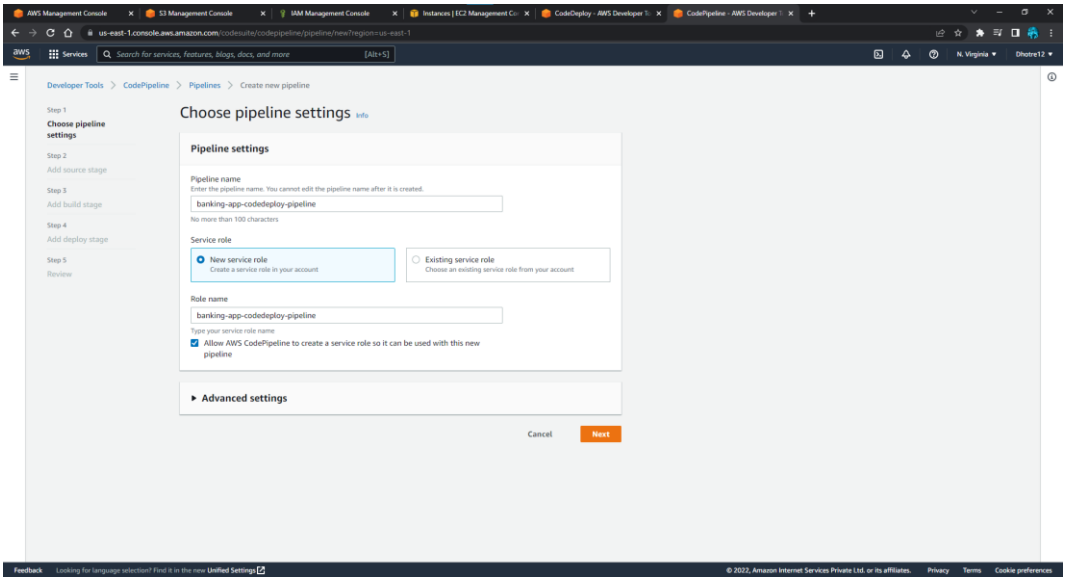

Developer Tools > CodeDeploy > Applications > banking-app > Create deployment group
Create deployment group
Application
Application
banking-app
Compute type
EC2/On-premises
Deployment group name
Enter a deployment group name
banking-app-group
100 character limit
Service role
Enter a service role
Enter a service role with CodeDeploy permissions that grants AWS CodeDeploy access to your target instances.
arn:aws:iam::319634705954: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

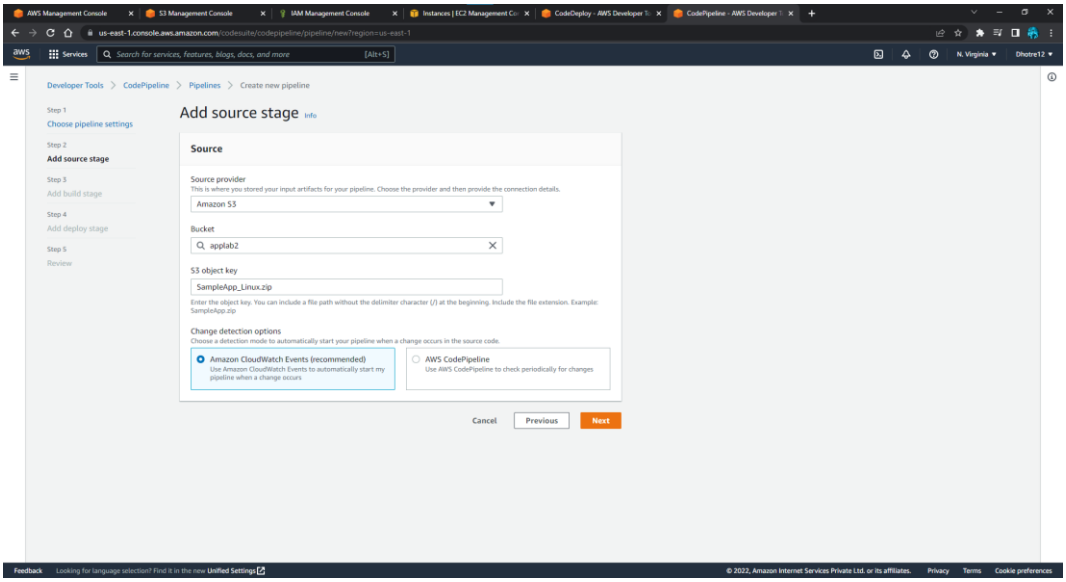

Environment configuration
Select any combination of Amazon EC2 Auto Scaling groups, Amazon EC2 instances, and on-premises instances to add to this deployment
Amazon EC2 Auto Scaling groups
Amazon EC2 instances
1 unique matched instance. Click here for details
You can add up to three groups of tags for 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
Name
EC2-instance-demo
Remove tag
Add tag
Add tag group
On-premises instances
Matching instances
1 unique matched instance. Click here for details
Agent configuration with AWS Systems Manager Info
AWS Systems Manager will install the CodeDeploy Agent on all instances and update it based on the configured frequency.
Complete the required prerequisites before AWS Systems Manager can install the CodeDeploy Agent.
Make sure the AWS Systems Manager Agent is installed on all instances and attach the required IAM policies to them. Learn more

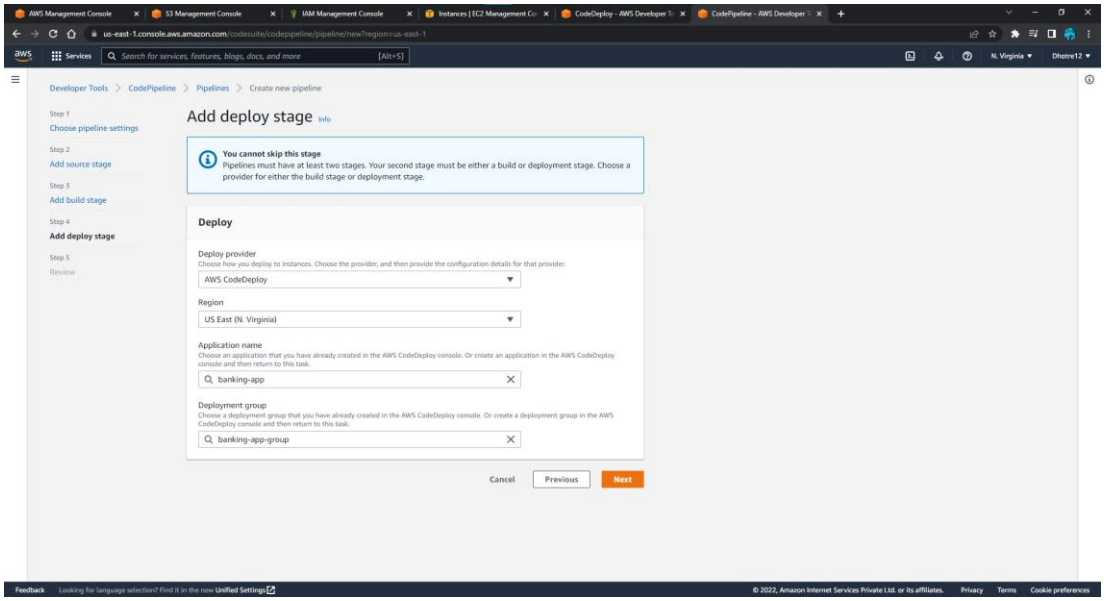

Complete the required prerequisites before AWS Systems Manager can install the CodeDeploy Agent.
Make sure the AWS Systems Manager Agent is installed on all instances and attach the required IAM policies to them. Learn more
Install AWS CodeDeploy Agent
Never
Only once
Now and schedule updates
Basic scheduler
Cron expression
14
Days
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
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

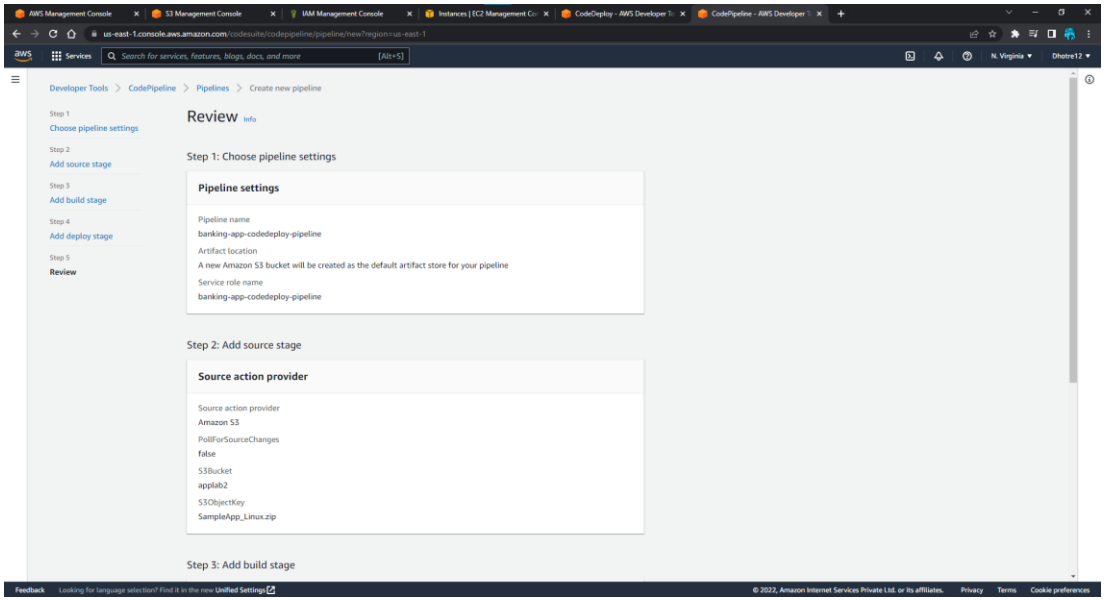


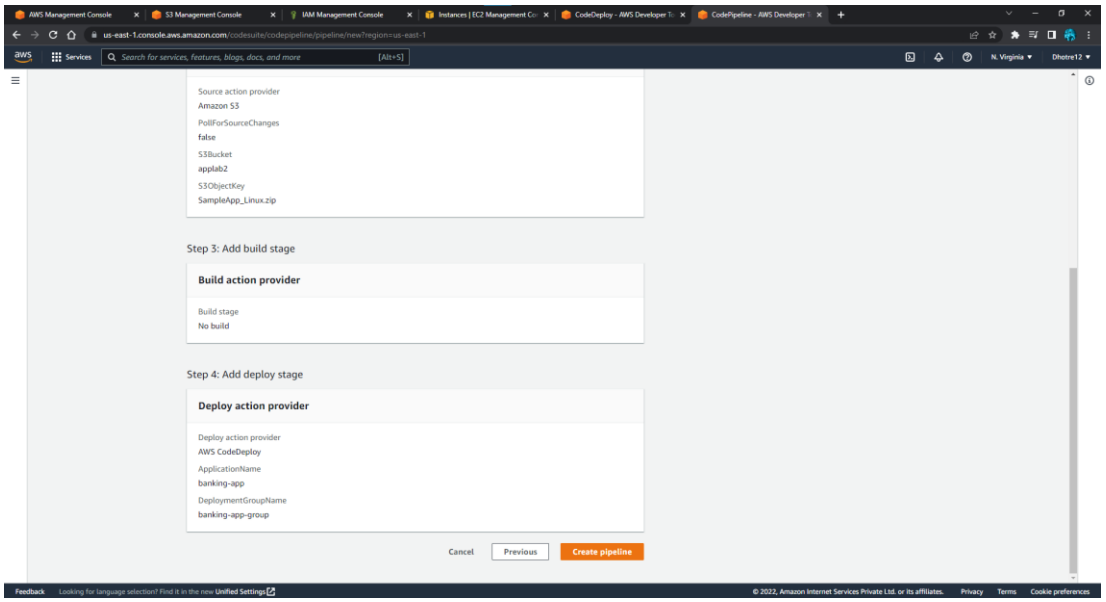
6. Setup Code Deploy pipeline

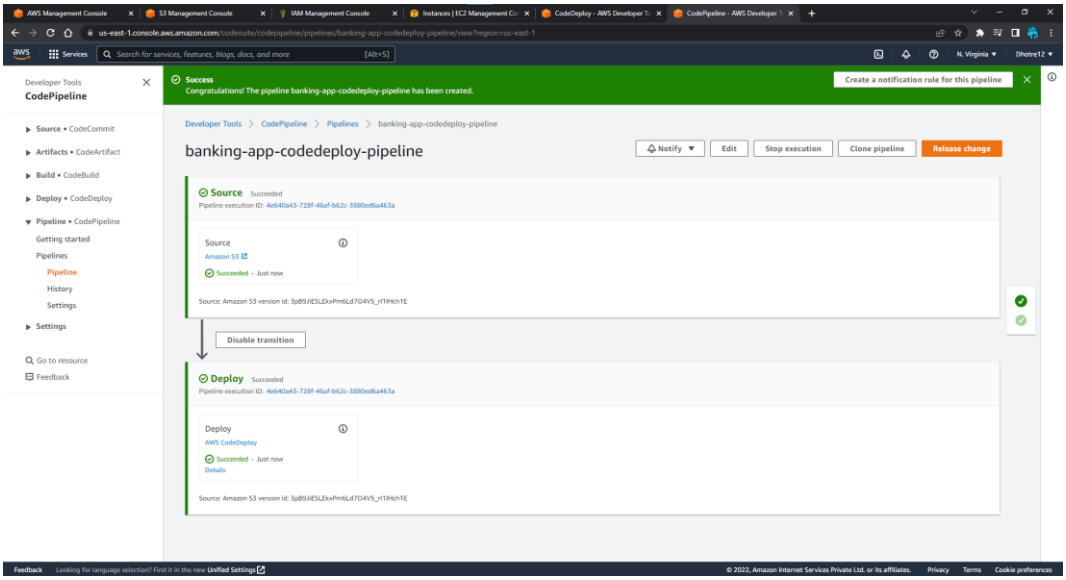




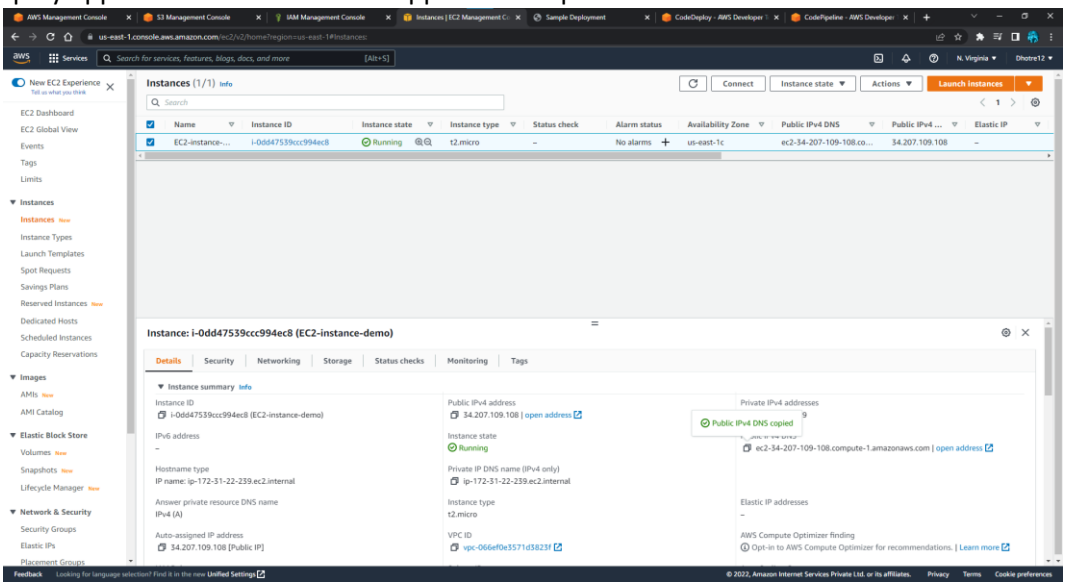

Developer Tools > CodePipeline > Pipelines > Create new pipeline
Step 1
Choose pipeline settings
Step 2
Add source stage
Step 3
Add build stage
Step 4
Add deploy stage
Step 5
Review
Add deploy stage Info
You cannot skip this stage
Pipelines must have at least two stages. Your second stage must be either a build or deployment stage. Choose a provider for either the build stage or deployment stage.
Deploy
Deploy provider
Choose how you deploy to instances. Choose the provider, and then provide the configuration details for that provider.
AWS CodeDeploy
Region
US East (N. Virginia)
Application name
Choose an application that you have already created in the AWS CodeDeploy console. Or create an application in the AWS CodeDeploy console and then return to this task.
banking-app
Deployment group
Choose a deployment group that you have already created in the AWS CodeDeploy console. Or create a deployment group in the AWS CodeDeploy console and then return to this task.
banking-app-group
Cancel
Previous
Next

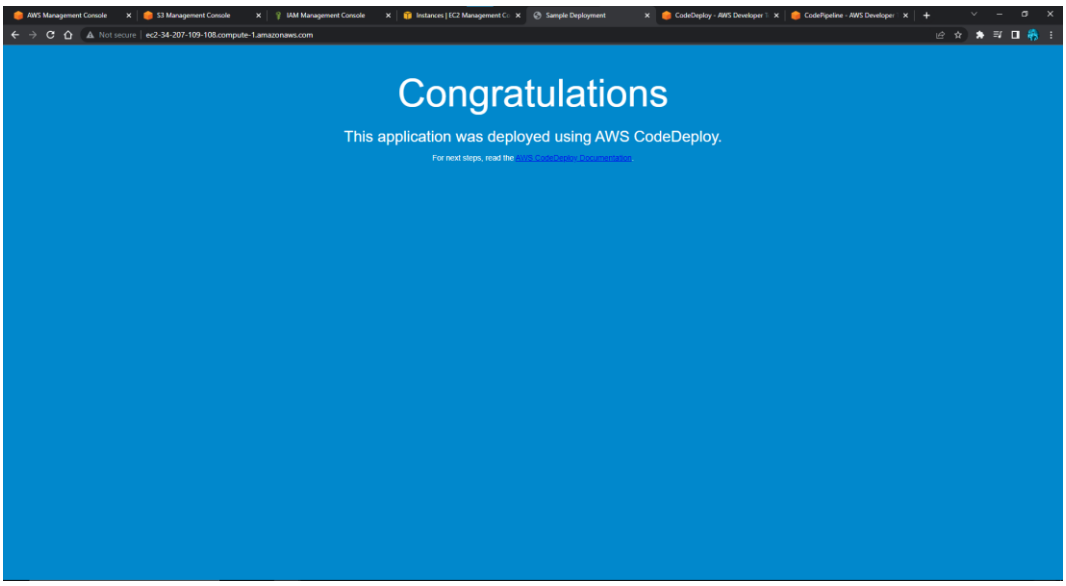

Developer Tools > CodePipeline > Pipelines > Create new pipeline
Step 1
Choose pipeline settings
Step 2
Add source stage
Step 3
Add build stage
Step 4
Add deploy stage
Step 5
Review
Review Info
Step 1: Choose pipeline settings
Pipeline settings
Pipeline name
banking-app-codedeploy-pipeline
Artifact location
A new Amazon S3 bucket will be created as the default artifact store for your pipeline
Service role name
banking-app-codedeploy-pipeline
Step 2: Add source stage
Source action provider
Source action provider
Amazon S3
PollForSourceChanges
false
S3Bucket
applab2
S3ObjectKey
SampleApp_Linux.zip
Step 3: Add build stage


Source action provider
Amazon S3
PollForSourceChanges
false
S3Bucket
applab2
S3ObjectKey
SampleApp_Linux.zip
Step 3: Add build stage
Build action provider
Build stage
No build
Step 4: Add deploy stage
Deploy action provider
Deploy action provider
AWS CodeDeploy
ApplicationName
banking-app
DeploymentGroupName
banking-app-group
Cancel
Previous
Create pipeline

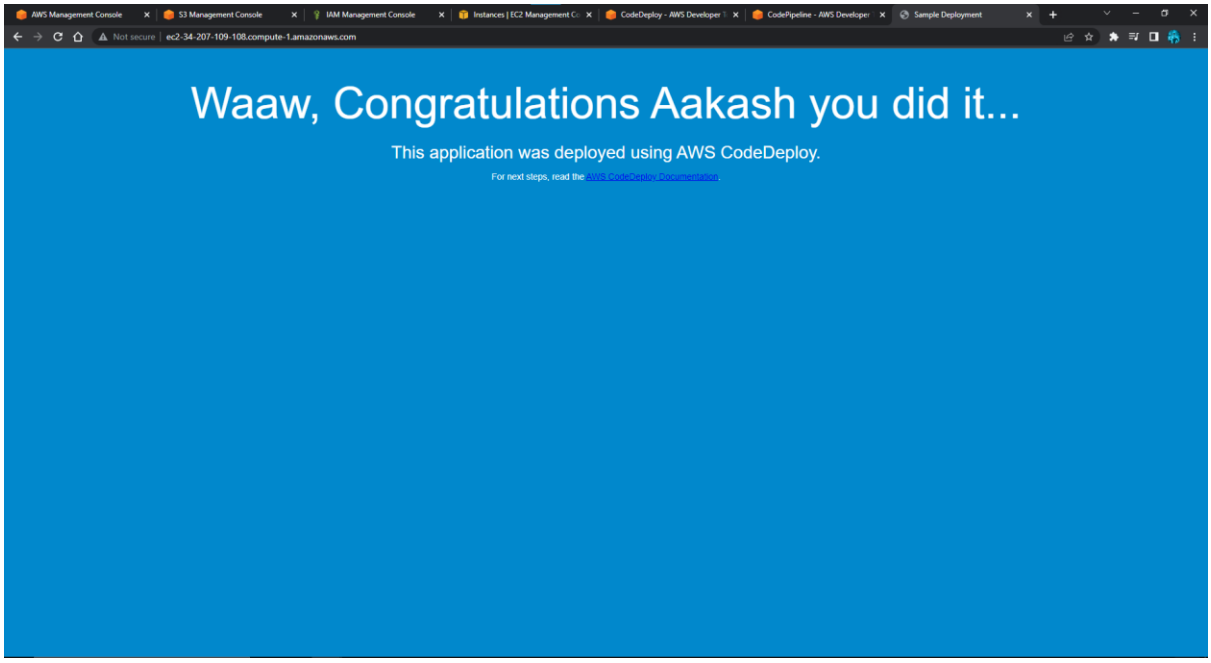


7. Deploy application on EC2 as soon application copied on S3





After changing some text to verify it is running properly or not.



Conclusion: Hence we have studied to build your application using AWS CodeBuild and Deploy on S3 / SEBS using AWS CodePipeline, deploy sample application on EC2 instance using AWS CodeDeploy and learned to implement them.

Questionnaire:

1. What is CI/CD?

Ans. **Continuous Integration (CI) and Continuous Deployment (CD**) gets rid of the traditional manual gate and implements fully automated verification of the acceptance environment to determine the scenario whether the pipeline can continue to production or not.

Continuous Integration focuses on the software development life cycle (SDLC) of the individual developer in the code repository. This can be executed multiple times with a primary goal to enable early detection of integration bugs, and errors.

Continuous Delivery focuses on automated code deployment in testing or production environment, taking the approval of updates to achieve automated software release process, pre-emptively discovering deployment issues.

2. What is CI/CD Tools Offered By AWS?

Ans. AWS offers an end-to-end CI/CD stack comprised of the following four services:

AWS CodeCommit – It is a fully-managed source control service that hosts secure Git-based repositories. CodeCommit makes it easy for teams to collaborate on code in a secure and highly scalable ecosystem.

AWS CodeBuild – A fully managed continuous integration service that compiles source code, runs tests and produces software packages that are ready to deploy, on a dynamically created build server.

AWS CodeDeploy – A fully managed deployment service that automates software deployments to a variety of computing services such as Amazon EC2, AWS Fargate, AWS Lambda, and you're on-premises servers.

AWS CodePipeline – A fully configured continuous delivery service that helps the user to automate their released pipelines for fast and reliable application and infrastructure updates.

3. What are the steps to deploy web application using AWS CodePipeline?

Ans. We will be performing 4 steps to deploy a web application

Step 1: Create an S3 bucket for your application

Step 2: Create Amazon EC2 Windows instances and install the CodeDeploy agent

Step 3: Create an application in CodeDeploy

Step 4: Create your first pipeline in CodePipeline

4. How is the build project used in AWS CodeBuild functions?

Ans. AWS CodeBuild builds your code and stores the artifacts into an Amazon S3 bucket, or you can use a build command to upload them to an artifact repository.

5. Which AWS service is used to trigger code build test and deployment?

Ans. AWS CodeBuild – A fully managed continuous integration service that compiles source code, runs tests, and produces software packages that are ready to deploy, on a dynamically created build server.

Experiment 3

Aim: To understand the Kubernetes Cluster Architecture, install and Sign Up a Kubernetes Cluster on Linux Machines/Cloud Platforms.

Theory:

Kubernetes: More than just container orchestration

As stated before (but is worth stating again), Kubernetes is an open-source platform for deploying and managing containers. It provides a container runtime, container orchestration, container-centric infrastructure orchestration, self-healing mechanisms, service discovery and load balancing. It's used for the deployment, scaling, management, and composition of application containers across clusters of hosts.

But Kubernetes is more than just a container orchestrator. It could be thought of as the operating system for cloud-native applications in the sense that it's the platform that applications run on, just as desktop applications run on MacOS, Windows, or Linux.

It aims to reduce the burden of orchestrating underlying compute, network, and storage infrastructure, and enable application operators and developers to focus entirely on container-centric workflows for self-service operation. It allows developers to build customized workflows and higher-level automation to deploy and manage applications composed of multiple containers.

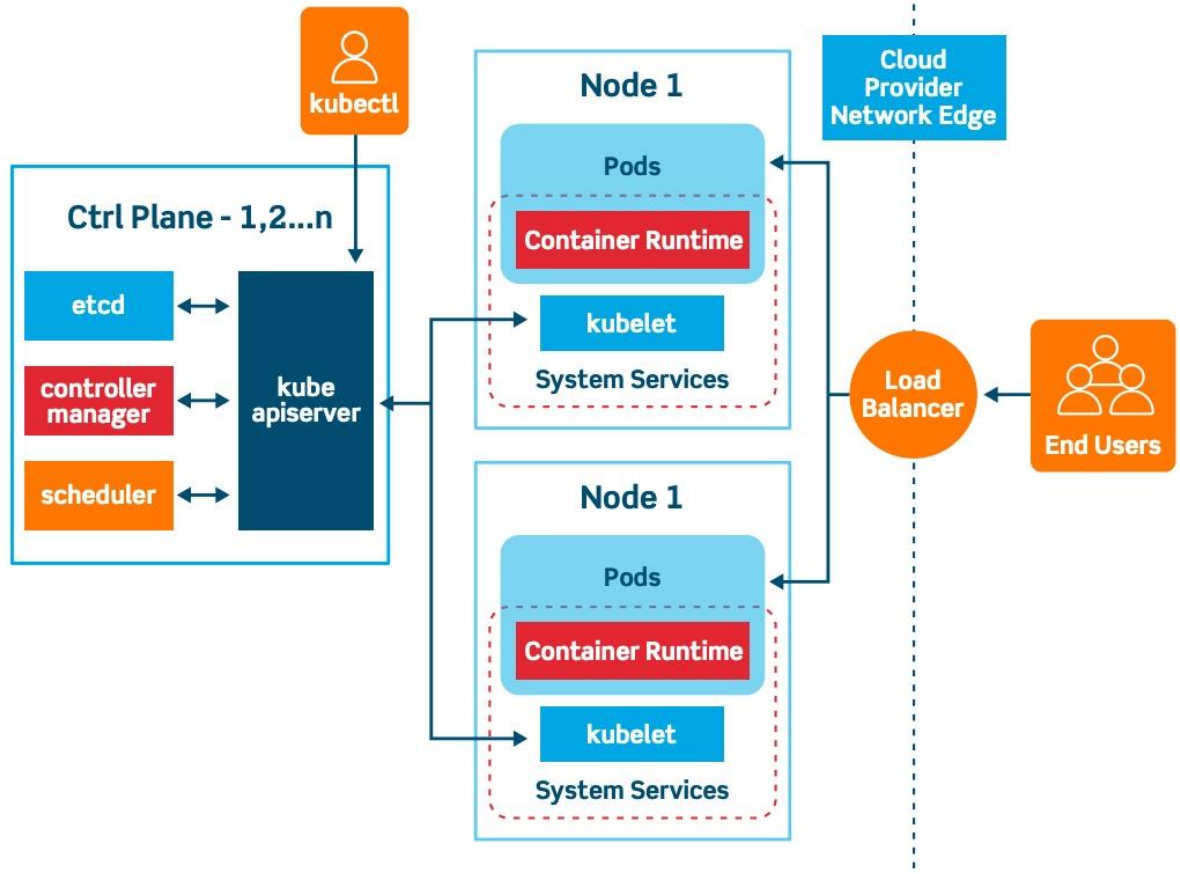
While Kubernetes runs all major categories of workloads, such as monoliths, stateless or stateful applications, microservices, services, batch jobs and everything in between, it's commonly used for the microservices category of workloads.

In the early years of the project, it mostly ran stateless applications, but as the platform has gained popularity, more and more storage integrations have been developed to natively support stateful applications.

Kubernetes is a very flexible and extensible platform. It allows you to consume its functionality a-la-carte, or use your own solution in lieu of built-in functionality. On the other hand, you can also integrate Kubernetes into your environment and add additional capabilities.

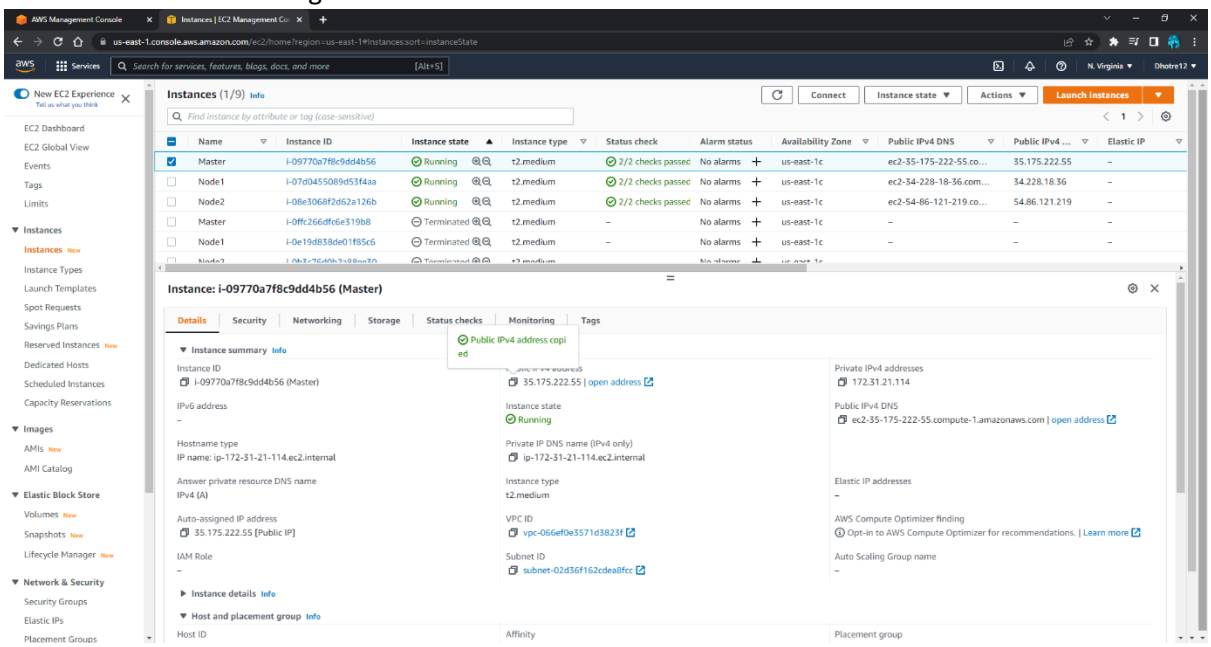
Kubernetes Architecture and Concepts

From a high level, a Kubernetes environment consists of a control plane (master), a distributed storage system for keeping the cluster state consistent (etcd), and a number of cluster nodes (Kubelets).



Result:

Create EC2 Instances using ubuntu



Conclusion: Hence we have studied to understand the Kubernetes Cluster Architecture, install and Spin Up a Kubernetes Cluster on Linux Machines/Cloud Platforms and learned to implement them.

Questionnaire:

1. What is Kubernetes?

This is one of the most basic Kubernetes interview questions yet one of the most important ones! Kubernetes is an open-source container orchestration tool or system that is used to automate tasks such as the management, monitoring, scaling, and deployment of containerized applications. It is used to easily manage several containers (since it can handle grouping of containers), which provides for logical units that can be discovered and managed.

2. What are K8s?

K8s is another term for Kubernetes.

3. How are Kubernetes and Docker related?

This is one of the most frequently asked Kubernetes interview questions, where the interviewer might as well ask you to share your experience working with any of them. Docker is an open-source platform used to handle software development. Its main benefit is that it packages the settings and dependencies that the software/application needs to run into a container, which allows for portability and several other advantages. Kubernetes allows for the manual linking and orchestration of several containers, running on multiple hosts that have been created using Docker.

4. What are the features of Kubernetes?

- Kubernetes places control for the user where the server will host the container. It will control how to launch. So, Kubernetes automates various manual processes.
- Kubernetes manages various clusters at the same time.
- It provides various additional services like management of containers, security, networking, and storage.
- Kubernetes self-monitors the health of nodes and containers.
- With Kubernetes, users can scale resources not only vertically but also horizontally that too easily and quickly.

5. What are the main components of Kubernetes architecture?

There are two primary components of Kubernetes Architecture: the master node and the worker node. Each of these components has individual components in them.

Experiment 4

Aim: To install Kubectl and execute Kubectl commands to manage the Kubernetes cluster and deploy Your First Kubernetes Application.

Theory:

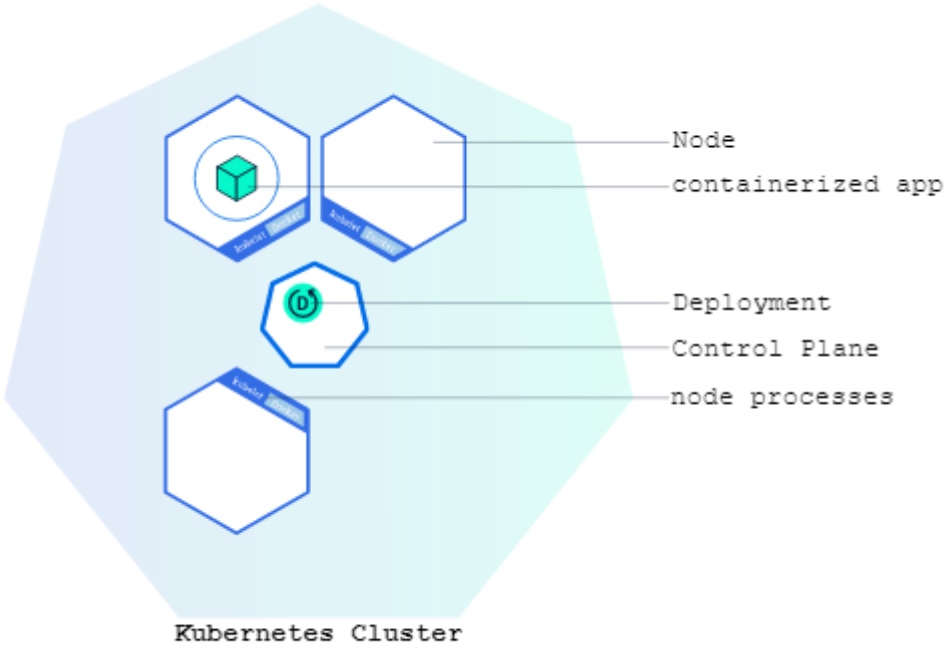
Kubernetes Deployments

Once you have a running Kubernetes cluster, you can deploy your containerized applications on top of it. To do so, you create a Kubernetes Deployment configuration. The Deployment instructs Kubernetes how to create and update instances of your application. Once you've created a Deployment, the Kubernetes control plane schedules the application instances included in that Deployment to run on individual Nodes in the cluster.

Once the application instances are created, a Kubernetes Deployment Controller continuously monitors those instances. If the Node hosting an instance goes down or is deleted, the Deployment controller replaces the instance with an instance on another Node in the cluster. This provides a self-healing mechanism to address machine failure or maintenance.

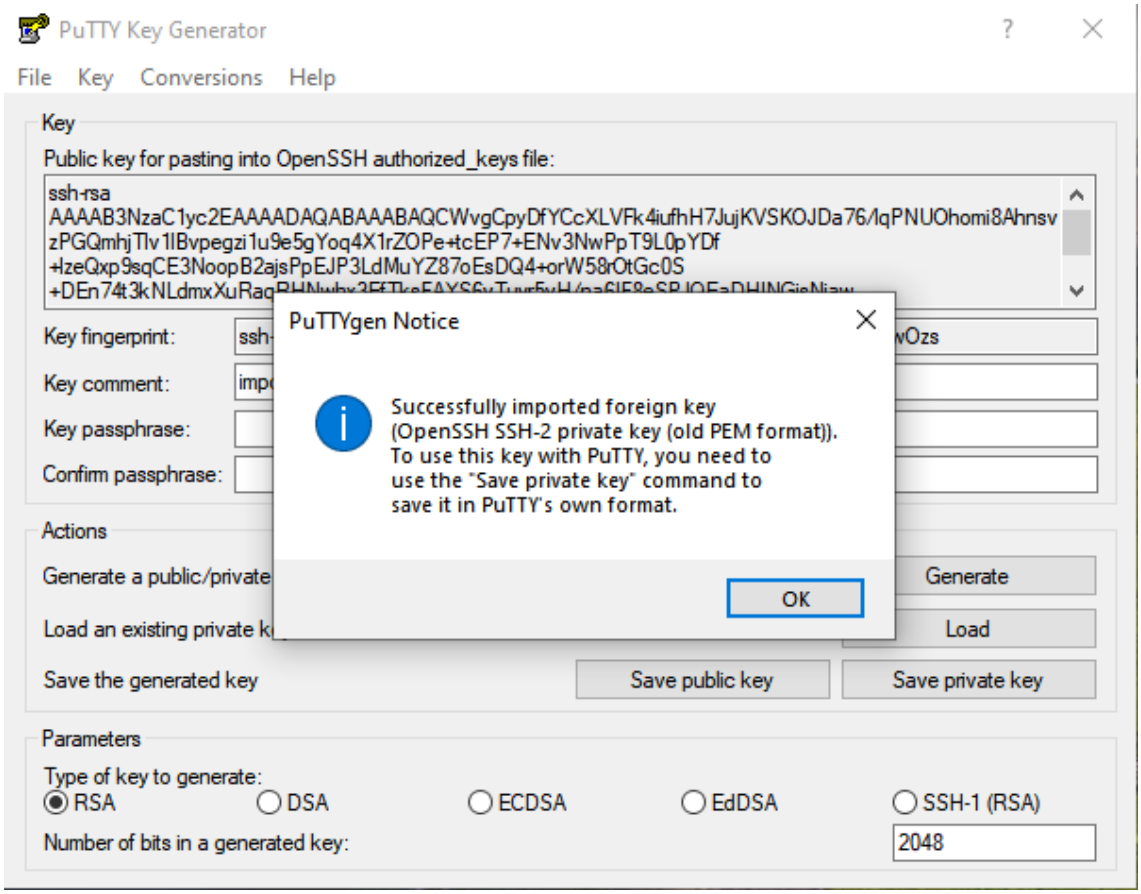
In a pre-orchestration world, installation scripts would often be used to start applications, but they did not allow recovery from machine failure. By both creating your application instances and keeping them running across Nodes, Kubernetes Deployments provide a fundamentally different approach to application management.

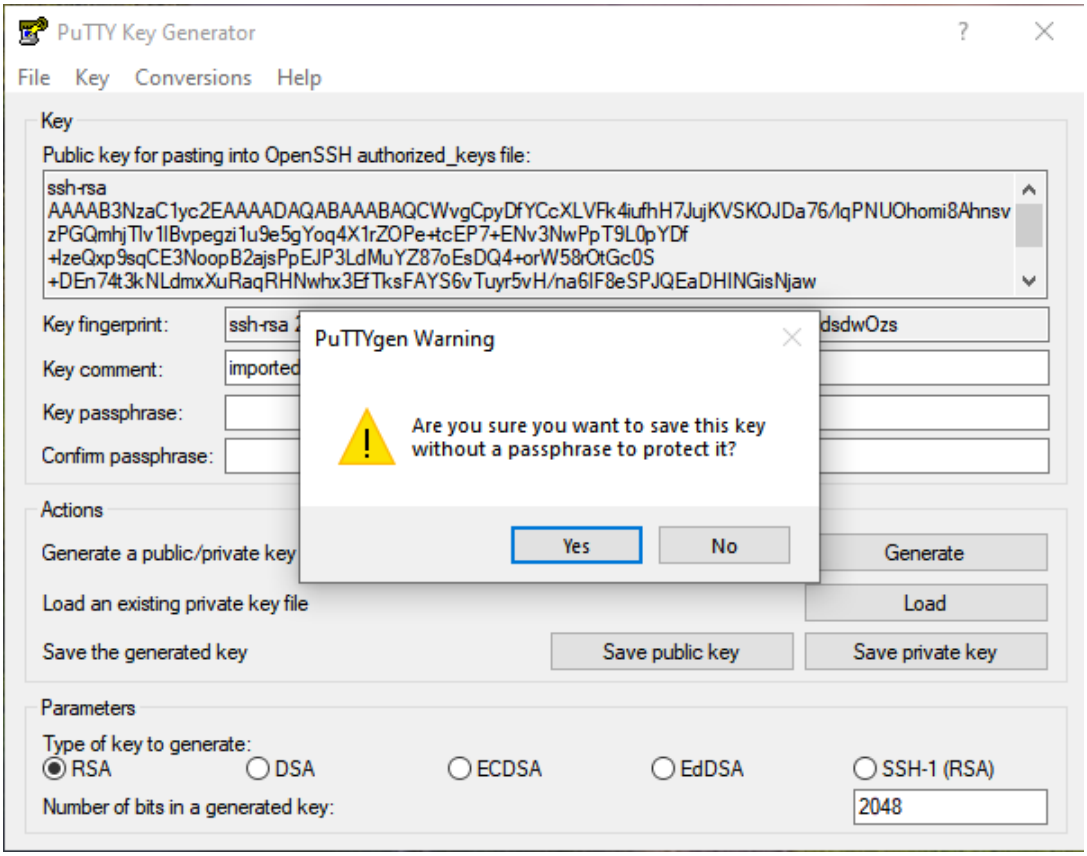
Deploying your first app on Kubernetes



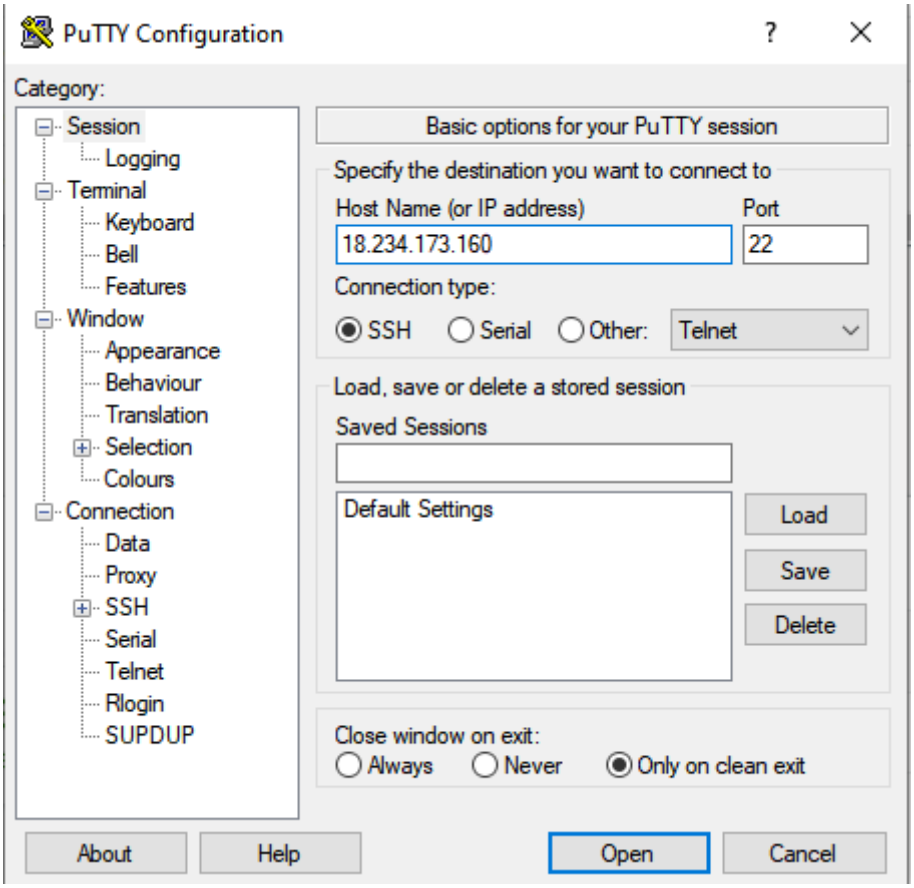
Result:

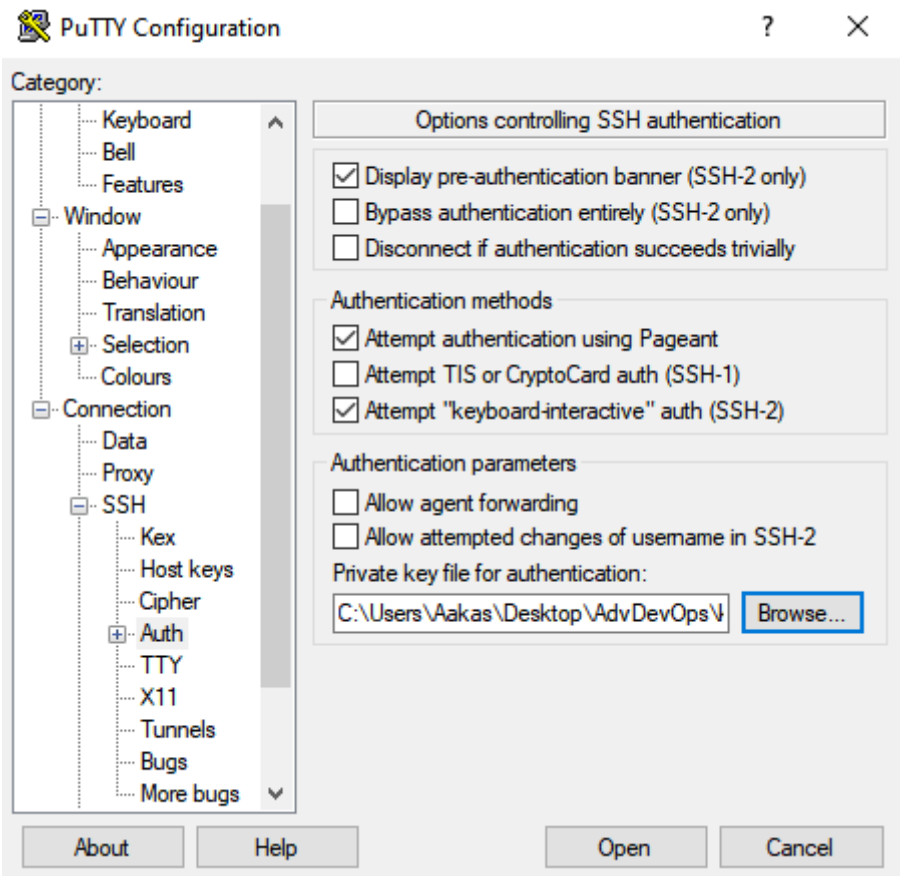
1. Open Puttygen



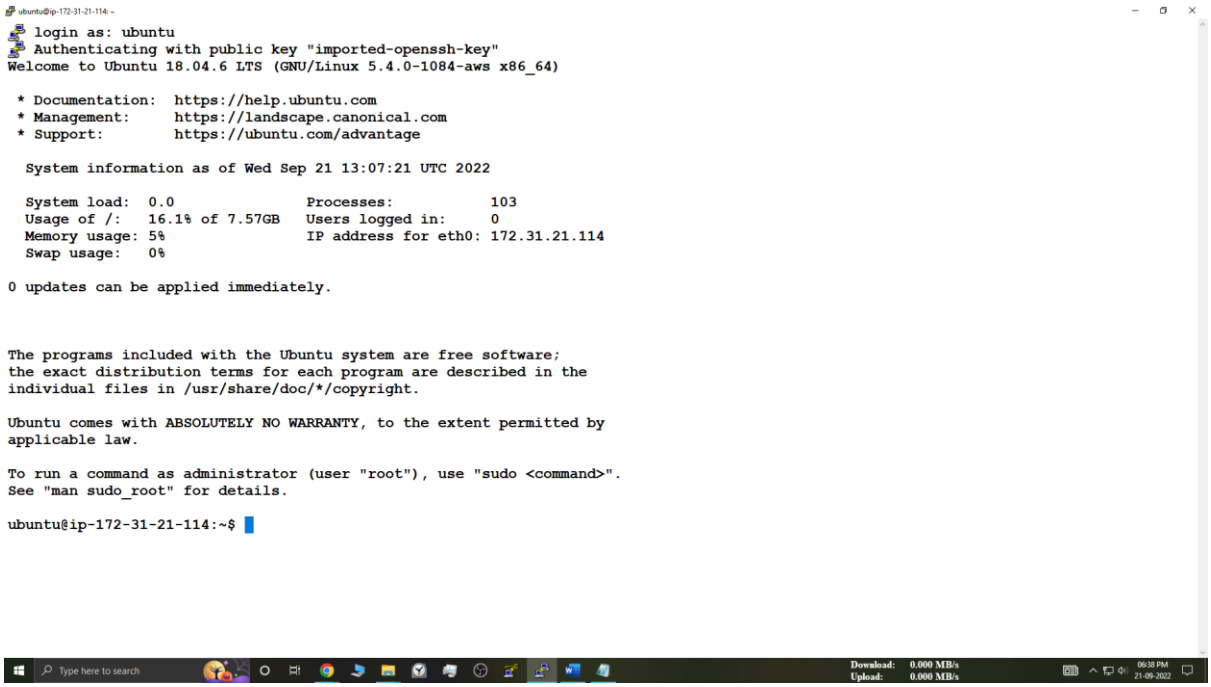


2. Open Putty

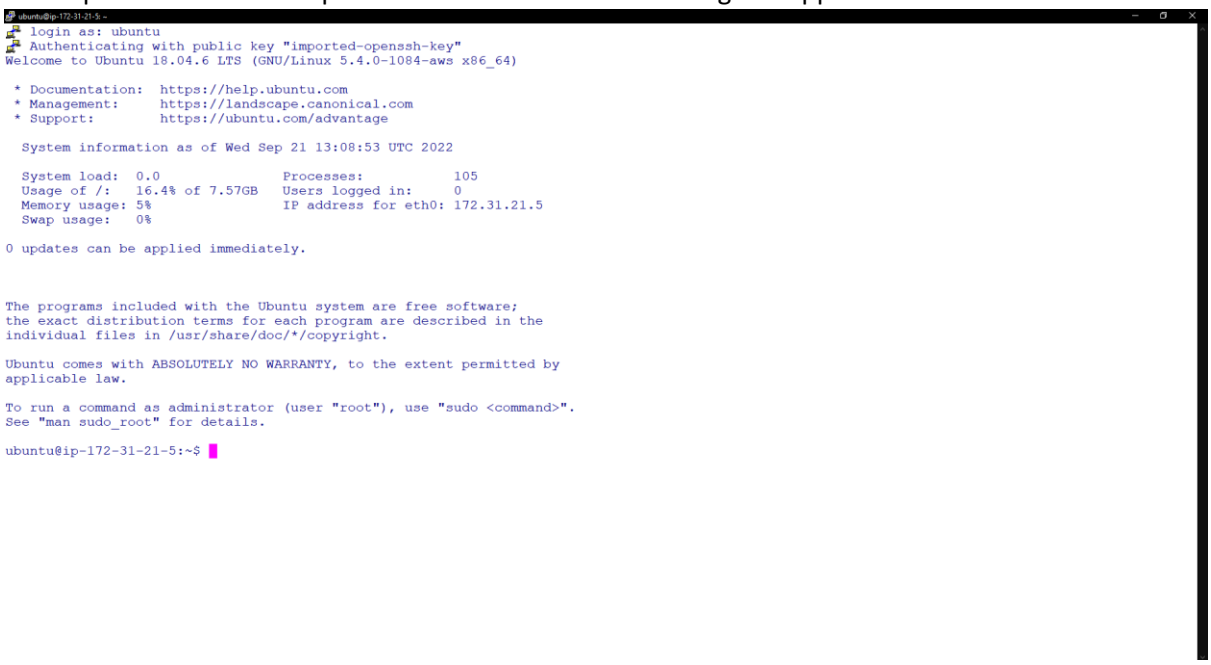

PuTTY Configuration
Category:
Session
Logging
Terminal
Keyboard
Bell
Features
Window
Appearance
Behaviour
Translation
Selection
Colours
Connection
Data
Proxy
SSH
Serial
Telnet
Rlogin
SUPDUP
Basic options for your PuTTY session
Specify the destination you want to connect to
Host Name (or IP address)
Port
18.234.173.160
22
Connection type:
SSH
Serial
Other:
Telnet
Load, save or delete a stored session
Saved Sessions
Default Settings
Load
Save
Delete
Close window on exit:
Always
Never
Only on clean exit
About
Help
Open
Cancel

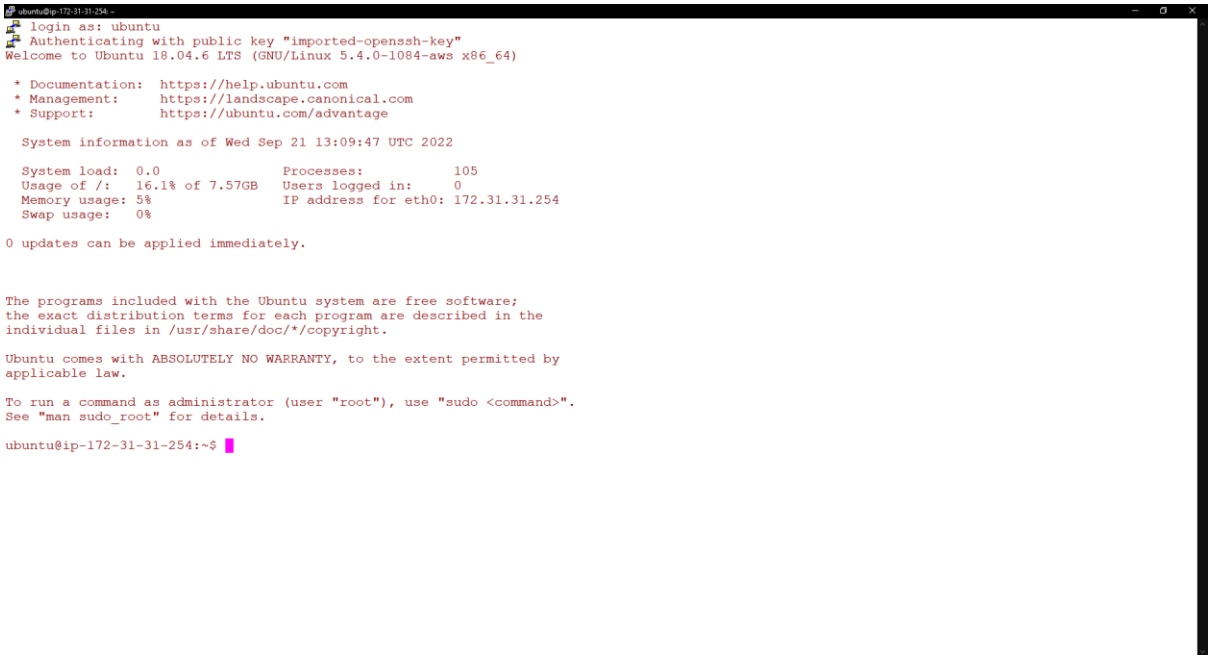

PuTTY Configuration
Category:
Keyboard
Bell
Features
Window
Appearance
Behaviour
Translation
Selection
Colours
Connection
Data
Proxy
SSH
Kex
Host keys
Cipher
Auth
TTY
X11
Tunnels
Bugs
More bugs
Options controlling SSH authentication
Display pre-authentication banner (SSH-2 only)
Bypass authentication entirely (SSH-2 only)
Disconnect if authentication succeeds trivially
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:
C:\Users\Aakas\Desktop\AdvDevOps\
Browse...
About
Help
Open
Cancel

3. Type login as: ubuntu



4. Repeat the 3rd – 5th step for Node1 and Node2 and change its appearance





5. Execute the following code in all the 3 terminals:

sudo su
apt-get update
apt-get install apt-transport-https

apt install docker.io -y
docker --version
systemctl start docker
systemctl enable docker

sudo curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key add

nano /etc/apt/sources.list.d/kubernetes.list

deb http://apt.kubernetes.io/ kubernetes-xenial main

apt-get update

apt-get install -y kubelet kubeadm kubectl kubernetes-cni

6. BOOTSTRAPPING THE MASTER NODE (IN MASTER)

kubeadm init

7. COPY THE COMMAND TO RUN IN NODES & SAVE IN NOTEPAD

mkdir -p $HOME/.kube
cp -i /etc/kubernetes/admin.conf $HOME/.kube/config

chown $(id -u):$(id -g) $HOME/.kube/config

kubectl apply -f https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.yml
kubectl apply -f https://raw.githubusercontent.com/coreos/flannel/master/Documentation/k8s-manifests/kube-flannel-rbac.yml

8. CONFIGURE WORKER NODES (IN NODES)

COPY LONG CODE PROVIDED MY MASTER IN NODE NOW LIKE CODE GIVEN BELOW
e.g-kubeadm join 172.31.21.114:6443 --token qmrfl0.rfwlwbw02c7z6gw2 --discovery-token-ca-cert-hash sha256:68afaa488df0dfab532d5bc6c1dc3d7a678e8bcb303bad6e70907e921b9fc36f

9. GO TO MASTER AND RUN THIS COMMAND

kubectl get nodes

```
Then you can join any number of worker nodes by running the following on each as root:

kubeadm join 172.31.21.114:6443 --token qmrfl0.rfwlwbw02c7z6gw2 \
        --discovery-token-ca-cert-hash sha256:68afaa488df0dfab532d5bc6c1dc3d7a678e8bcb303bad6e70907e921b9fc36f
root@ip-172-31-21-114:/home/ubuntu# mkdir -p $HOME/.kube
root@ip-172-31-21-114:/home/ubuntu# sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
root@ip-172-31-21-114:/home/ubuntu# sudo chown $(id -u):$(id -g) $HOME/.kube/config
root@ip-172-31-21-114:/home/ubuntu# kubectl apply -f https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel
.yml
namespace/kube-flannel created
clusterrole.rbac.authorization.k8s.io/flannel created
clusterrolebinding.rbac.authorization.k8s.io/flannel created
serviceaccount/flannel created
configmap/kube-flannel-cfg created
daemonset.apps/kube-flannel-ds created
root@ip-172-31-21-114:/home/ubuntu# kubectl apply -f https://raw.githubusercontent.com/coreos/flannel/master/Documentation/k8s-manifest
s/kube-flannel-rbac.yml
clusterrole.rbac.authorization.k8s.io/flannel unchanged
clusterrolebinding.rbac.authorization.k8s.io/flannel configured
root@ip-172-31-21-114:/home/ubuntu# kubectl get nodes
NAME               STATUS   ROLES           AGE     VERSION
ip-172-31-21-114   Ready    control-plane   4m34s   v1.25.1
ip-172-31-21-5     Ready    <none>          99s     v1.25.1
ip-172-31-31-254   Ready    <none>          60s     v1.25.1
root@ip-172-31-21-114:/home/ubuntu#
```

Conclusion: Hence we have studied to understand the Kubernetes Cluster Architecture, install and Spin Up a Kubernetes Cluster on Linux Machines/Cloud Platforms and learned to implement them.

Questionnaire:

1. Explain the working of the master node in Kubernetes?

Ans. The master node dignifies the node that controls and manages the set of worker nodes. This kind resembles a cluster in Kubernetes. The nodes are responsible for the cluster management and the API used to configure and manage the resources within the collection. The master nodes of Kubernetes can run with Kubernetes itself, the asset of dedicated pods.

2. What is a node in Kubernetes?

Ans. A node is the smallest fundamental unit of computing hardware. It represents a single machine in a cluster, which could be a physical machine in a data center or a virtual machine from a cloud provider. Each machine can substitute any other machine in a Kubernetes cluster. The master in Kubernetes controls the nodes that have containers.

3. What does the node status contain?

Ans. The main components of a node status are Address, Condition, Capacity, and Info.

4. What process runs on Kubernetes Master Node?

Ans. The Kube-api server process runs on the master node and serves to scale the deployment of more instances.

5. What is the job of the kube-scheduler?

Ans. The kube-scheduler assigns nodes to newly created pods.

6. What is a cluster of containers in Kubernetes?

Ans. A cluster of containers is a set of machine elements that are nodes. Clusters initiate specific routes so that the containers running on the nodes can communicate with each other. In Kubernetes, the container engine (not the server of the Kubernetes API) provides hosting for the API server.

7. What is the Google Container Engine?

Ans. The Google Container Engine is an open-source management platform tailor-made for Docker containers and clusters to provide support for the clusters that run in Google public cloud services.

8. What are Daemon sets?

Ans. A Daemon set is a set of pods that runs only once on a host. They are used for host layer attributes like a network or for monitoring a network, which you may not need to run on a host more than once.

9. What is 'Heapster' in Kubernetes?

Ans. In this Kubernetes interview question, the interviewer would expect a thorough explanation. You can explain what it is and also it has been useful to you (if you have used it in your work so far!). A Heapster is a performance monitoring and metrics collection system for data collected by the Kublet. This aggregator is natively supported and runs like any other pod within a Kubernetes cluster, which allows it to discover and query usage data from all nodes within the cluster.

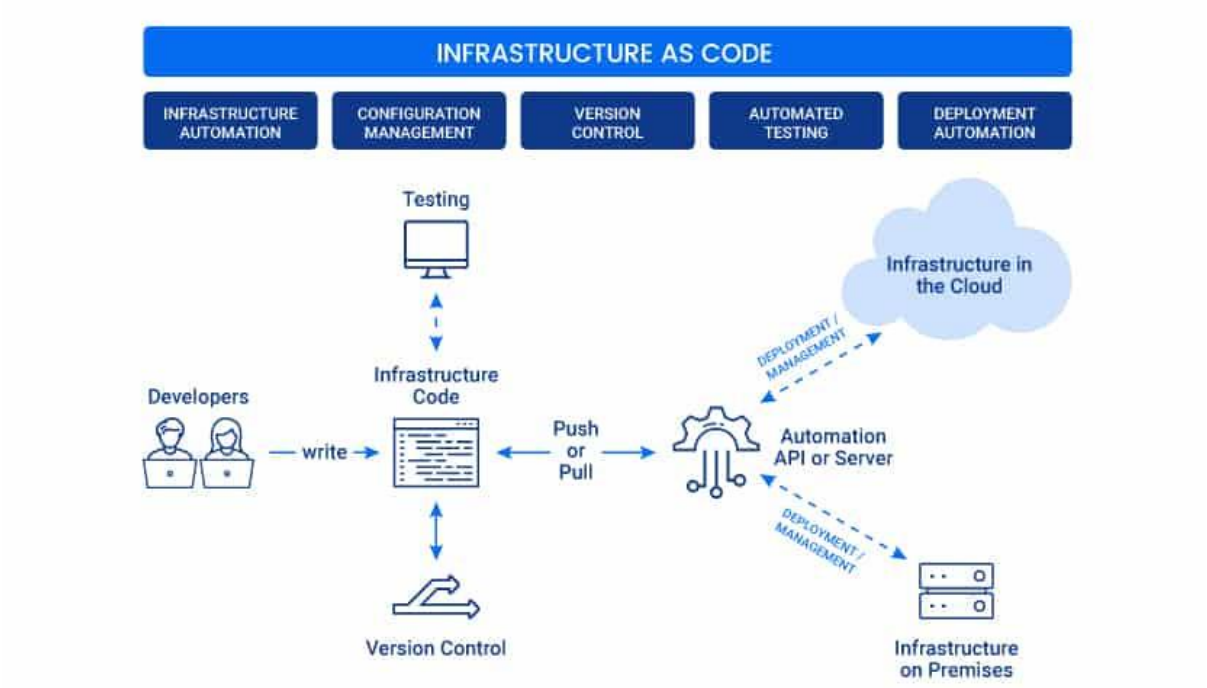
Experiment 5

Aim: To understand terraform lifecycle, core concepts/terminologies and install it on a Linux Machine.

Theory:

What Is Infrastructure as Code (IaC)?

Infrastructure as Code (IaC) is a widespread terminology among DevOps professionals and a key DevOps practice in the industry. It is the process of managing and provisioning the complete IT infrastructure (comprises both physical and virtual machines) using machine-readable definition files. It helps in automating the complete data center by using programming scripts.



Popular IaC Tools:

1. ***Terraform***: An open-source declarative tool that offers pre-written modules to build and manage an infrastructure.
2. ***Chef***: A configuration management tool that uses cookbooks and recipes to deploy the desired environment. Best used for Deploying and configuring applications using a pull-based approach.
3. ***Puppet***: Popular tool for configuration management that follows a Client-Server Model. Puppet needs agents to be deployed on the target machines before the puppet can start managing them.
4. ***Ansible***: Ansible is used for building infrastructure as well as deploying and configuring applications on top of them. Best used for Ad hoc analysis.
5. ***Packer***: Unique tool that generates VM images (not running VMs) based on steps you provide. Best used for Baking compute images.
6. ***Vagrant***: Builds VMs using a workflow. Best used for Creating pre-configured developer VMs within VirtualBox.

What Is Terraform?

Terraform is one of the most popular ***Infrastructure-as-code (IaC) tool***, used by DevOps teams to automate infrastructure tasks. It is used to automate the provisioning of your cloud resources.

Terraform is an open-source, cloud-agnostic provisioning tool developed by HashiCorp and written in GO language.

Terraform Lifecycle

Terraform lifecycle consists of – ***init, plan, apply,*** and ***destroy.***

1. ***Terraform init*** initializes the (local) Terraform environment. Usually executed only once per session.
2. ***Terraform plan*** compares the Terraform state with the as-is state in the cloud, build and display an execution plan. This does not change the deployment (read-only).
3. ***Terraform apply*** executes the plan. This potentially changes the deployment.
4. ***Terraform destroy*** deletes all resources that are governed by this specific terraform environment.

Terraform Core Concepts

1. ***Variables***: Terraform has input and output variables, it is a key-value pair. Input variables are used as parameters to input values at run time to customize our deployments. Output variables are return values of a terraform module that can be used by other configurations.

2. ***Provider***: Terraform users provision their infrastructure on the major cloud providers such as AWS, Azure, OCI, and others. A provider is a plugin that interacts with the various APIs required to create, update, and delete various resources.

3. ***Module***: Any set of Terraform configuration files in a folder is a module. Every Terraform configuration has at least one module, known as its ***root module.***

4. ***State***: Terraform records information about what infrastructure is created in a Terraform state file. With the state file, terraform is able to find the resources it created previously, supposed to manage and update them accordingly.

5. ***Resources***: Cloud Providers provides various services in their offerings; they are referenced as Resources in Terraform. Terraform resources can be anything from compute instances, virtual networks to higher-level components such as DNS records. Each resource has its own attributes to define that resource.

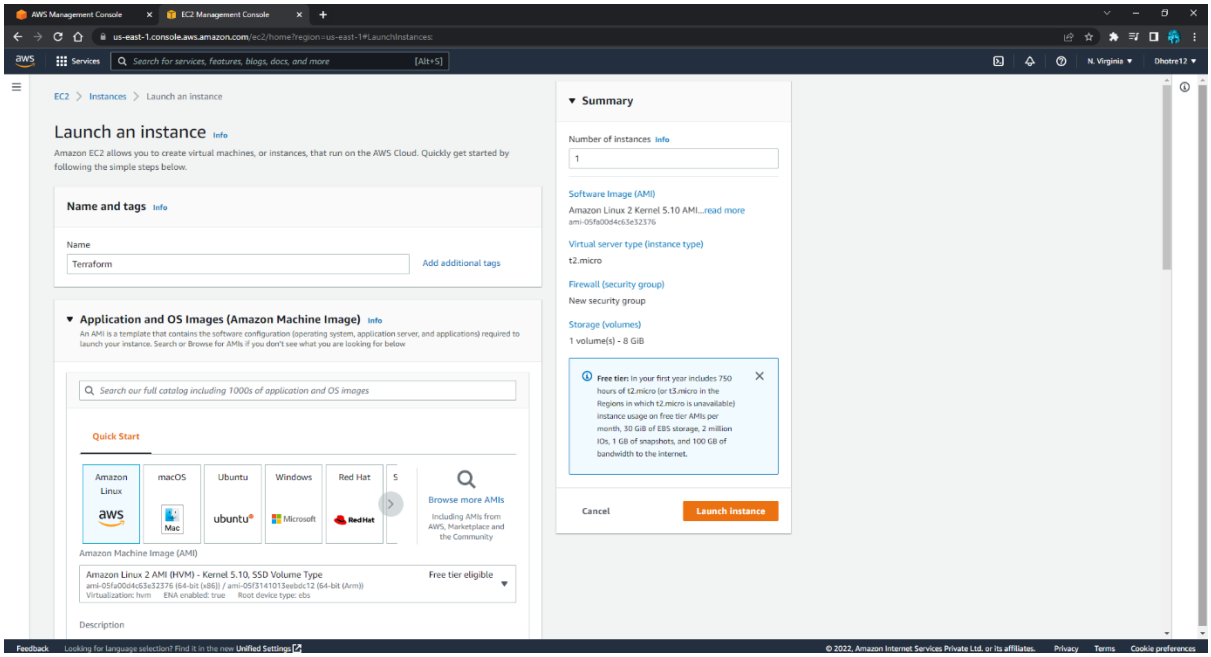
6. ***Data Source***: Data source performs a read-only operation. It allows data to be fetched or computed from resources/entities that are not defined or managed by Terraform or the current Terraform configuration.

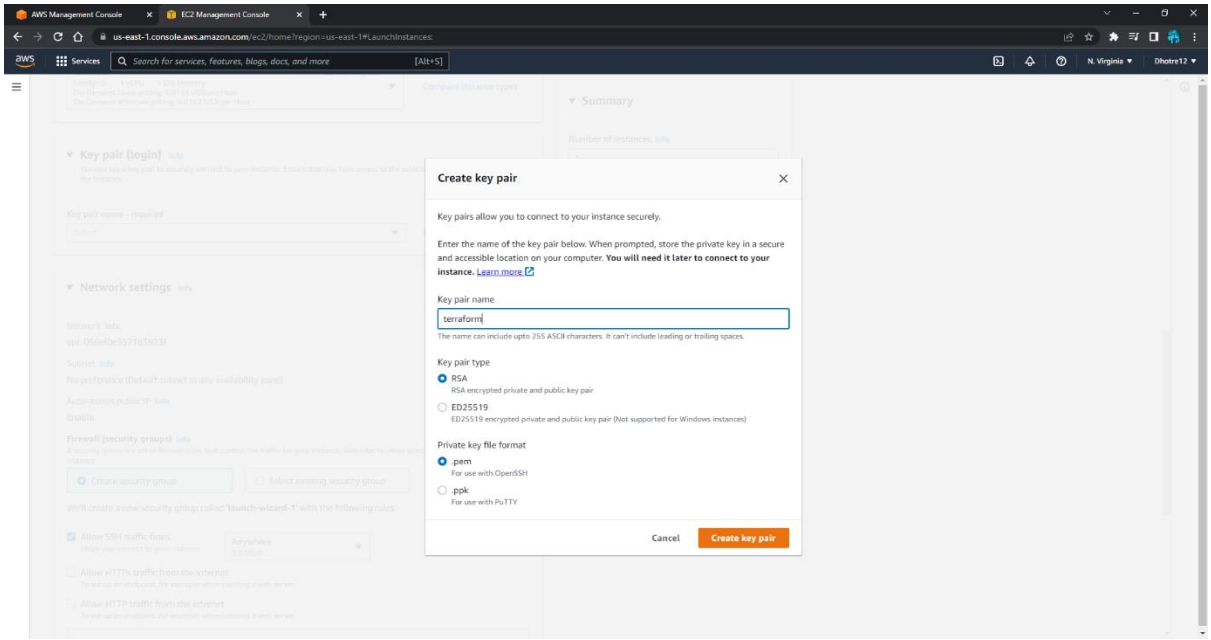
7. ***Plan***: It is one of the stages in the Terraform lifecycle where it determines what needs to be created, updated, or destroyed to move from the real/current state of the infrastructure to the desired state.

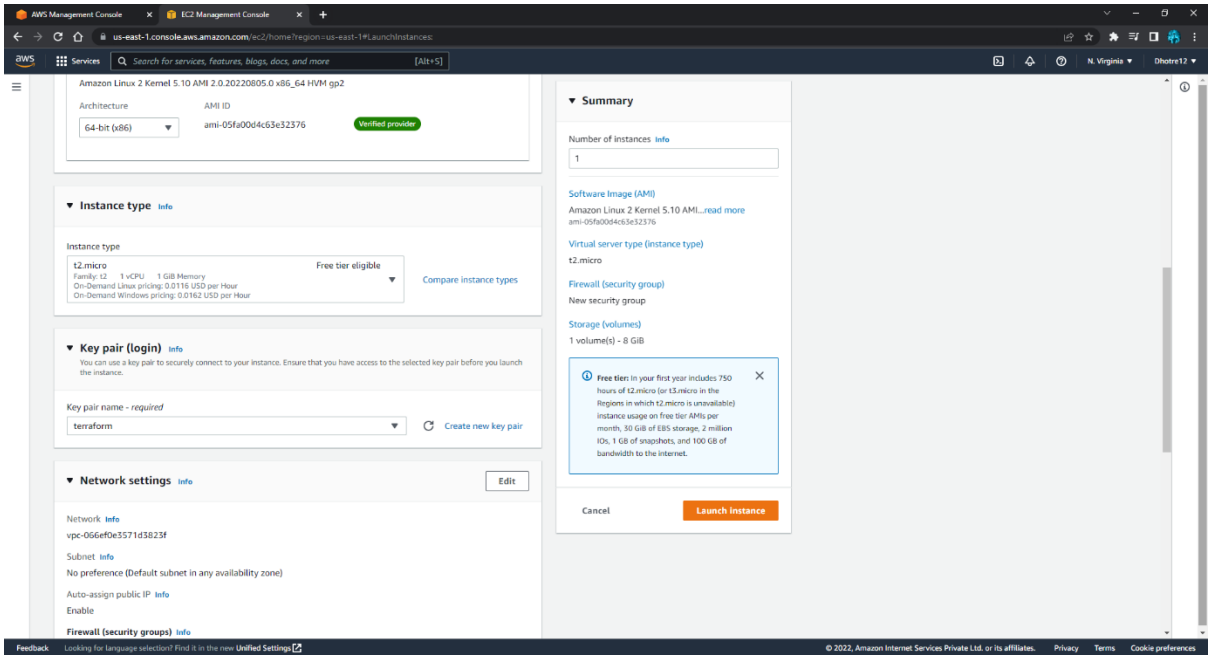
8. ***Apply***: It is one of the stages in the Terraform lifecycle where it applies the changes real/current state of the infrastructure in order to achieve the desired state.

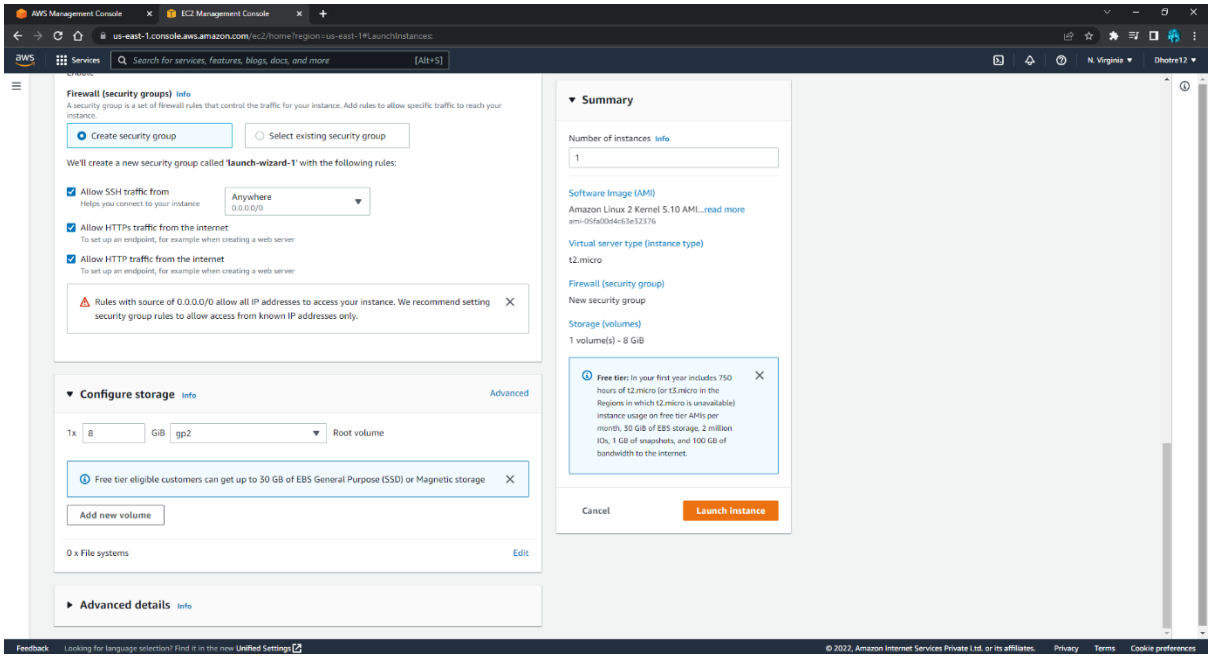
Result:

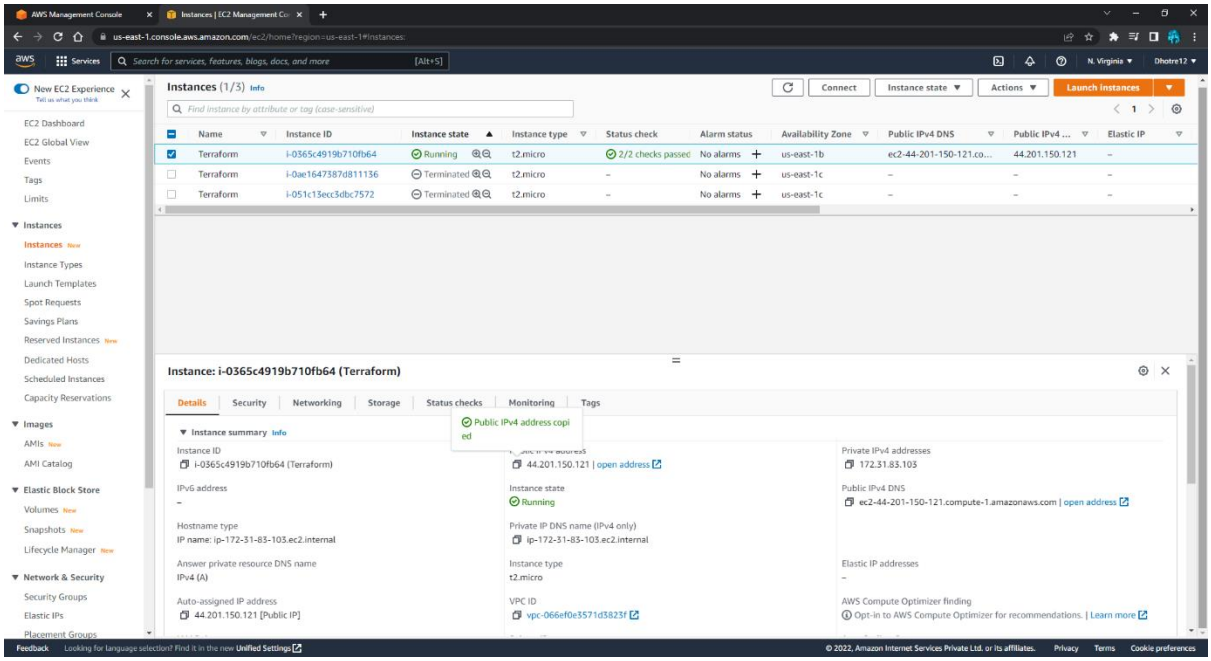
1. Create EC2 Instances



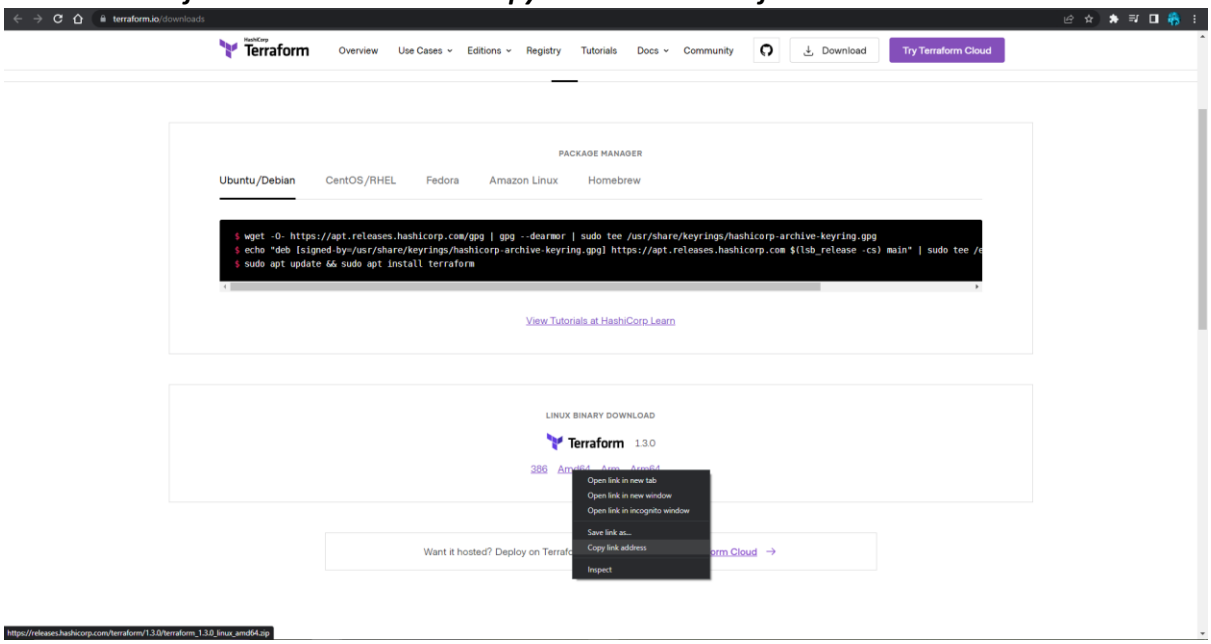



AWS Management Console
EC2 Management Console
us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LaunchInstances:
Services
Search for services, features, blogs, docs, and more
[Alt+S]
N. Virginia
Dhotre12
Amazon Linux 2 Kernel 5.10 AMI 2.0.20220805.0 x86_64 HVM gp2
Architecture
64-bit (x86)
AMI ID
ami-05fa00d4c63e32376
Verified provider
Instance type Info
Instance type
t2.micro
Free tier eligible
Family: t2 1 vCPU 1 GiB Memory
On-Demand Linux pricing: 0.0116 USD per Hour
On-Demand Windows pricing: 0.0162 USD per Hour
Compare instance types
Key pair (login) Info
You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.
Key pair name - required
terraform
Create new key pair
Network settings Info
Edit
Network Info
vpc-066ef0e3571d3823f
Subnet Info
No preference (Default subnet in any availability zone)
Auto-assign public IP Info
Enable
Firewall (security groups) Info
Summary
Number of instances Info
1
Software Image (AMI)
Amazon Linux 2 Kernel 5.10 AMI...read more
ami-05fa00d4c63e32376
Virtual server type (instance type)
t2.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.
Cancel
Launch instance
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Internet Services Private Ltd. or its affiliates.
Privacy
Terms
Cookie preferences


AWS Management Console
EC2 Management Console
us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LaunchInstances:
Services
Search for services, features, blogs, docs, and more
[Alt+S]
N. Virginia
Dhotre12
Firewall (security groups) Info
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.
Create security group
Select existing security group
We'll create a new security group called 'launch-wizard-1' with the following rules:
Allow SSH traffic from
Helps you connect to your instance
Anywhere
0.0.0.0/0
Allow HTTPs traffic from the internet
To set up an endpoint, for example when creating a web server
Allow HTTP traffic from the internet
To set up an endpoint, for example when creating a web server
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.
Configure storage Info
Advanced
1x
8
GiB
gp2
Root volume
Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage
Add new volume
0 x File systems
Edit
Advanced details Info
Summary
Number of instances Info
1
Software Image (AMI)
Amazon Linux 2 Kernel 5.10 AMI...read more
ami-05fa00d4c63e32376
Virtual server type (instance type)
t2.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.
Cancel
Launch instance
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Internet Services Private Ltd. or its affiliates.
Privacy
Terms
Cookie preferences



2. *Go to terraform download site and copy the link address of Linux*



3. In git bash execute the following code:

```
ec2-user@ip-172-31-83-103:~

Aakash@Aakash MINGW64 ~/Desktop
$ ssh -i terraform.pem ec2-user@44.201.150.121
The authenticity of host '44.201.150.121 (44.201.150.121)' can't be established.
ED25519 key fingerprint is SHA256:tZ9dkIwk2gkxNI7GMnHaK4Dsw+NNsKHLkoqmViGVEG8.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '44.201.150.121' (ED25519) to the list of known hosts
.

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
4 package(s) needed for security, out of 11 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-83-103 ~]$ wget https://releases.hashicorp.com/terraform/1.3
.0/terraform_1.3.0_linux_amd64.zip
--2022-09-21 15:30:10--  https://releases.hashicorp.com/terraform/1.3.0/terrafor
m_1.3.0_linux_amd64.zip
Resolving releases.hashicorp.com (releases.hashicorp.com)... 52.85.151.35, 52.85
.151.63, 52.85.151.96, ...
Connecting to releases.hashicorp.com (releases.hashicorp.com)|52.85.151.35|:443.
.. connected.
HTTP request sent, awaiting response... 200 OK
Length: 19450952 (19M) [application/zip]
Saving to: 'terraform_1.3.0_linux_amd64.zip'

100%[======================================>] 19,450,952  47.9MB/s   in 0.4s

2022-09-21 15:30:10 (47.9 MB/s) - 'terraform_1.3.0_linux_amd64.zip' saved [19450
952/19450952]

[ec2-user@ip-172-31-83-103 ~]$ ls
terraform_1.3.0_linux_amd64.zip
[ec2-user@ip-172-31-83-103 ~]$ unzip terraform_1.3.0_linux_amd64.zip
Archive:  terraform_1.3.0_linux_amd64.zip
  inflating: terraform
[ec2-user@ip-172-31-83-103 ~]$ ls
terraform  terraform_1.3.0_linux_amd64.zip
[ec2-user@ip-172-31-83-103 ~]$ sudo mv terraform /usr/local/bin/
[ec2-user@ip-172-31-83-103 ~]$ terraform --version
Terraform v1.3.0
on linux_amd64
[ec2-user@ip-172-31-83-103 ~]$
```

Conclusion: We have studied to understand terraform lifecycle, core concepts/terminologies and install it on a Linux Machine and also learned to implement them.

Questionnaire:

1. What do you understand by Terraform in AWS?

Terraform is a part of the AWS DevOps Competency and also an AWS Partner Network (APN) advanced technology partner. It is similar to AWS Cloud Formation in the sense that it is also an "infrastructure as code" tool that allows you to create, update, and version your AWS infrastructure.

2. What are the key features of Terraform?

Terraform helps you manage all of your infrastructures as code and construct it as and when needed. Here are its key main features:

- A console that allows users to observe functions
- The ability to translate HCL code into JSON format
- A configuration language that supports interpolation
- A module count that keeps track of the number of modules applied to the infrastructure.

3. Define IAC?

IAC or Infrastructure as Code allows you to build, change, and manage your infrastructure through coding instead of manual processes. The configuration files are created according to your infrastructure specifications and these configurations can be edited and distributed securely within an organization.

4. What are the most useful Terraform commands?

Some of the most useful Terraform commands are:

- ***terraform init*** - initializes the current directory
- ***terraform refresh*** - refreshes the state file
- ***terraform output*** - views Terraform outputs
- ***terraform apply*** - applies the Terraform code and builds stuff
- ***terraform destroy*** - destroys what has been built by Terraform
- ***terraform graph*** - creates a DOT-formatted graph
- ***terraform plan*** - a dry run to see what Terraform will do

5. Are callbacks possible with Terraform on Azure?

By using the Azure Event Hubs, callbacks are probable on Azure. Terraform's Azure supplier provides effortless functionality to users. Microsoft Azure Cloud Shell provides an already installed Terraform occurrence.

Experiment 6

Aim: To Build, change, and destroy AWS / GCP /Microsoft Azure/ DigitalOcean infrastructure Using Terraform.

Theory:
Terraform is an open-source, infrastructure as code, software tool created by HashiCorp. Users define and provide data center infrastructure using a declarative configuration language known as HashiCorp Configuration Language (HCL), or optionally JSON.

Design
Terraform manages external resources (such as public cloud infrastructure, private cloud infrastructure, network appliances, software as a service, and platform as a service) with "providers". HashiCorp maintains an extensive list of official providers, and can also integrate with community-developed providers. Users can interact with Terraform providers by declaring resources or by calling data sources. Rather than using imperative commands to provision resources, Terraform uses declarative configuration to describe the desired final state. Once a user invokes Terraform on a given resource, Terraform will perform CRUD actions on the user's behalf to accomplish the desired state. The infrastructure as code can be written as modules, promoting reusability and maintainability.

Terraform supports a number of cloud infrastructure providers such as Amazon Web Services, Microsoft Azure, IBM Cloud, Serverspace, Google Cloud Platform, DigitalOcean, Oracle Cloud Infrastructure, Yandex.Cloud, VMware vSphere, and OpenStack.

HashiCorp maintains a Terraform Module Registry, launched in 2017. In 2019, Terraform introduced the paid version called Terraform Enterprise for larger organizations.

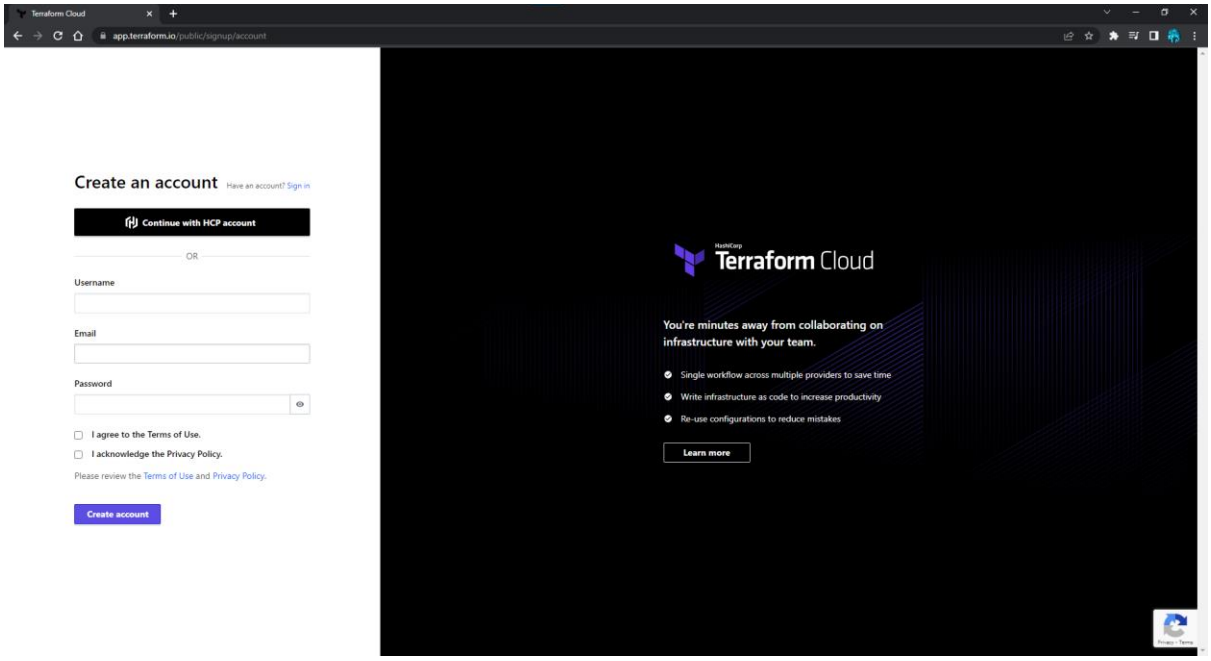
Terraform has a great set of features that make it worth adding to your tool belt, including:

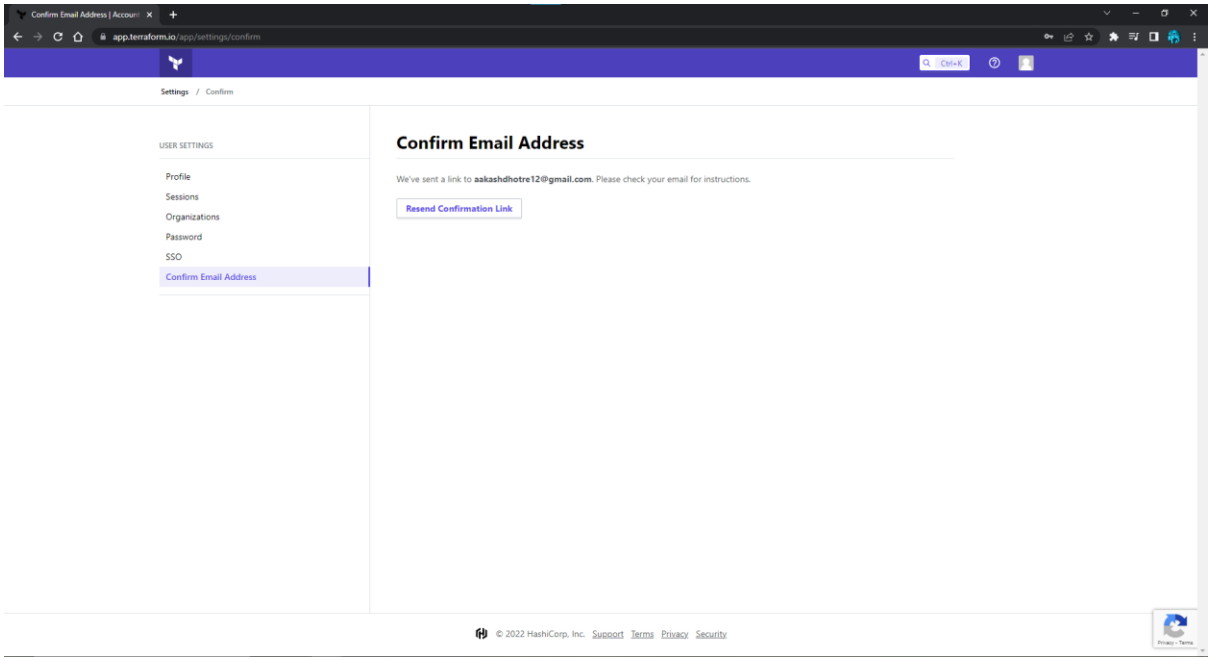
- Friendly custom syntax, but also has support for JSON.
- Visibility into changes before they actually happen.
- Built-in graphing feature to visualize the infrastructure.
- Understands resource relationships. One example is failures are isolated to dependent resources while non-dependent resources still get created, updated, or destroyed.
- Open source project with a community of thousands of contributors who add features and updates.
- The ability to break down the configuration into smaller chunks for better organization, re-use, and maintainability. The last part of this article goes into this feature in detail.

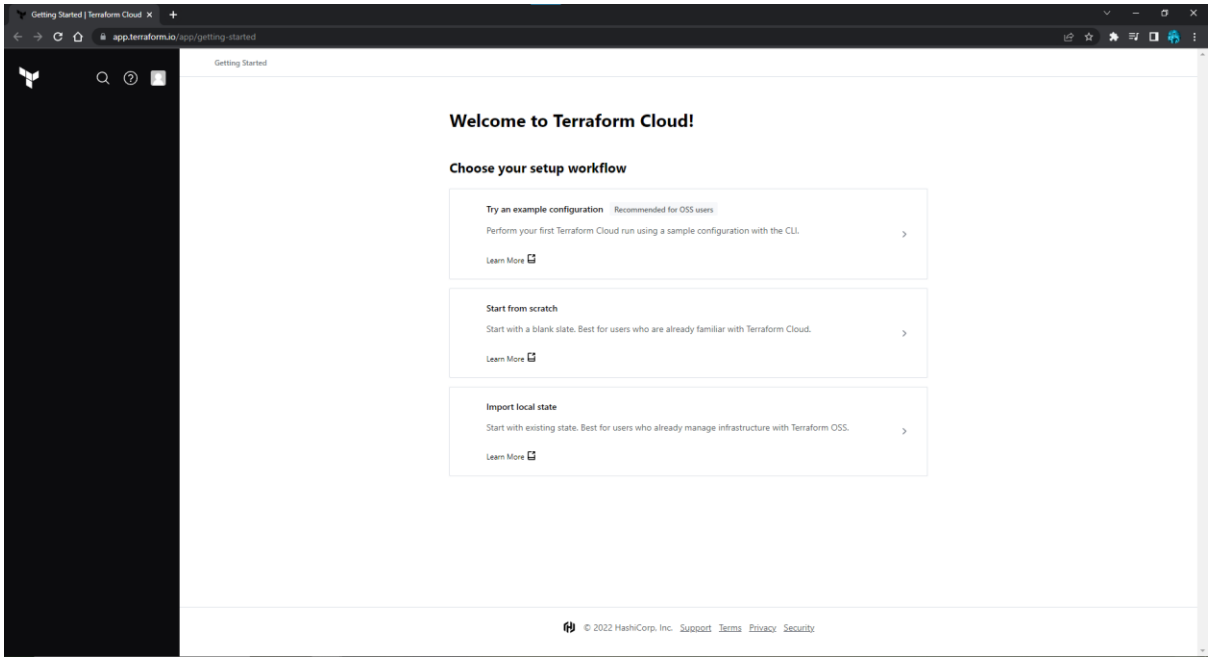
Result:

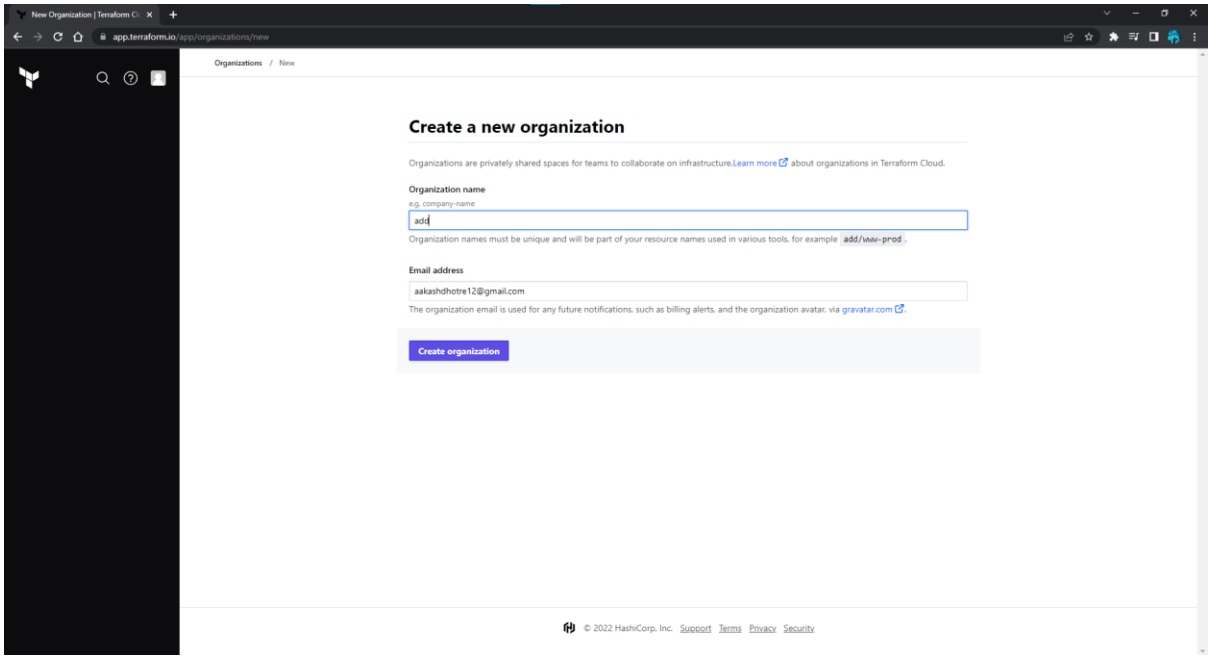
Step 1:

1. Create terraform cloud account.
2. Follow this link:https://app.terraform.io/signup/account









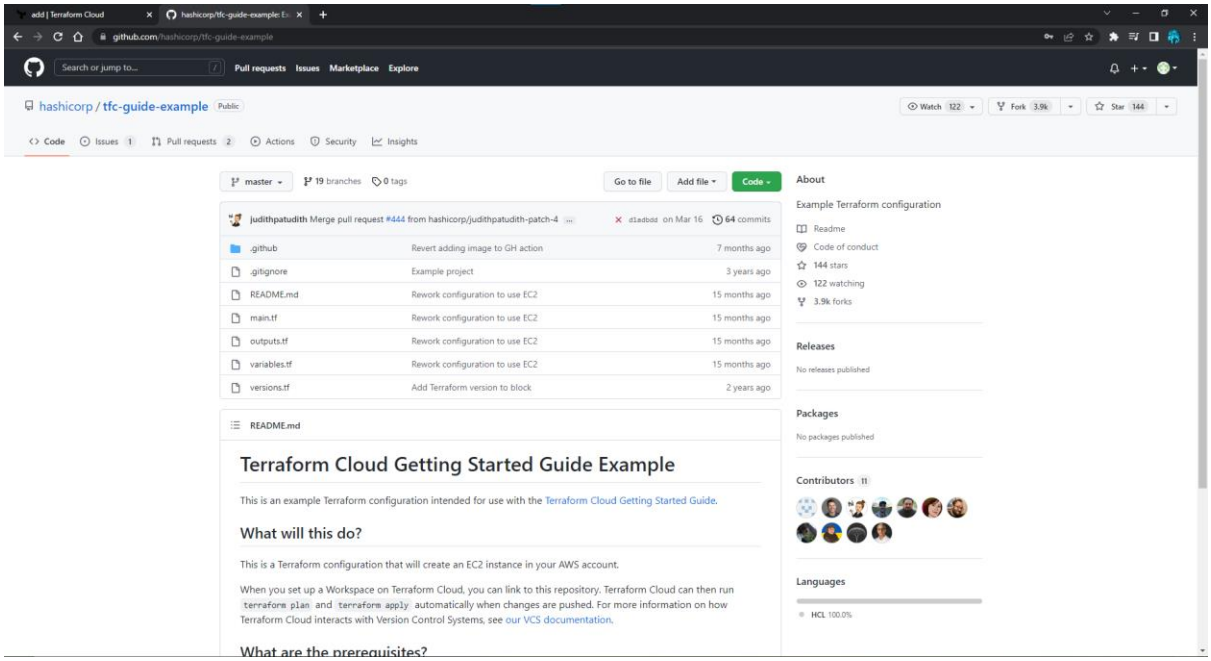
Step 2:

Prerequisites

- An AWS account
- A Github account

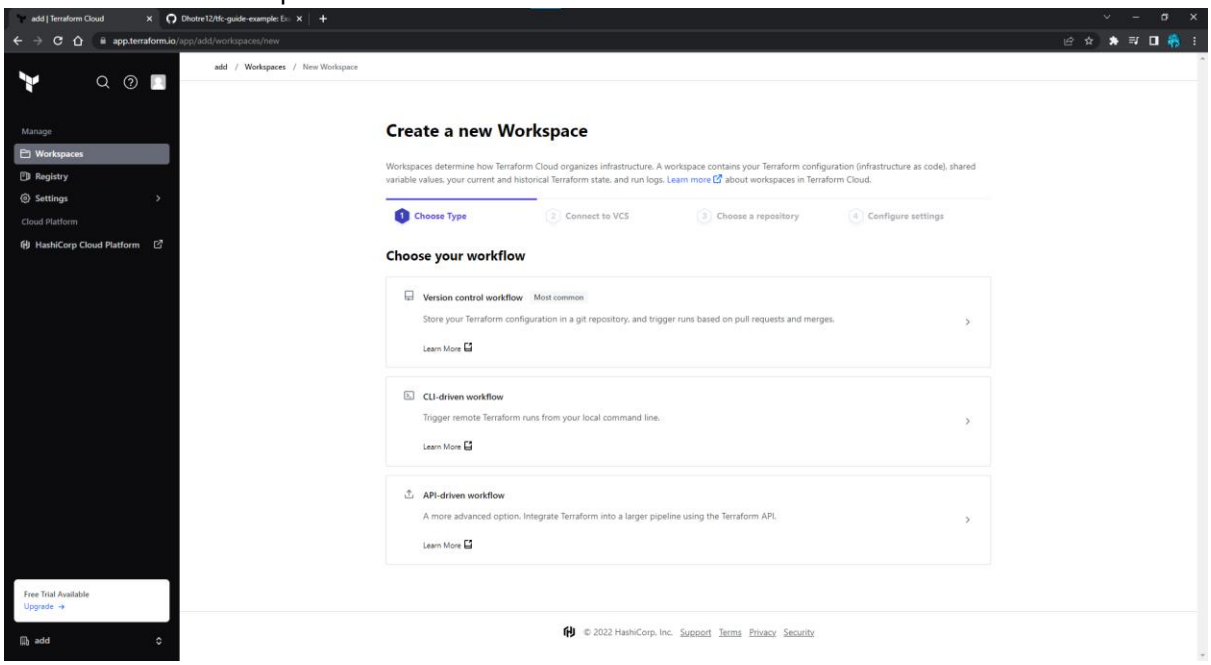
Fork a Github repository

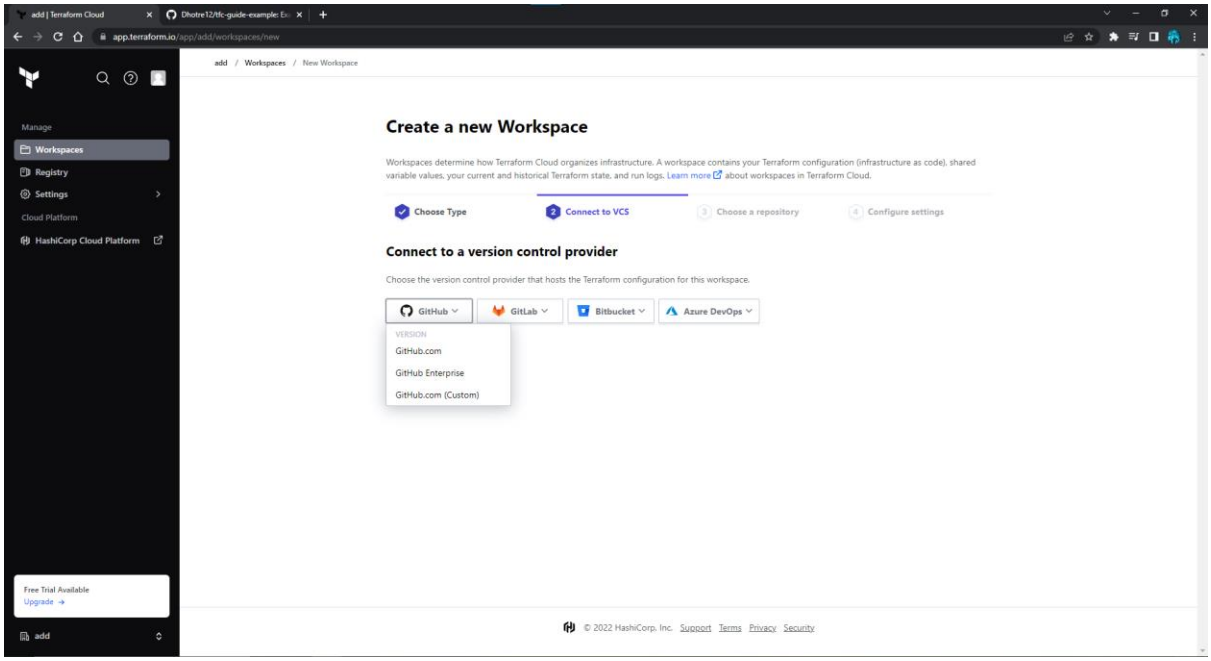
- Click the"Fork"button at the top right of the page to copy the repository to your GitHub account.



Step 3:

- Connect Terraform Cloud to GitHub
- Click on new workspace





Create a new Workspace

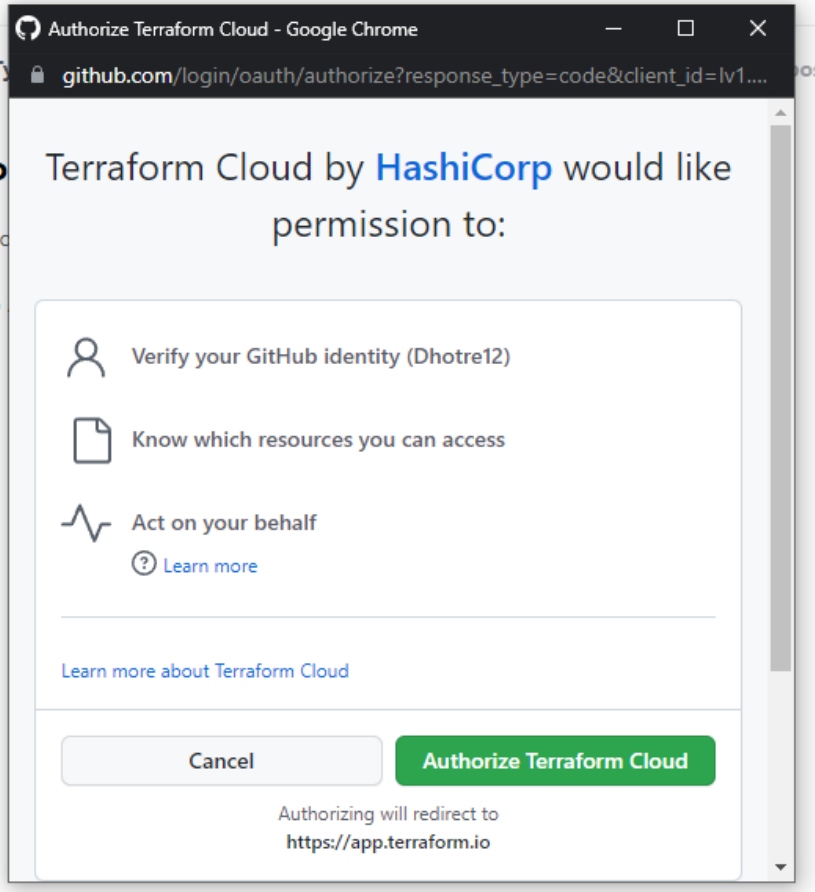
Workspaces determine how Terraform Cloud organizes infrastructure. A workspace contains your Terraform co
variable values, your current and historical Terraform state, and run logs. Learn more about workspaces in T

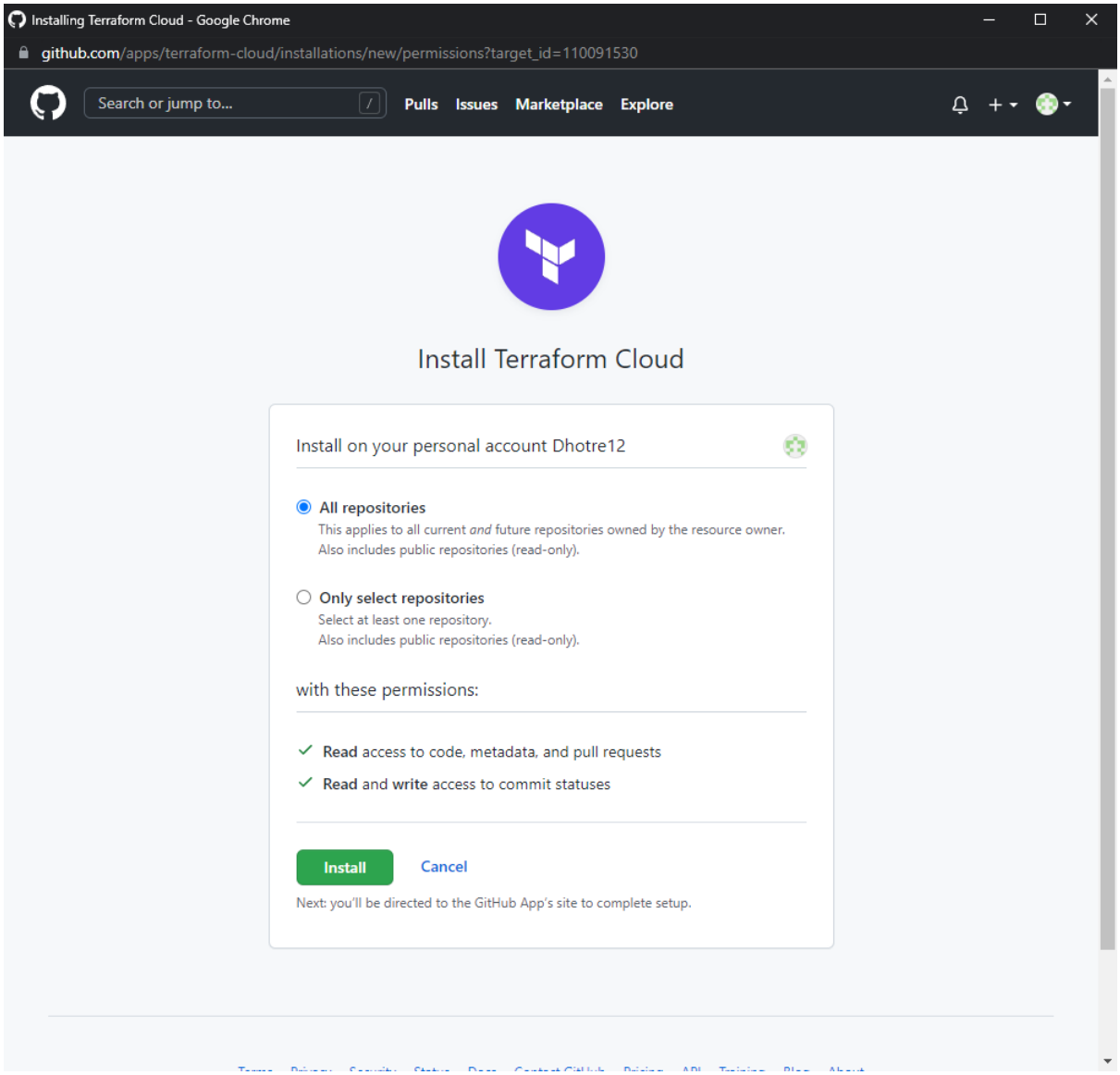
Choose Ty ository

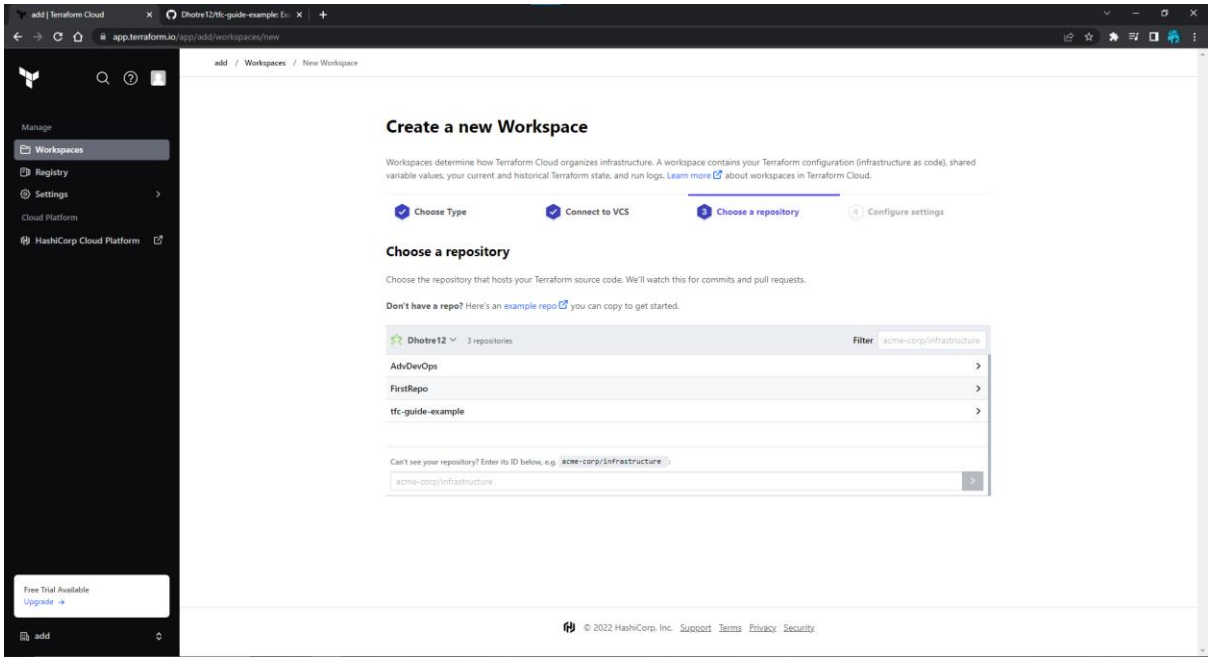
Connect to

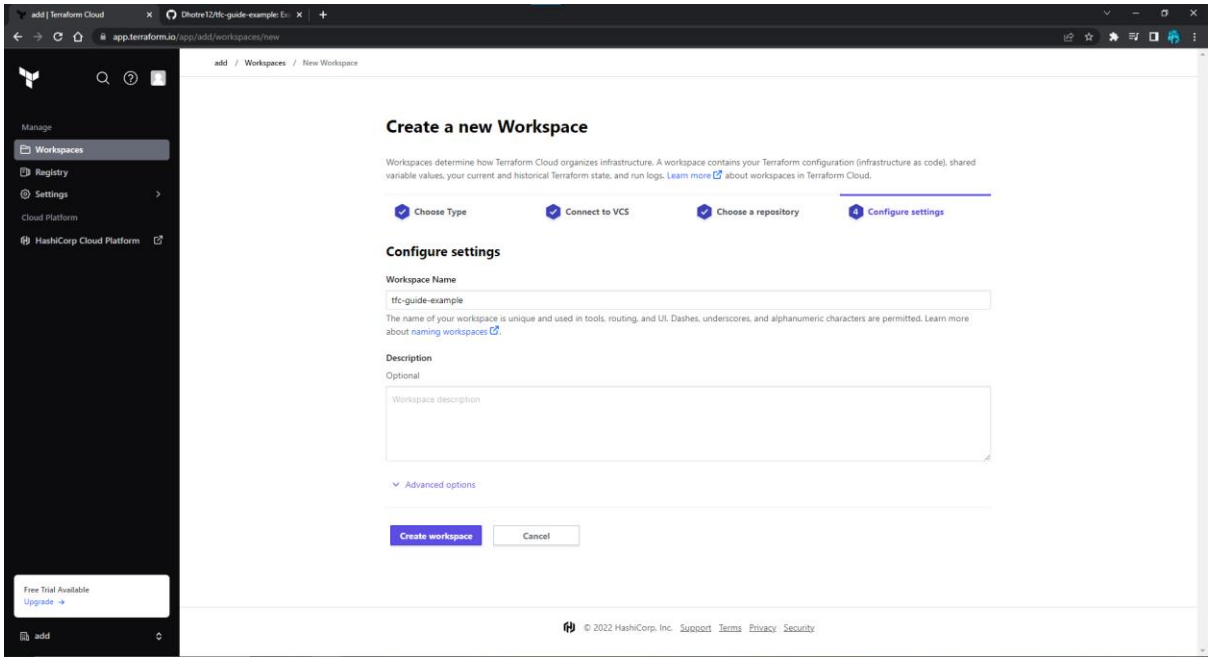
Choose the versio

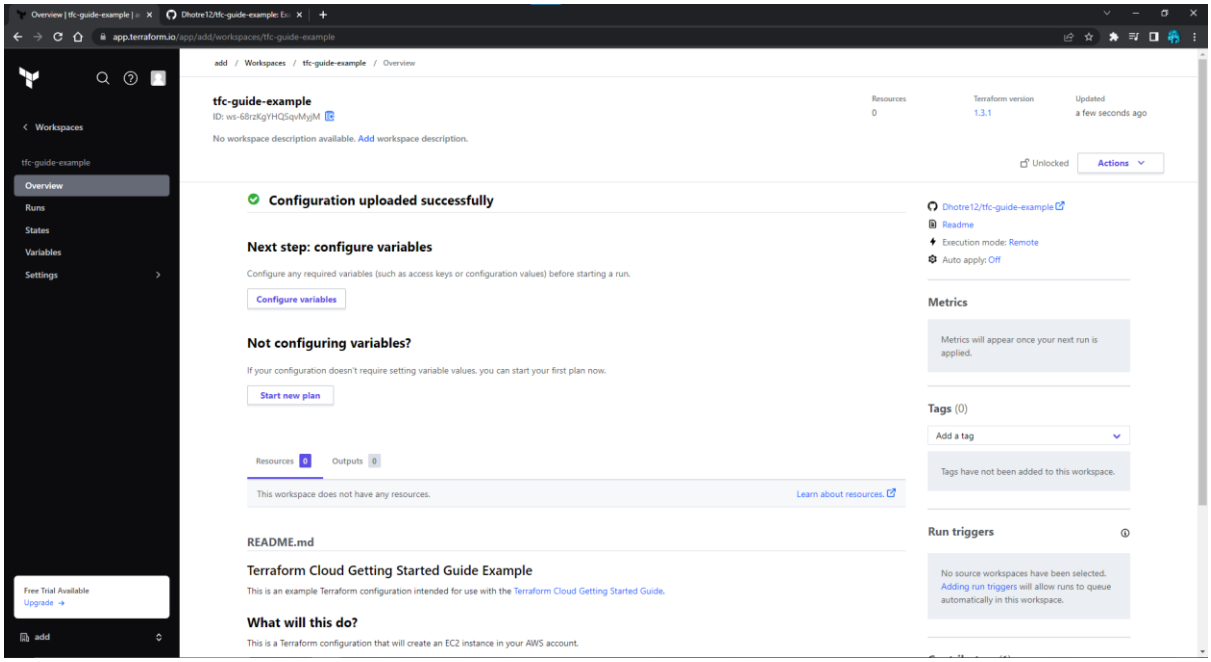
Installing GitHub




Installing Terraform Cloud - Google Chrome
github.com/apps/terraform-cloud/installations/new/permissions?target_id=110091530
Search or jump to...
Pulls
Issues
Marketplace
Explore
Install Terraform Cloud
Install on your personal account Dhotre12
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 code, metadata, and pull requests
Read and write access to commit statuses
Install
Cancel
Next: you'll be directed to the GitHub App's site to complete setup.


app.terraform.io/app/add/workspaces/new
add / Workspaces / New Workspace
Manage
Workspaces
Registry
Settings
Cloud Platform
HashiCorp Cloud Platform
Create a new Workspace
Workspaces determine how Terraform Cloud organizes infrastructure. A workspace contains your Terraform configuration (infrastructure as code), shared variable values, your current and historical Terraform state, and run logs. Learn more about workspaces in Terraform Cloud.
Choose Type
Connect to VCS
Choose a repository
Configure settings
Choose a repository
Choose the repository that hosts your Terraform source code. We'll watch this for commits and pull requests.
Don't have a repo? Here's an example repo you can copy to get started.
Dhotre12
3 repositories
Filter
acme-corp/infrastructure
AdvDevOps
FirstRepo
tfc-guide-example
Can't see your repository? Enter its ID below, e.g. acme-corp/infrastructure
Free Trial Available
Upgrade
add
© 2022 HashiCorp, Inc. Support Terms Privacy Security

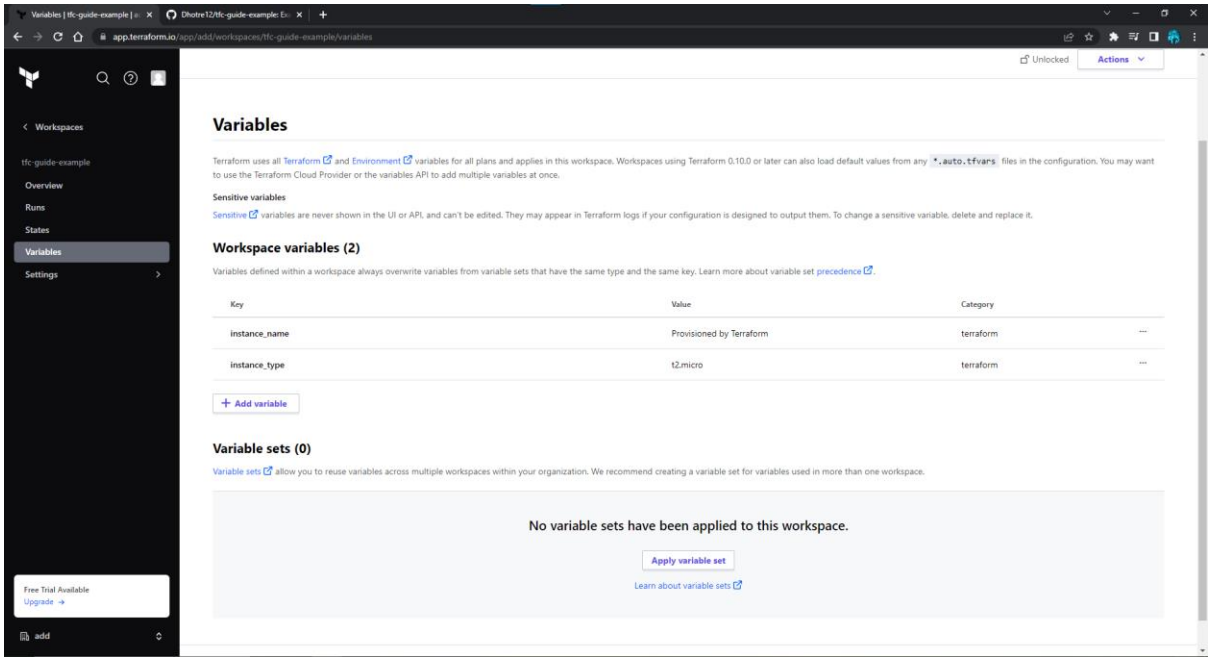


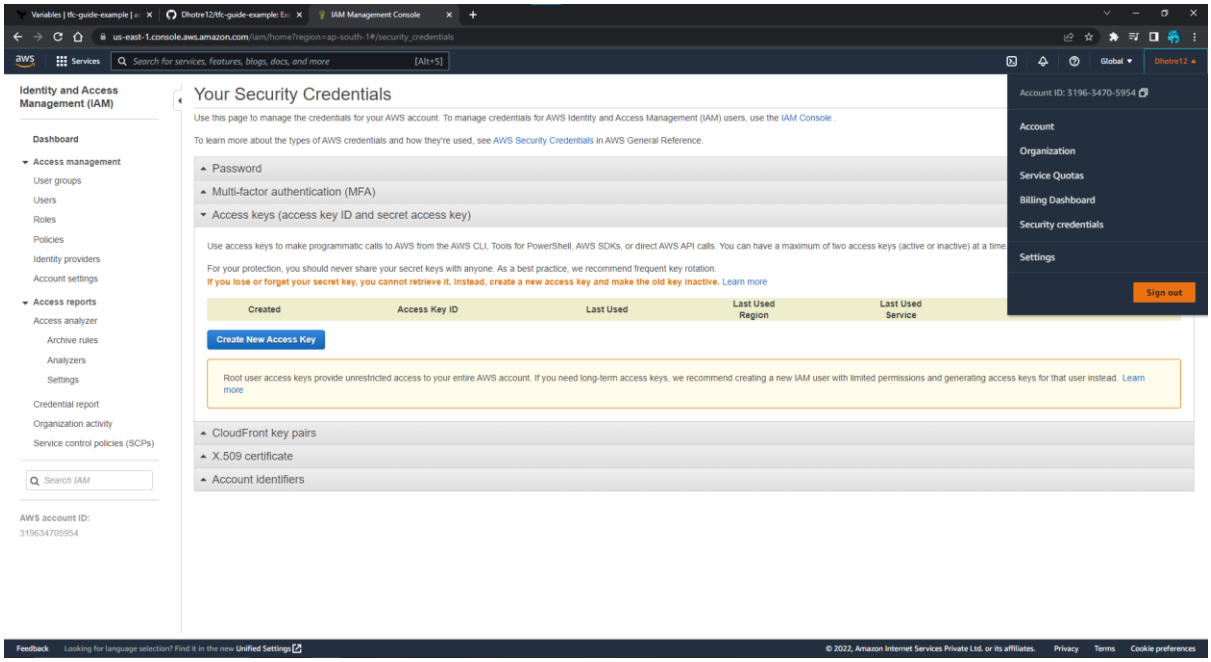


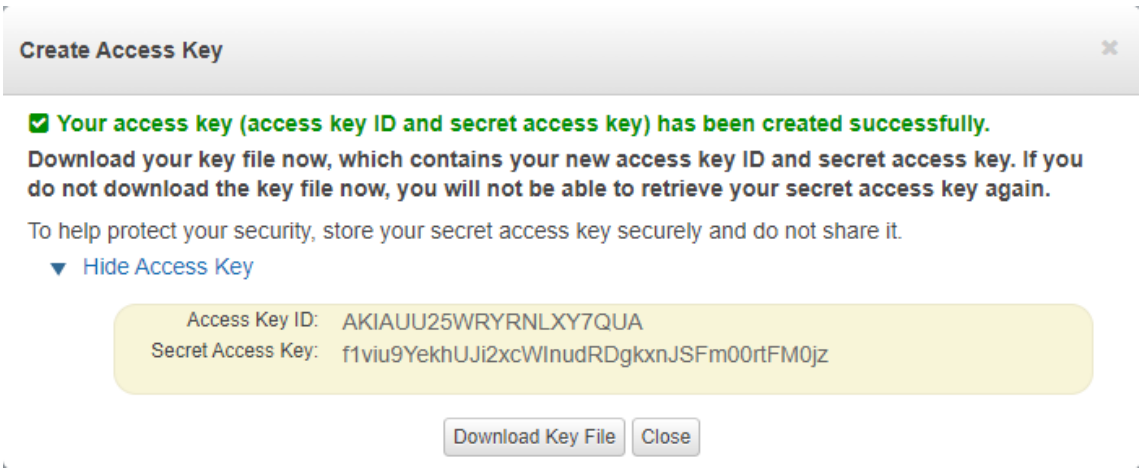
Step 4:

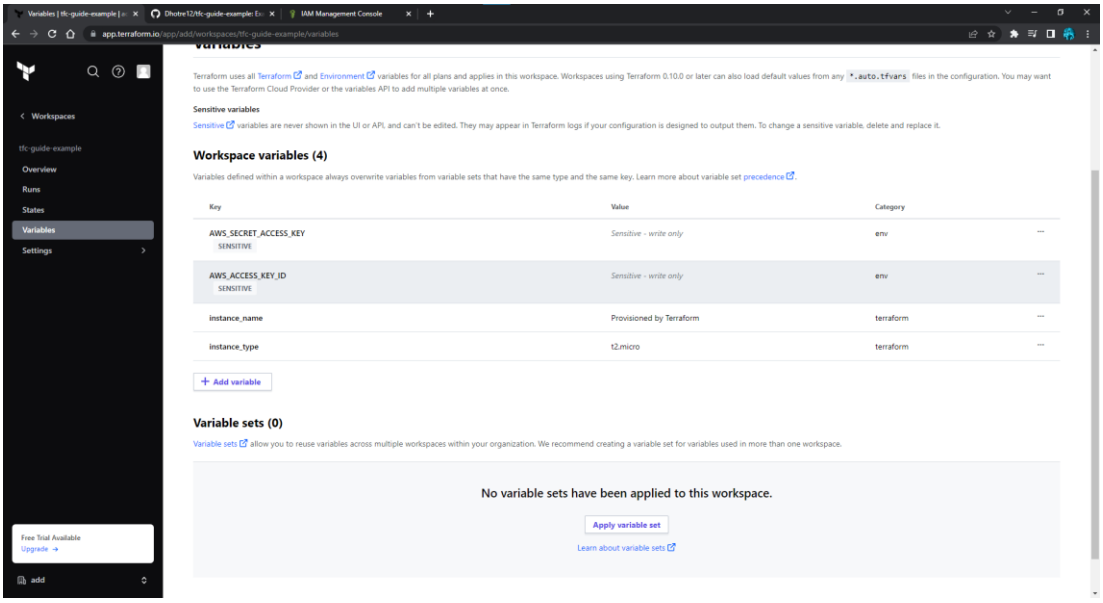
Create Infrastructure

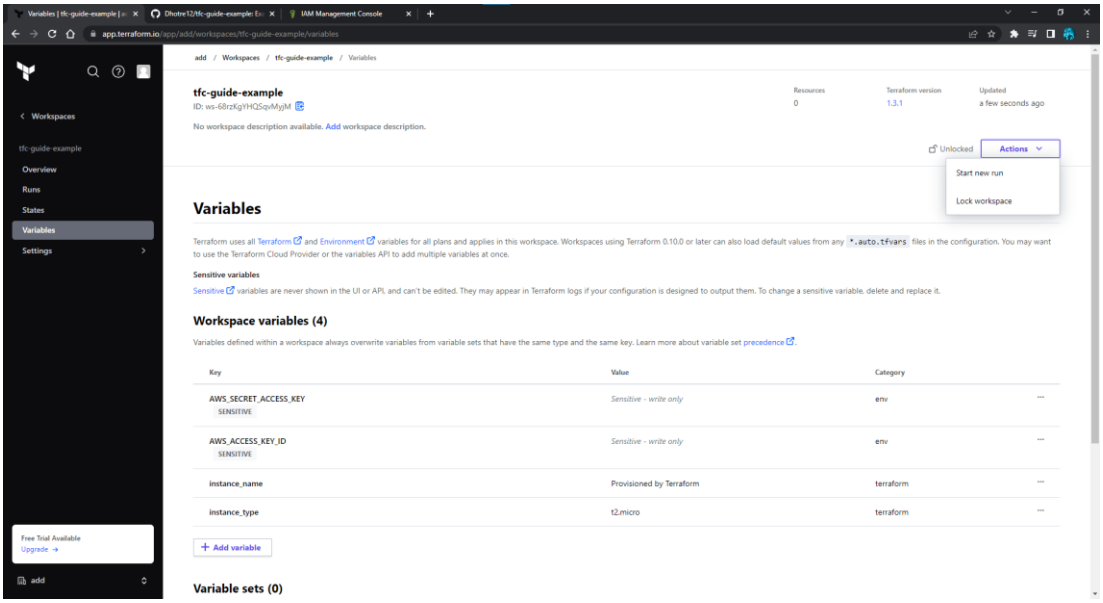
● Configure Terraform variables :Terraform Cloud will define Terraform Variables as input variables in Terraform's configuration language.You can use them to customize the infrastructure that Terraform creates from your configuration.

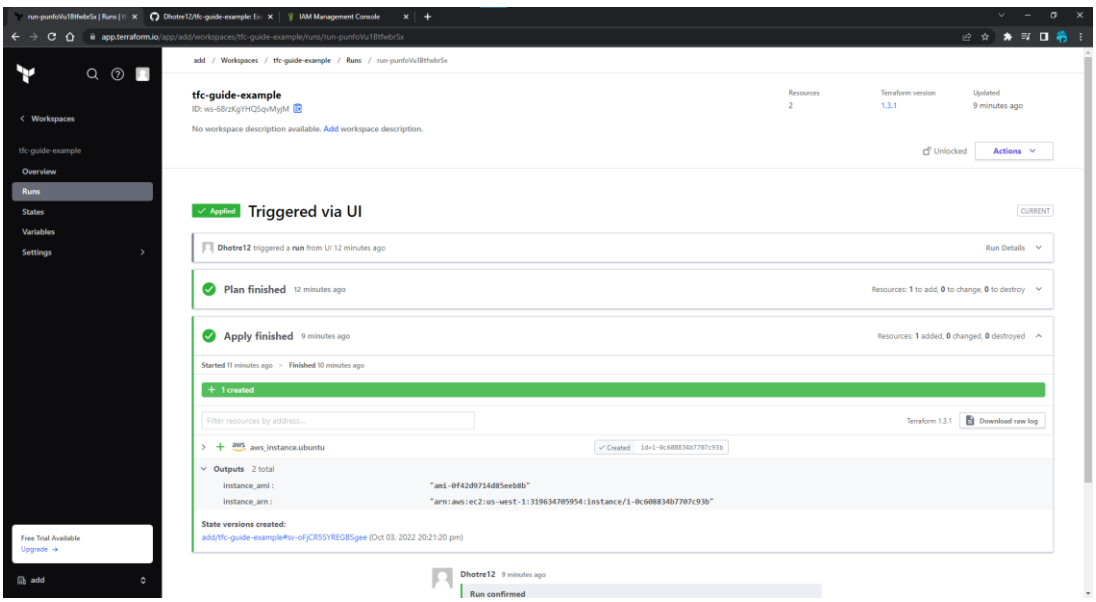

Variables
Workspaces
tfc-guide-example
Overview
Runs
States
Variables
Settings
Unlocked
Actions
Terraform uses all Terraform and Environment variables for all plans and applies in this workspace. Workspaces using Terraform 0.10.0 or later can also load default values from any *.auto.tfvars files in the configuration. You may want to use the Terraform Cloud Provider or the variables API to add multiple variables at once.
Sensitive variables
Sensitive variables are never shown in the UI or API, and can't be edited. They may appear in Terraform logs if your configuration is designed to output them. To change a sensitive variable, delete and replace it.
Workspace variables (2)
Variables defined within a workspace always overwrite variables from variable sets that have the same type and the same key. Learn more about variable set precedence.
Key
Value
Category
instance_name
Provisioned by Terraform
terraform
instance_type
t2.micro
terraform
Add variable
Variable sets (0)
Variable sets allow you to reuse variables across multiple workspaces within your organization. We recommend creating a variable set for variables used in more than one workspace.
No variable sets have been applied to this workspace.
Apply variable set
Learn about variable sets
Free Trial Available
Upgrade
add

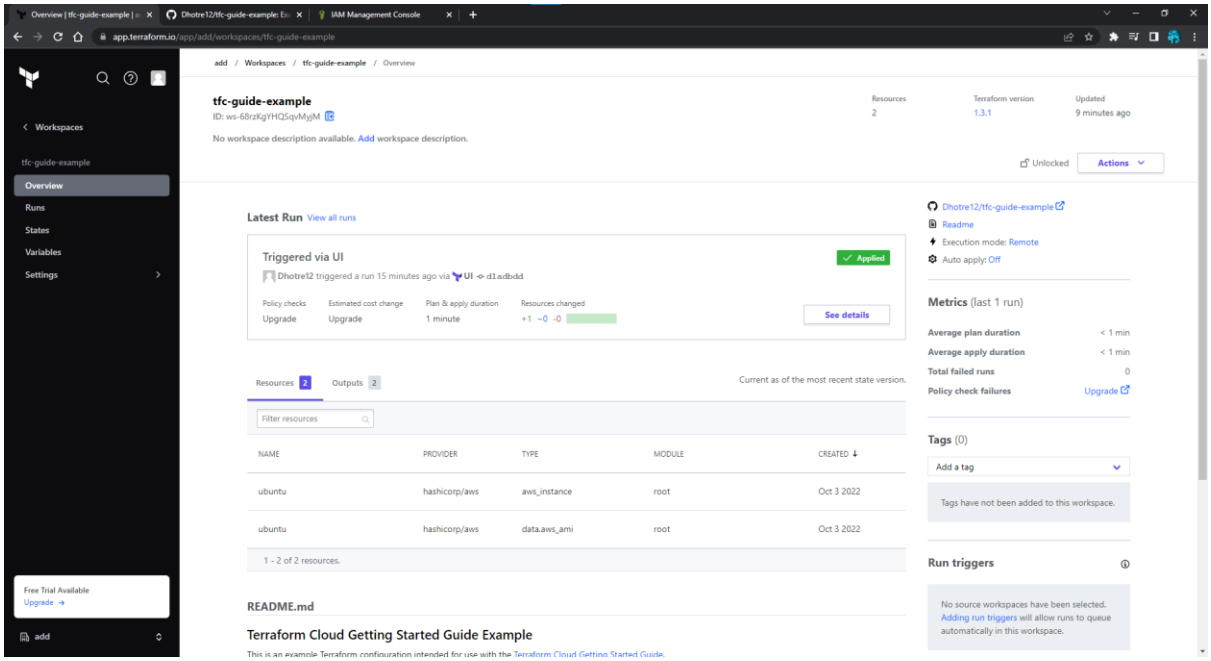

Identity and Access Management (IAM)
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Access reports
Access analyzer
Archive rules
Analyzers
Settings
Credential report
Organization activity
Service control policies (SCPs)
Search IAM
AWS account ID:
319634705954
Your Security Credentials
Use this page to manage the credentials for your AWS account. To manage credentials for AWS Identity and Access Management (IAM) users, use the IAM Console.
To learn more about the types of AWS credentials and how they're used, see AWS Security Credentials in AWS General Reference.
Password
Multi-factor authentication (MFA)
Access keys (access key ID and secret access key)
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
Created
Access Key ID
Last Used
Last Used Region
Last Used Service
Create New Access Key
Root user access keys provide unrestricted access to your entire AWS account. If you need long-term access keys, we recommend creating a new IAM user with limited permissions and generating access keys for that user instead. Learn more
CloudFront key pairs
X.509 certificate
Account identifiers
Account ID: 3196-3470-5954
Account
Organization
Service Quotas
Billing Dashboard
Security credentials
Settings
Sign out
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Internet Services Private Ltd. or its affiliates.
Privacy
Terms
Cookie preferences


Create Access Key
Your access key (access key ID and secret access key) has been created successfully.
Download your key file now, which contains your new access key ID and secret access key. If you do not download the key file now, you will not be able to retrieve your secret access key again.
To help protect your security, store your secret access key securely and do not share it.
Hide Access Key
Access Key ID: AKIAUU25WRYRNLXY7QUA
Secret Access Key: f1viu9YekhUJi2xcWInudRDgkxnJSFm00rtFM0jz
Download Key File
Close


Workspaces
tfc-guide-example
Overview
Runs
States
Variables
Settings
Free Trial Available
Upgrade
add
Terraform uses all Terraform and Environment variables for all plans and applies in this workspace. Workspaces using Terraform 0.10.0 or later can also load default values from any *.auto.tfvars files in the configuration. You may want to use the Terraform Cloud Provider or the variables API to add multiple variables at once.
Sensitive variables
Sensitive variables are never shown in the UI or API, and can't be edited. They may appear in Terraform logs if your configuration is designed to output them. To change a sensitive variable, delete and replace it.
Workspace variables (4)
Variables defined within a workspace always overwrite variables from variable sets that have the same type and the same key. Learn more about variable set precedence.
Key
Value
Category
AWS_SECRET_ACCESS_KEY
SENSITIVE
Sensitive - write only
env
AWS_ACCESS_KEY_ID
SENSITIVE
Sensitive - write only
env
instance_name
Provisioned by Terraform
terraform
instance_type
t2.micro
terraform
Add variable
Variable sets (0)
Variable sets allow you to reuse variables across multiple workspaces within your organization. We recommend creating a variable set for variables used in more than one workspace.
No variable sets have been applied to this workspace.
Apply variable set
Learn about variable sets


add / Workspaces / tfc-guide-example / Variables
tfc-guide-example
ID: ws-68rzKgYHQSqvMyjM
No workspace description available. Add workspace description.
Resources
0
Terraform version
1.3.1
Updated
a few seconds ago
Unlocked
Actions
Start new run
Lock workspace
Variables
Terraform uses all Terraform and Environment variables for all plans and applies in this workspace. Workspaces using Terraform 0.10.0 or later can also load default values from any *.auto.tfvars files in the configuration. You may want to use the Terraform Cloud Provider or the variables API to add multiple variables at once.
Sensitive variables
Sensitive variables are never shown in the UI or API, and can't be edited. They may appear in Terraform logs if your configuration is designed to output them. To change a sensitive variable, delete and replace it.
Workspace variables (4)
Variables defined within a workspace always overwrite variables from variable sets that have the same type and the same key. Learn more about variable set precedence.
Key
Value
Category
AWS_SECRET_ACCESS_KEY
SENSITIVE
Sensitive - write only
env
AWS_ACCESS_KEY_ID
SENSITIVE
Sensitive - write only
env
instance_name
Provisioned by Terraform
terraform
instance_type
t2.micro
terraform
Add variable
Variable sets (0)

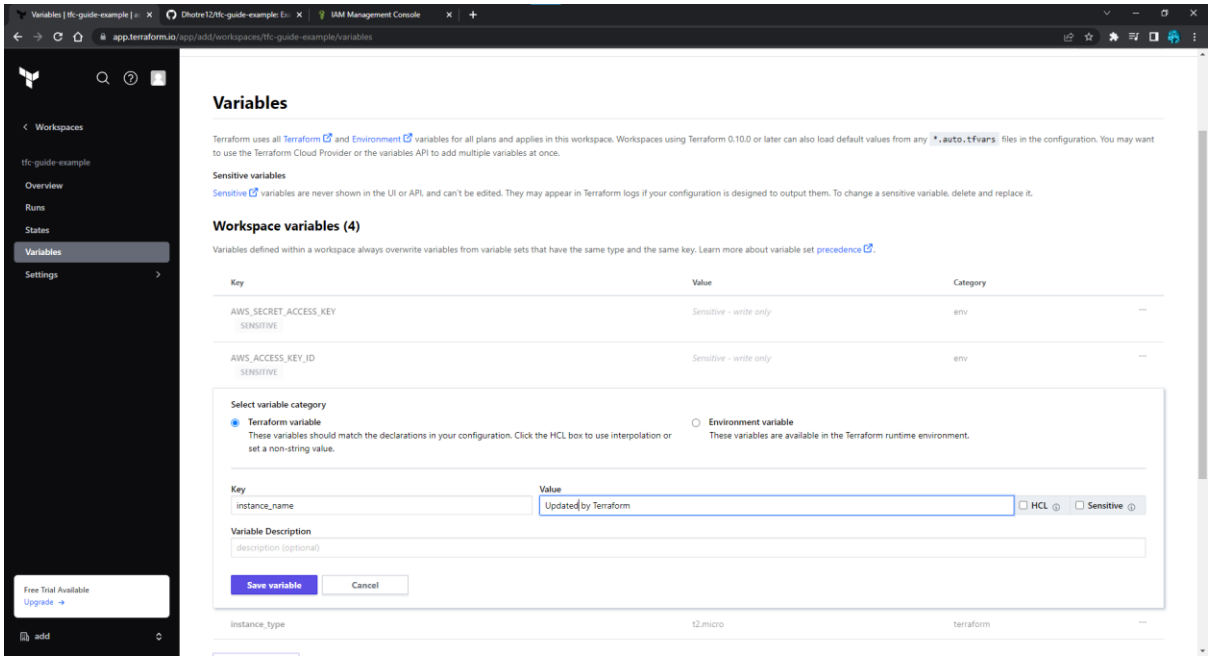

add / Workspaces / tfc-guide-example / Runs / run-punfoVu1BtfwbrSx
tfc-guide-example
ID: ws-68rzKgYHQSqvMyjM
No workspace description available. Add workspace description.
Resources
2
Terraform version
1.3.1
Updated
9 minutes ago
Unlocked
Actions
Applied
Triggered via UI
CURRENT
Dhotre12 triggered a run from UI 12 minutes ago
Run Details
Plan finished 12 minutes ago
Resources: 1 to add, 0 to change, 0 to destroy
Apply finished 9 minutes ago
Resources: 1 added, 0 changed, 0 destroyed
Started 11 minutes ago
Finished 10 minutes ago
1 created
Terraform 1.3.1
Download raw log
aws_instance.ubuntu
Created
Outputs 2 total
instance_ami : "ami-0f42d9714d85eeb8b"
instance_arn : "arn:aws:ec2:us-west-1:319634705954:instance/i-0c608834b7707c93b"
State versions created:
add/tfc-guide-example#sv-oFjCR5SYREGBSgee (Oct 03, 2022 20:21:20 pm)
Dhotre12 9 minutes ago
Run confirmed

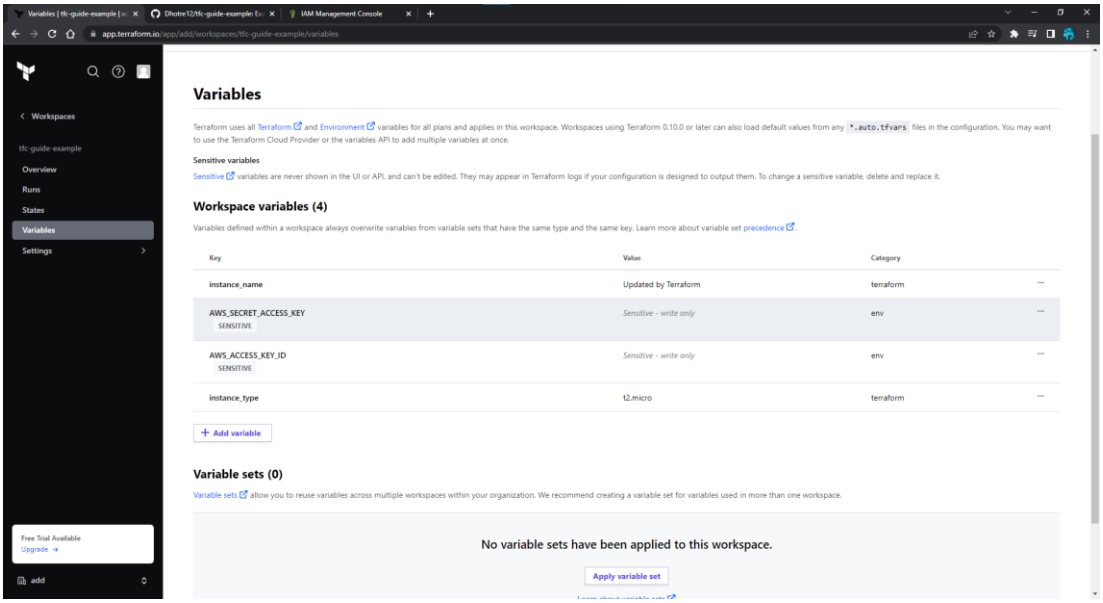


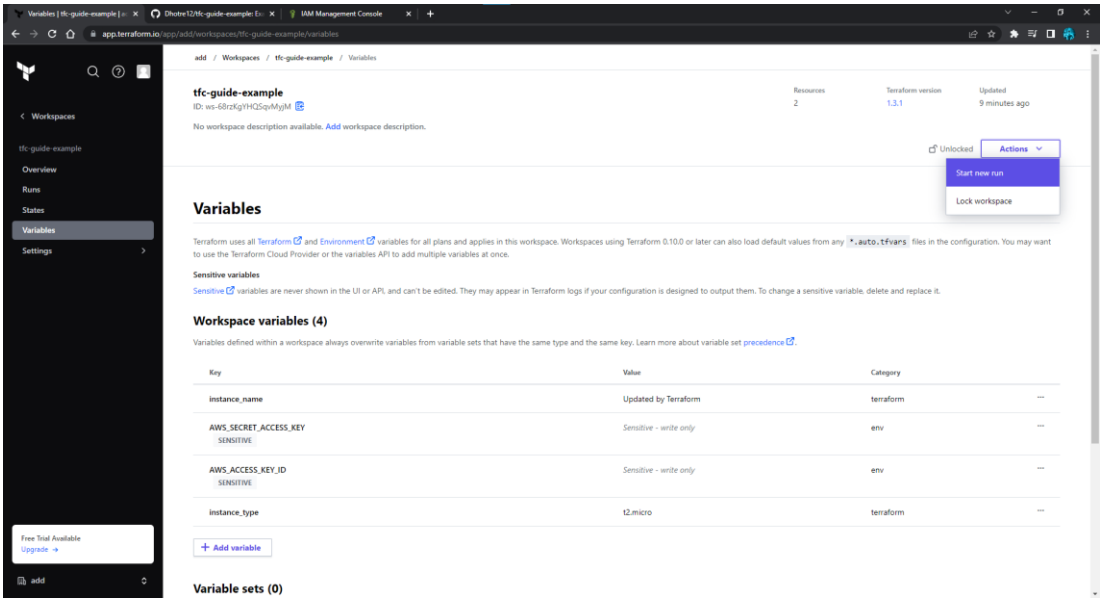
Step 5:

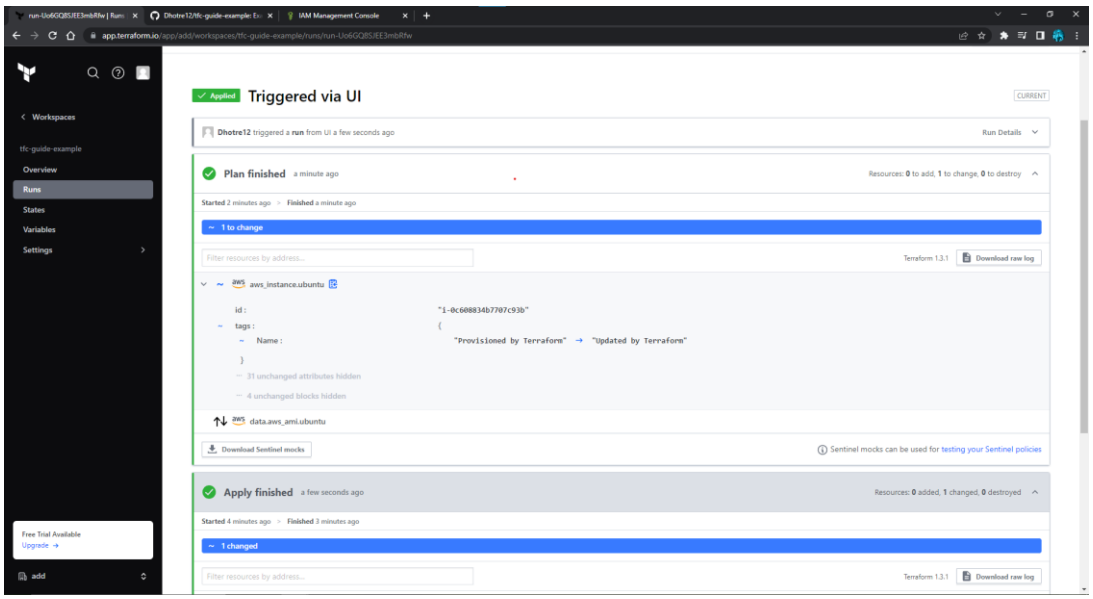
Change Infrastructure

There are two ways to update your workspace deployments on Terraform Cloud-changing the configuration in VCS or updating variables in the Terraform Cloud UI.




Variables
Terraform uses all Terraform and Environment variables for all plans and applies in this workspace. Workspaces using Terraform 0.10.0 or later can also load default values from any *.auto.tfvars files in the configuration. You may want to use the Terraform Cloud Provider or the variables API to add multiple variables at once.
Sensitive variables
Sensitive variables are never shown in the UI or API, and can't be edited. They may appear in Terraform logs if your configuration is designed to output them. To change a sensitive variable, delete and replace it.
Workspace variables (4)
Variables defined within a workspace always overwrite variables from variable sets that have the same type and the same key. Learn more about variable set precedence.
Key
Value
Category
instance_name
Updated by Terraform
terraform
AWS_SECRET_ACCESS_KEY
SENSITIVE
Sensitive - write only
env
AWS_ACCESS_KEY_ID
SENSITIVE
Sensitive - write only
env
instance_type
t2.micro
terraform
Add variable
Variable sets (0)
Variable sets allow you to reuse variables across multiple workspaces within your organization. We recommend creating a variable set for variables used in more than one workspace.
No variable sets have been applied to this workspace.
Apply variable set

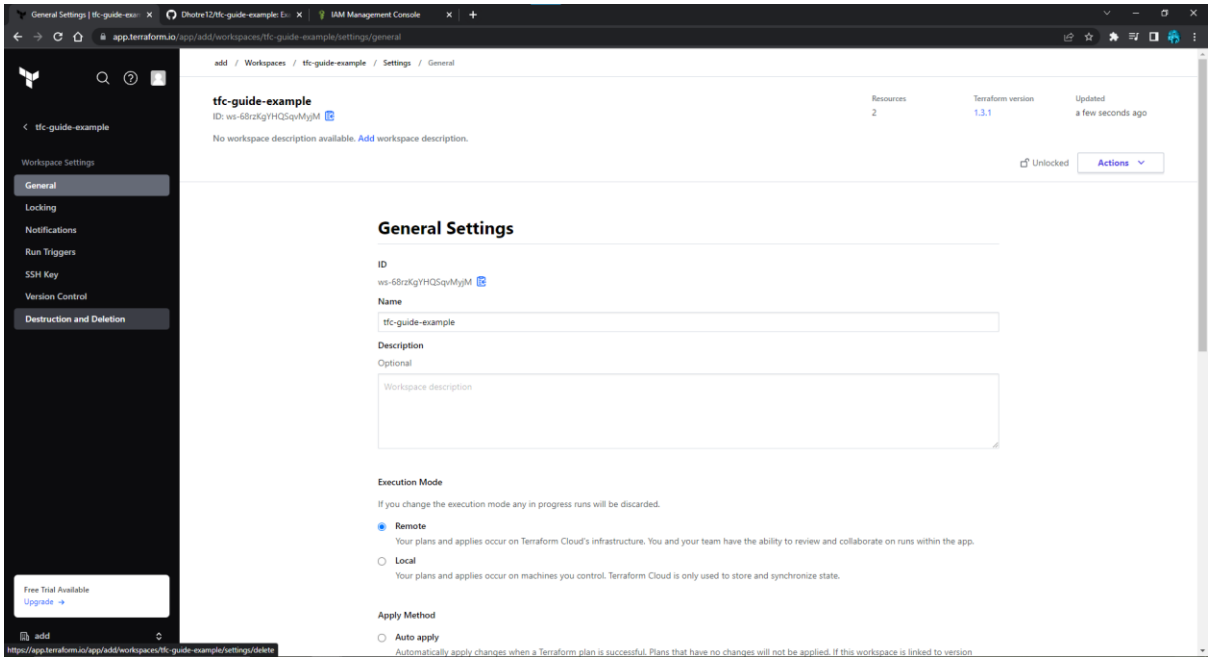

tfc-guide-example
ID: ws-68rzKgYHQSqvMyjM
No workspace description available. Add workspace description.
Resources
2
Terraform version
1.3.1
Updated
9 minutes ago
Unlocked
Actions
Start new run
Lock workspace
Variables
Terraform uses all Terraform and Environment variables for all plans and applies in this workspace. Workspaces using Terraform 0.10.0 or later can also load default values from any *.auto.tfvars files in the configuration. You may want to use the Terraform Cloud Provider or the variables API to add multiple variables at once.
Sensitive variables
Sensitive variables are never shown in the UI or API, and can't be edited. They may appear in Terraform logs if your configuration is designed to output them. To change a sensitive variable, delete and replace it.
Workspace variables (4)
Variables defined within a workspace always overwrite variables from variable sets that have the same type and the same key. Learn more about variable set precedence.
Key
Value
Category
instance_name
Updated by Terraform
terraform
AWS_SECRET_ACCESS_KEY
SENSITIVE
Sensitive - write only
env
AWS_ACCESS_KEY_ID
SENSITIVE
Sensitive - write only
env
instance_type
t2.micro
terraform
Add variable
Variable sets (0)

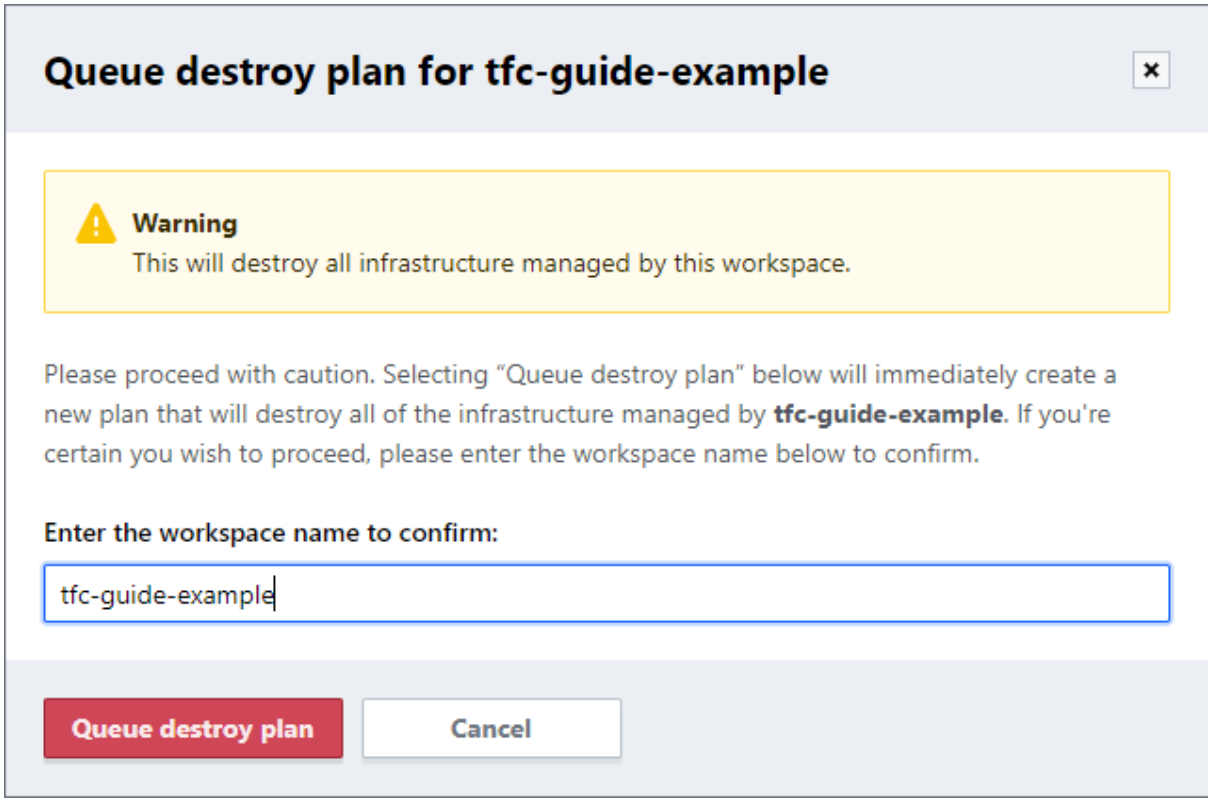

Applied
Triggered via UI
CURRENT
Dhotre12 triggered a run from UI a few seconds ago
Run Details
Plan finished a minute ago
Resources: 0 to add, 1 to change, 0 to destroy
Started 2 minutes ago
Finished a minute ago
1 to change
Terraform 1.3.1
Download raw log
aws_instance.ubuntu
id :
"i-0c608834b7707c93b"
tags :
Name :
"Provisioned by Terraform" → "Updated by Terraform"
31 unchanged attributes hidden
4 unchanged blocks hidden
data.aws_ami.ubuntu
Download Sentinel mocks
Sentinel mocks can be used for testing your Sentinel policies
Apply finished a few seconds ago
Resources: 0 added, 1 changed, 0 destroyed
Started 4 minutes ago
Finished 3 minutes ago
1 changed
Terraform 1.3.1
Download raw log

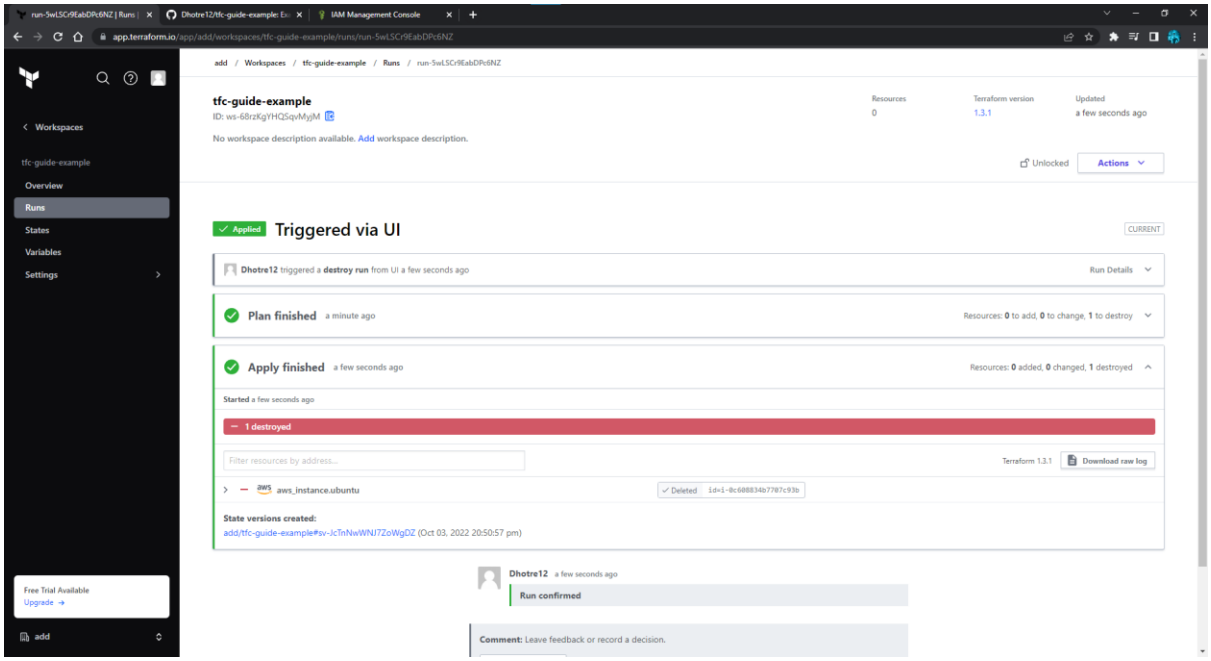
Step 6:

Destroy Resources and Workspaces

Now that you have provisioned and changed infrastructure with Terraform Cloud,the final stage of your infrastructure's lifecycle is to destroy it. Terraform Cloud allows you to destroy the infrastructure you have provisioned as a part of the standard workflow.







Conclusion: Hence we have studied to Build, change, and destroy AWS / GCP /Microsoft Azure/ DigitalOcean infrastructure Using Terraform and learned to implement them.

Questionnaire:

1. What is Terraform init?

Ans. Terraform init is a control to initialize an operational index that contains Terraform pattern files. This control can be looped multiple times. It is the first command that should be run after writing the new Terraform design.

2. What is Terraform D?

Ans. Terraform D is a plugin used on most in-service systems and Windows. Terraform init by default searches next directories for plugins.

3. Is history the same as it is on the web while using TFS API to provide resources?

Ans. Yes, the narration is similar to on the web because UI keeps API as the base. The whole thing that is on the UI is availed during other methods and the API.

4. Why is Terraform used for DevOps?

Ans. Terraform uses a JSON-like configuration language called the HashiCorp Configuration Language (HCL). HCL has a very simple syntax that makes it easy for DevOps teams to define and enforce infrastructure configurations across multiple clouds and on-premises data centers.

5. Define null resource in Terraform.

Ans. null_resource implements standard resource library, but no further action is taken. The triggers argument allows an arbitrary set of values that will cause the replacement of resources when changed.

Experiment 7

Aim: To understand Static Analysis SAST process and learn to integrate Jenkins SAST to SonarQube/GitLab.

Theory:

Static Code Analysis

- Static analysis,also called static code analysis, is a method of computer program debugging that is done by examining the code without executing the program. The process provides an understanding of the code structure,and can help to ensure that the code adheres to industry standards.
- Automated tools can assist programmers and developers in carrying out static analysis. The process of scrutinizing code by visual inspection alone(by looking at a printout,for example),without the assistance of automated tools, is sometimes called program understanding or program comprehension.

SonarQube

- It is a static testing open source tool developed by SonarSource for continuous inspection of code quality, perform automatic detection of static analysis of code to detect bugs, code smells,and security vulnerabilities on 25+ programming languages.SAST- Static Application Security Testing.
- SonarQube offers reports on duplicated code, coding standards,unit tests, code coverage, code complexity, comments, bugs, and security vulnerabilities.

Sonar Scanner

- Sonar Scanner is a separate client type application that in connection with the SonarQube server will run project analysis and then send the results to the SonarQube server to process it.
- Sonar Scanner can handle most programming languages supported by SonarQube except C#and VB.

Technical Terms

- Quality Gate: Default or custom
- Reliability/Bugs: Error or Bugs in the code
- Security/Vulnerabilities: Security bugs or vulnerabilities detection
- Maintainability /Code Smells: Bad smell in code
- Coverage: coverage of the unit /integration test cases
- Duplications: duplicate lines of code/function/block/files
- Size:lines/classes /comments/files on code.
- Languages: programming languages
- Complexity:difficult to understand

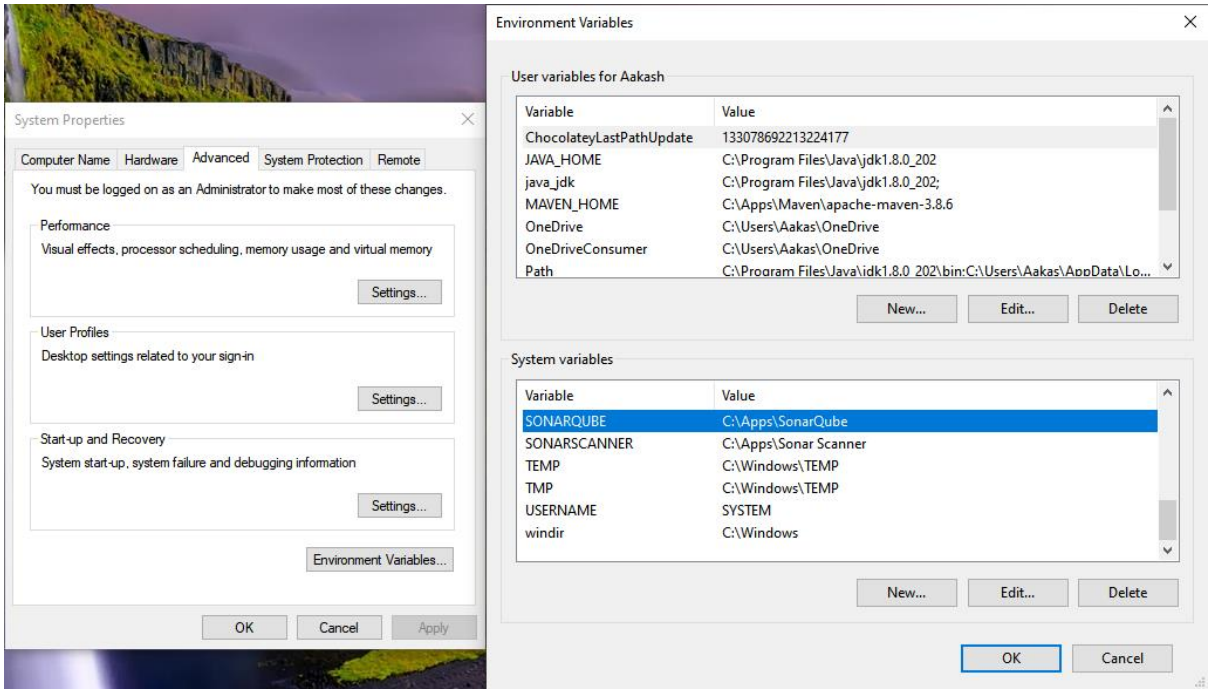
Projects

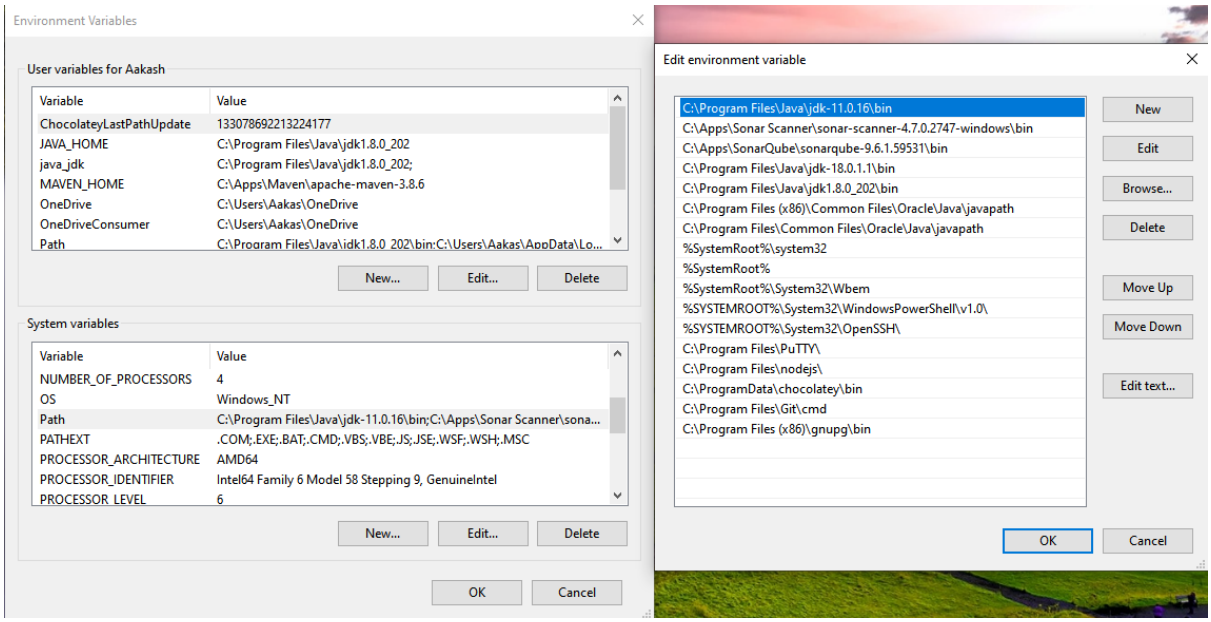
1) Ant Project
2) Maven Project
3) Gradle Project
4) NodeJS Project
5) Python Project

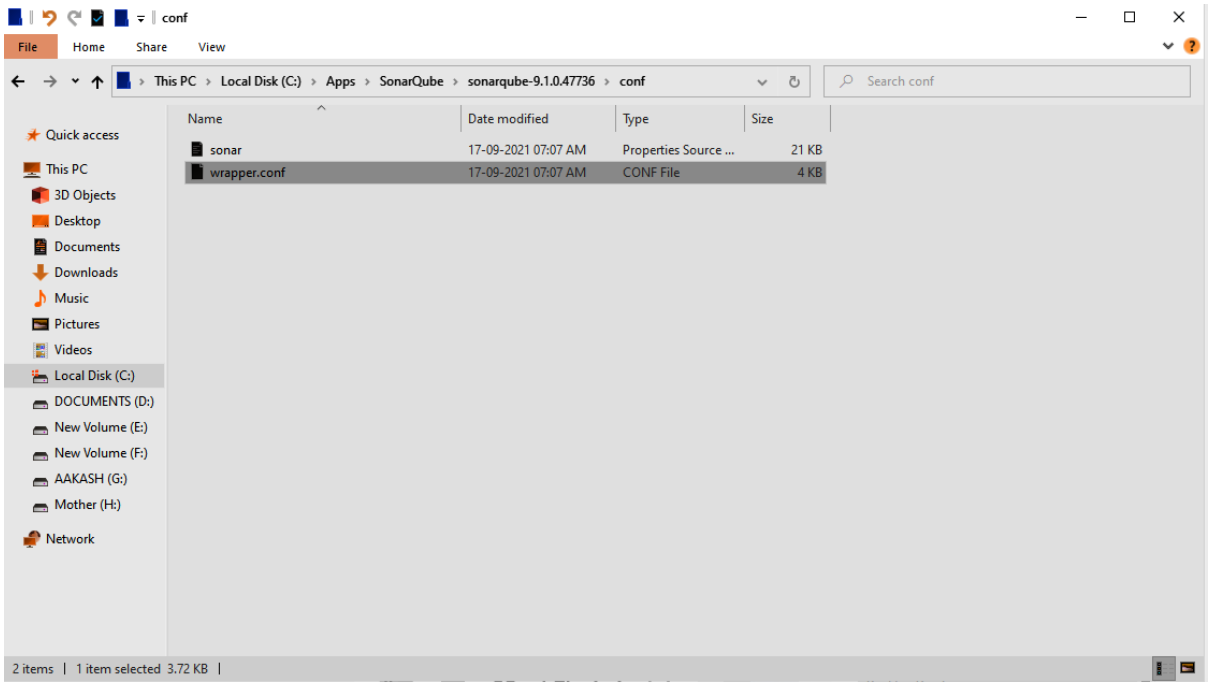
Prerequisite:

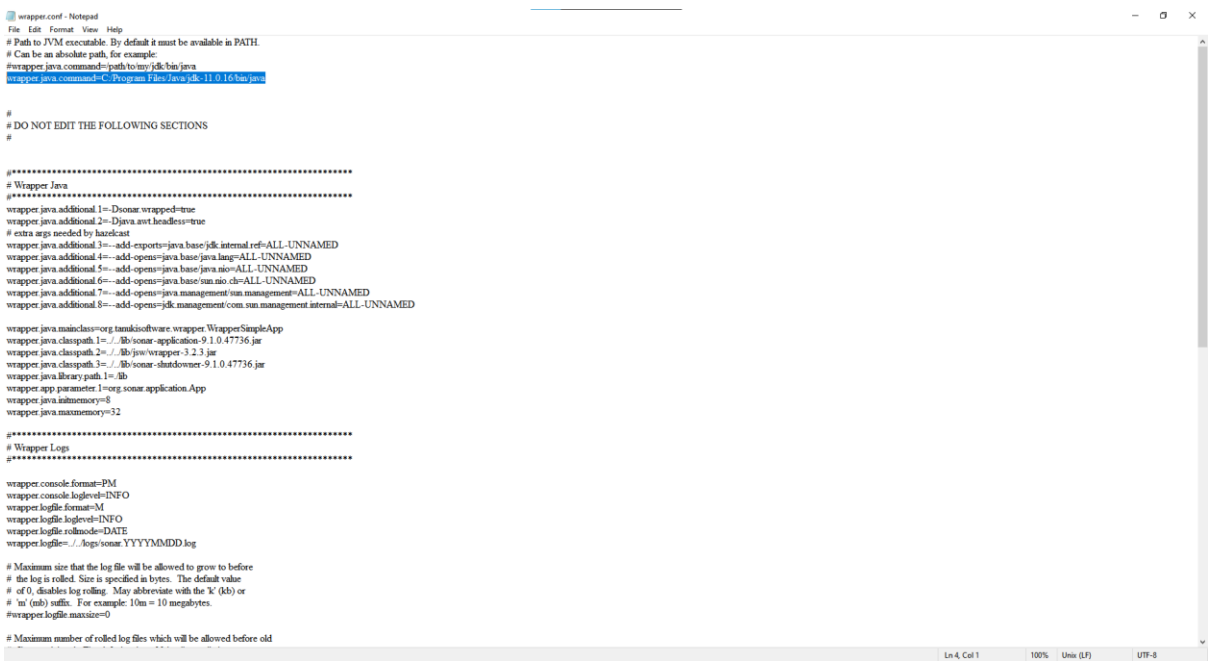
1. JAVA 1.11 version or higher version
2. SonarQube latest version
3. Sonar Scanner latest version
4. Set path for SonarQube and Sonar Scanner

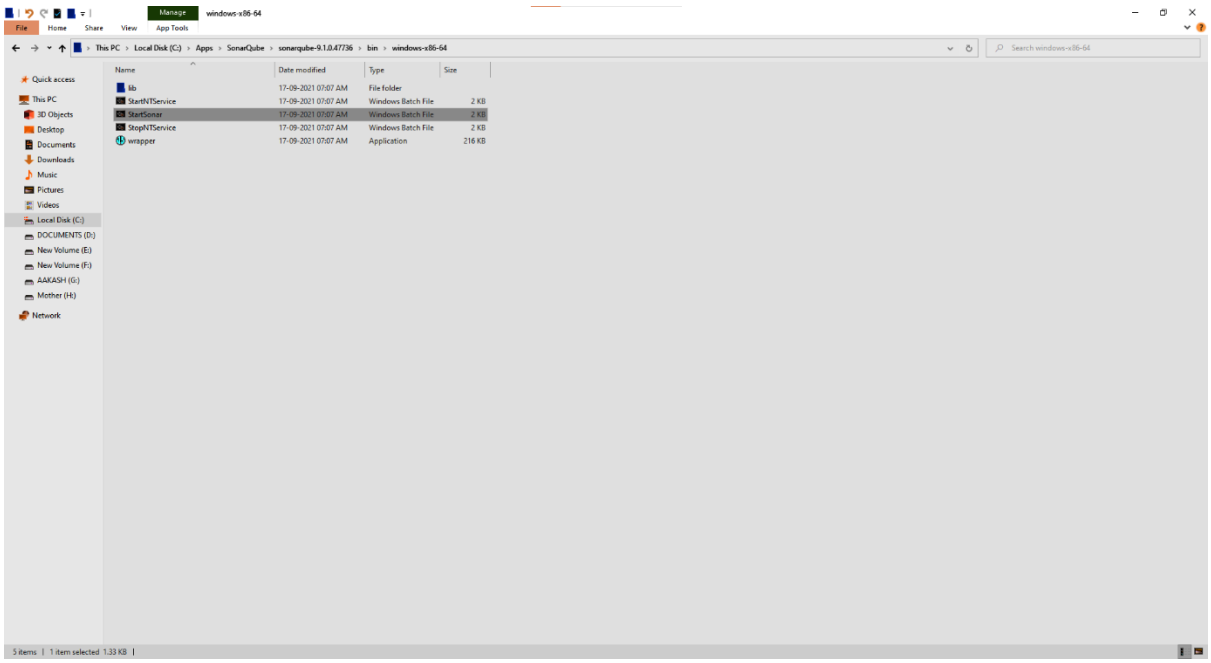
Result:



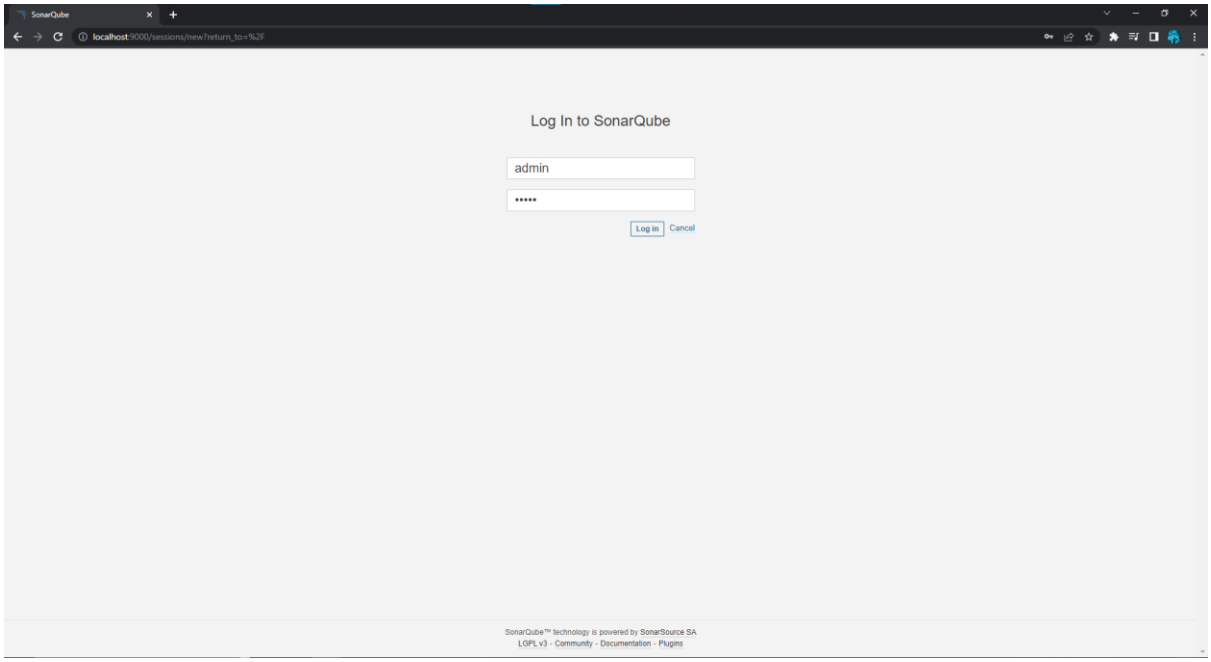


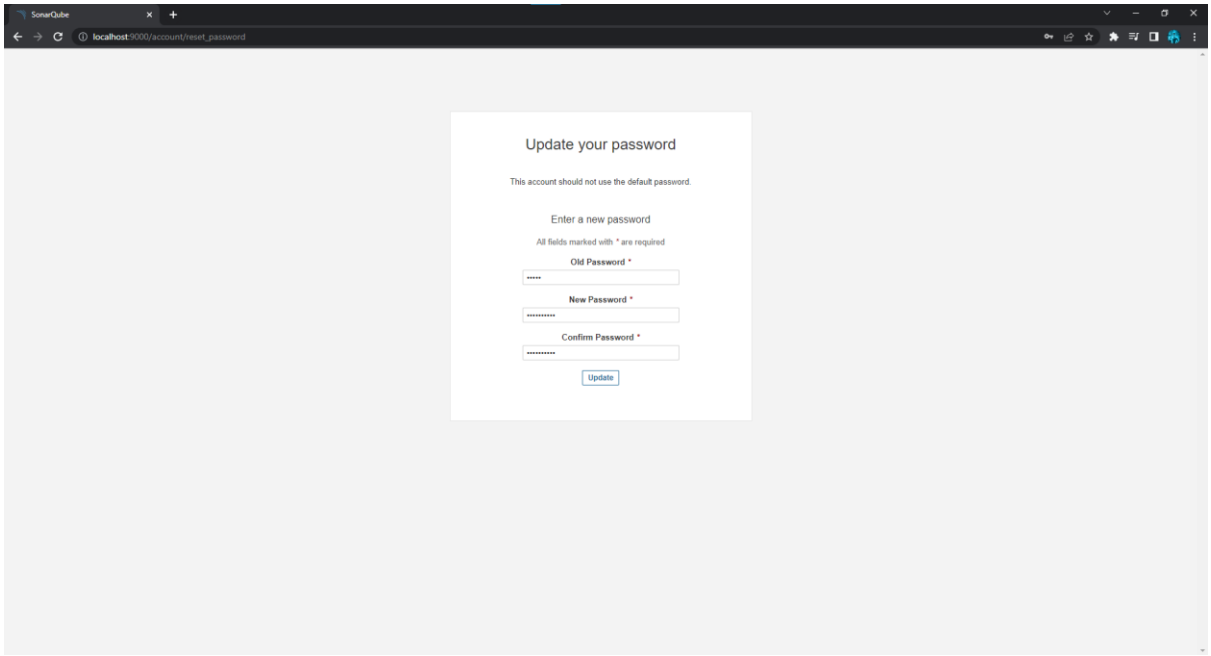

conf
File
Home
Share
View
This PC
Local Disk (C:)
Apps
SonarQube
sonarqube-9.1.0.47736
conf
Search conf
Quick access
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Local Disk (C:)
DOCUMENTS (D:)
New Volume (E:)
New Volume (F:)
AAKASH (G:)
Mother (H:)
Network
Name
Date modified
Type
Size
sonar
17-09-2021 07:07 AM
Properties Source ...
21 KB
wrapper.conf
17-09-2021 07:07 AM
CONF File
4 KB
2 items | 1 item selected 3.72 KB

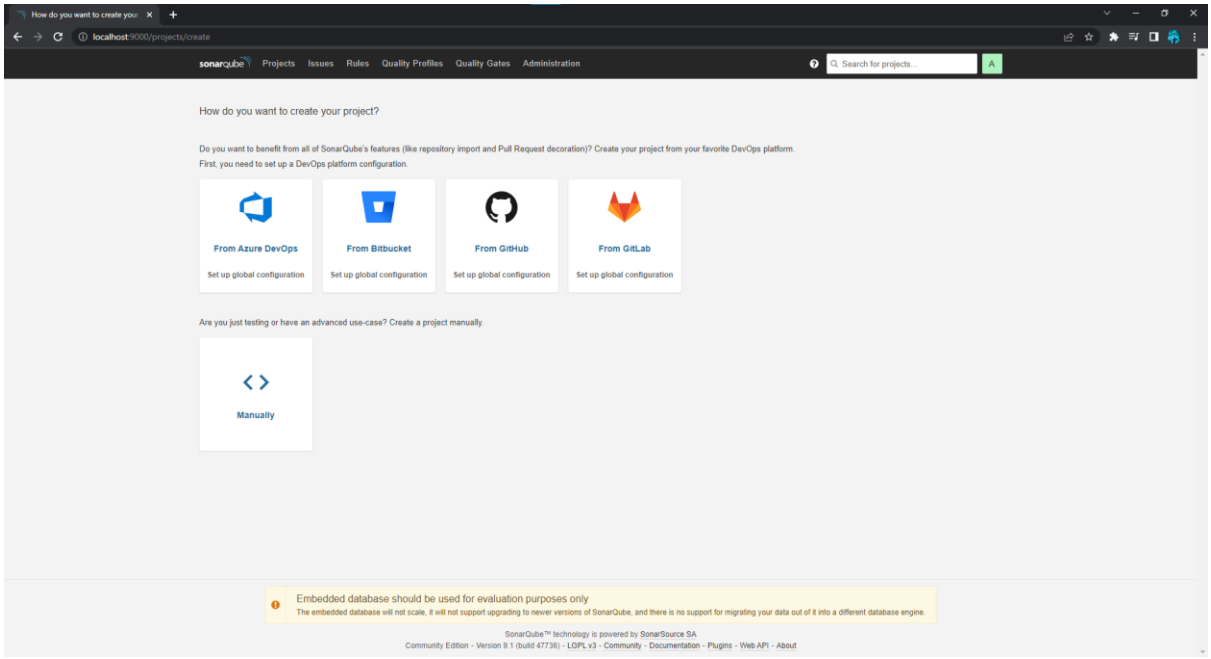

wrapper.conf - Notepad
File Edit Format View Help
Path to JVM executable. By default it must be available in PATH.
Can be an absolute path, for example:
#wrapper.java.command=/path/to/my/jdk/bin/java
wrapper.java.command=C:/Program Files/Java/jdk-11.0.16/bin/java
#
DO NOT EDIT THE FOLLOWING SECTIONS
#
#***
Wrapper Java
#***
wrapper.java.additional.1=-Dsonar.wrapped=true
wrapper.java.additional.2=-Djava.awt.headless=true
# extra args needed by hazelcast
wrapper.java.additional.3=--add-exports=java.base/jdk.internal.ref=ALL-UNNAMED
wrapper.java.additional.4=--add-opens=java.base/java.lang=ALL-UNNAMED
wrapper.java.additional.5=--add-opens=java.base/java.nio=ALL-UNNAMED
wrapper.java.additional.6=--add-opens=java.base/sun.nio.ch=ALL-UNNAMED
wrapper.java.additional.7=--add-opens=java.management/sun.management=ALL-UNNAMED
wrapper.java.additional.8=--add-opens=jdk.management/com.sun.management.internal=ALL-UNNAMED
wrapper.java.mainclass=org.tanukisoftware.wrapper.WrapperSimpleApp
wrapper.java.classpath.1=../../lib/sonar-application-9.1.0.47736.jar
wrapper.java.classpath.2=../../lib/jsw/wrapper-3.2.3.jar
wrapper.java.classpath.3=../../lib/sonar-shutdowner-9.1.0.47736.jar
wrapper.java.library.path.1=../lib
wrapper.app.parameter.1=org.sonar.application.App
wrapper.java.initmemory=8
wrapper.java.maxmemory=32
#***
Wrapper Logs
#***
wrapper.console.format=PM
wrapper.console.loglevel=INFO
wrapper.logfile.format=M
wrapper.logfile.loglevel=INFO
wrapper.logfile.rollmode=DATE
wrapper.logfile=../../logs/sonar.YYYYMMDD.log
# Maximum size that the log file will be allowed to grow to before
# the log is rolled. Size is specified in bytes. The default value
# of 0, disables log rolling. May abbreviate with the 'k' (kb) or
# 'm' (mb) suffix. For example: 10m = 10 megabytes.
#wrapper.logfile.maxsize=0
Maximum number of rolled log files which will be allowed before old
Ln 4, Col 1
100%
Unix (LF)
UTF-8

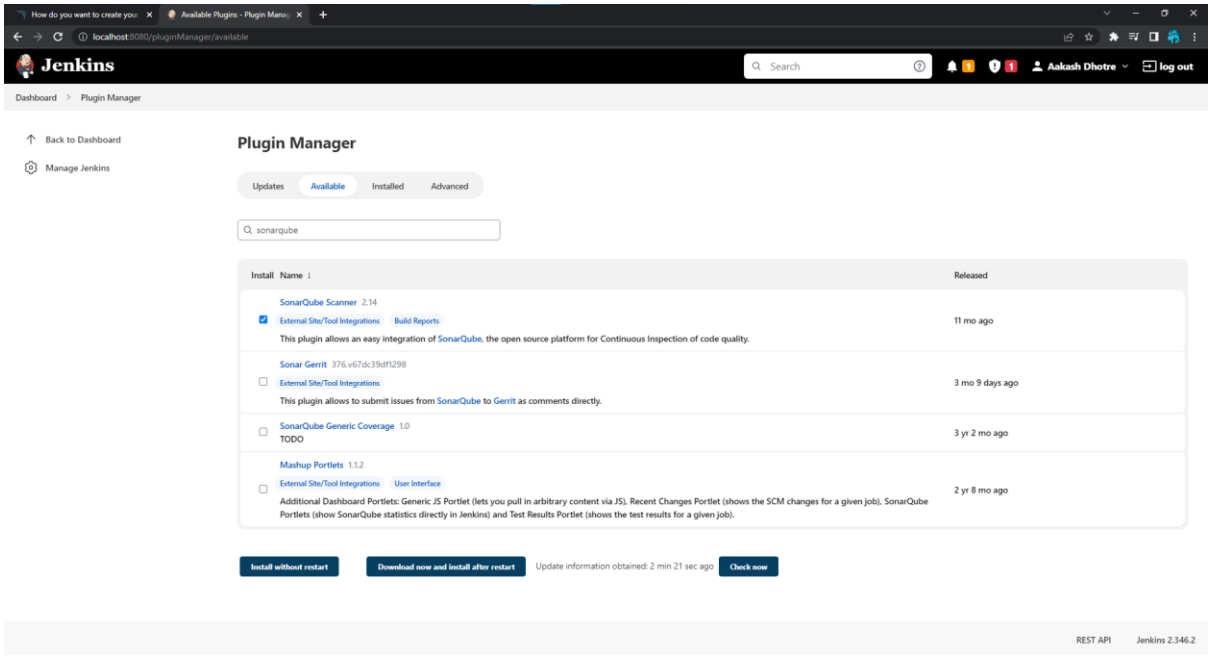

Manage
windows-x86-64
File
Home
Share
View
App Tools
This PC › Local Disk (C:) › Apps › SonarQube › sonarqube-9.1.0.47736 › bin › windows-x86-64
Search windows-x86-64
Quick access
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Local Disk (C:)
DOCUMENTS (D:)
New Volume (E:)
New Volume (F:)
AAKASH (G:)
Mother (H:)
Network
Name
Date modified
Type
Size
lib
17-09-2021 07:07 AM
File folder
StartNTService
17-09-2021 07:07 AM
Windows Batch File
2 KB
StartSonar
17-09-2021 07:07 AM
Windows Batch File
2 KB
StopNTService
17-09-2021 07:07 AM
Windows Batch File
2 KB
wrapper
17-09-2021 07:07 AM
Application
216 KB
5 items | 1 item selected 1.33 KB

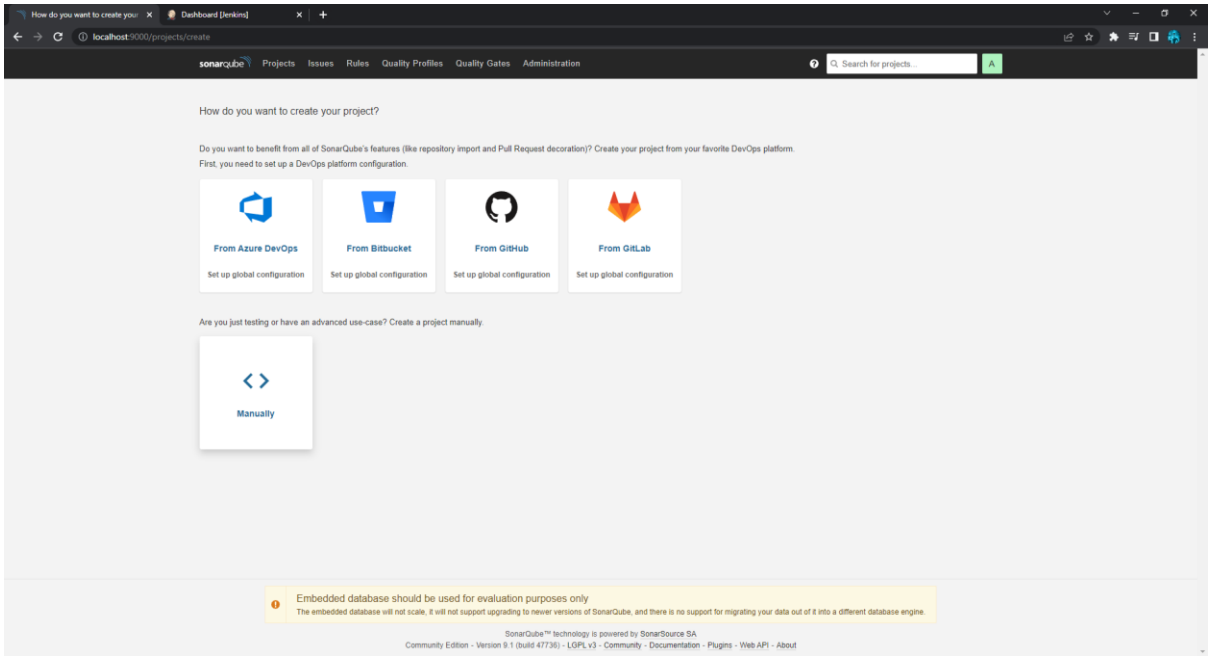

SonarQube
wrapper | Launching a JVM...
jvm 1 | Wrapper (Version 3.2.3) http://wrapper.tanukisoftware.org
jvm 1 | Copyright 1999-2006 Tanuki Software, Inc. All Rights Reserved.
jvm 1 |
jvm 1 | 2022.10.12 18:20:38 INFO app[][o.s.a.AppFileSystem] Cleaning or creating temp directory C:\Apps\SonarQube\sonarqube-9.1.0.47736\temp
jvm 1 | 2022.10.12 18:20:38 INFO app[][o.s.a.es.EsSettings] Elasticsearch listening on [HTTP: 127.0.0.1:9001, TCP: 127.0.0.1:1025]
jvm 1 | 2022.10.12 18:20:38 INFO app[][o.s.a.ProcessLauncherImpl] Launch process[[key='es', ipcIndex=1, logFilenamePrefix=es]] from [C:\Apps\SonarQube\sonarqube-9.1.0.47736\elasticsearch]: C:\Program Files\Java\jdk-11.0.16\bin\java -XX:+UseG1GC -Djava.io.tmpdir=C:\Apps\SonarQube\sonarqube-9.1.0.47736\temp -XX:ErrorFile=../logs/es_hs_err_pid%p.log -Des.networkaddress.cache.ttl=60 -Des.networkaddress.cache.negative.ttl=10 -XX:+AlwaysPreTouch -Xss1m -Djava.awt.headless=true -Dfile.encoding=UTF-8 -Djna.nosys=true -XX:-OmitStackTraceInFastThrow -Dio.netty.noUnsafe=true -Dio.netty.noKeySetOptimization=true -Dio.netty.recycler.maxCapacityPerThread=0 -Dio.netty.allocator.numDirectArenas=0 -Dlog4j.shutdownHookEnabled=false -Dlog4j2.disable.jmx=true -Djava.locale.providers=COMPAT -Xmx512m -Xms512m -XX:MaxDirectMemorySize=256m -XX:+HeapDumpOnOutOfMemoryError -Delasticsearch -Des.path.home=C:\Apps\SonarQube\sonarqube-9.1.0.47736\elasticsearch -Des.path.conf=C:\Apps\SonarQube\sonarqube-9.1.0.47736\temp\conf\es -cp lib/* org.elasticsearch.bootstrap.Elasticsearch
jvm 1 | 2022.10.12 18:20:39 INFO app[][o.s.a.SchedulerImpl] Waiting for Elasticsearch to be up and running
jvm 1 | 2022.10.12 18:21:00 INFO app[][o.s.a.SchedulerImpl] Process[es] is up
jvm 1 | 2022.10.12 18:21:00 INFO app[][o.s.a.ProcessLauncherImpl] Launch process[[key='web', ipcIndex=2, logFilenamePrefix=web]] from [C:\Apps\SonarQube\sonarqube-9.1.0.47736]: C:\Program Files\Java\jdk-11.0.16\bin\java -Djava.awt.headless=true -Dfile.encoding=UTF-8 -Djava.io.tmpdir=C:\Apps\SonarQube\sonarqube-9.1.0.47736\temp -XX:-OmitStackTraceInFastThrow --add-opens=java.base/java.util=ALL-UNNAMED --add-opens=java.base/java.lang=ALL-UNNAMED --add-opens=java.base/java.io=ALL-UNNAMED --add-opens=java.rmi/sun.rmi.transport=ALL-UNNAMED --add-exports=java.base/jdk.internal.ref=ALL-UNNAMED --add-opens=java.base/java.nio=ALL-UNNAMED --add-opens=java.base/sun.nio.ch=ALL-UNNAMED --add-opens=java.management/sun.management=ALL-UNNAMED --add-opens=jdk.management/com.sun.management.internal=ALL-UNNAMED -Xmx512m -Xms128m -XX:+HeapDumpOnOutOfMemoryError -Dhttp.nonProxyHosts=localhost|127.*|[::1] -cp ./lib/sonar-application-9.1.0.47736.jar;C:\Apps\SonarQube\sonarqube-9.1.0.47736\lib\jdbc\h2\h2-1.4.199.jar org.sonar.server.app.WebServer C:\Apps\SonarQube\sonarqube-9.1.0.47736\temp\sq-process12389756060030717 72properties
jvm 1 | 2022.10.12 18:22:29 INFO app[][o.s.a.SchedulerImpl] Process[web] is up
jvm 1 | 2022.10.12 18:22:29 INFO app[][o.s.a.ProcessLauncherImpl] Launch process[[key='ce', ipcIndex=3, logFilenamePrefix=ce]] from [C:\Apps\SonarQube\sonarqube-9.1.0.47736]: C:\Program Files\Java\jdk-11.0.16\bin\java -Djava.awt.headless=true -Dfile.encoding=UTF-8 -Djava.io.tmpdir=C:\Apps\SonarQube\sonarqube-9.1.0.47736\temp -XX:-OmitStackTraceInFastThrow --add-opens=java.base/java.util=ALL-UNNAMED --add-exports=java.base/jdk.internal.ref=ALL-UNNAMED --add-opens=java.base/java.lang=ALL-UNNAMED --add-opens=java.base/java.nio=ALL-UNNAMED --add-opens=java.base/sun.nio.ch=ALL-UNNAMED --add-opens=java.management/sun.management=ALL-UNNAMED --add-opens=jdk.management/com.sun.management.internal=ALL-UNNAMED -Xmx512m -Xms128m -XX:+HeapDumpOnOutOfMemoryError -Dhttp.nonProxyHosts=localhost|127.*|[::1] -cp ./lib/sonar-application-9.1.0.47736.jar;C:\Apps\SonarQube\sonarqube-9.1.0.47736\lib\jdbc\h2\h2-1.4.199.jar org.sonar.ce.app.CeServer C:\Apps\SonarQube\sonarqube-9.1.0.47736\temp\sq-process3700008610206978113properties
jvm 1 | 2022.10.12 18:22:29 WARN app[][startup] ##
jvm 1 | 2022.10.12 18:22:29 WARN app[][startup] Default Administrator credentials are still being used. Make sure to change the password or deactivate the account.
jvm 1 | 2022.10.12 18:22:29 WARN app[][startup] ##
jvm 1 | 2022.10.12 18:22:38 INFO app[][o.s.a.SchedulerImpl] Process[ce] is up
jvm 1 | 2022.10.12 18:22:38 INFO app[][o.s.a.SchedulerImpl] SonarQube is up

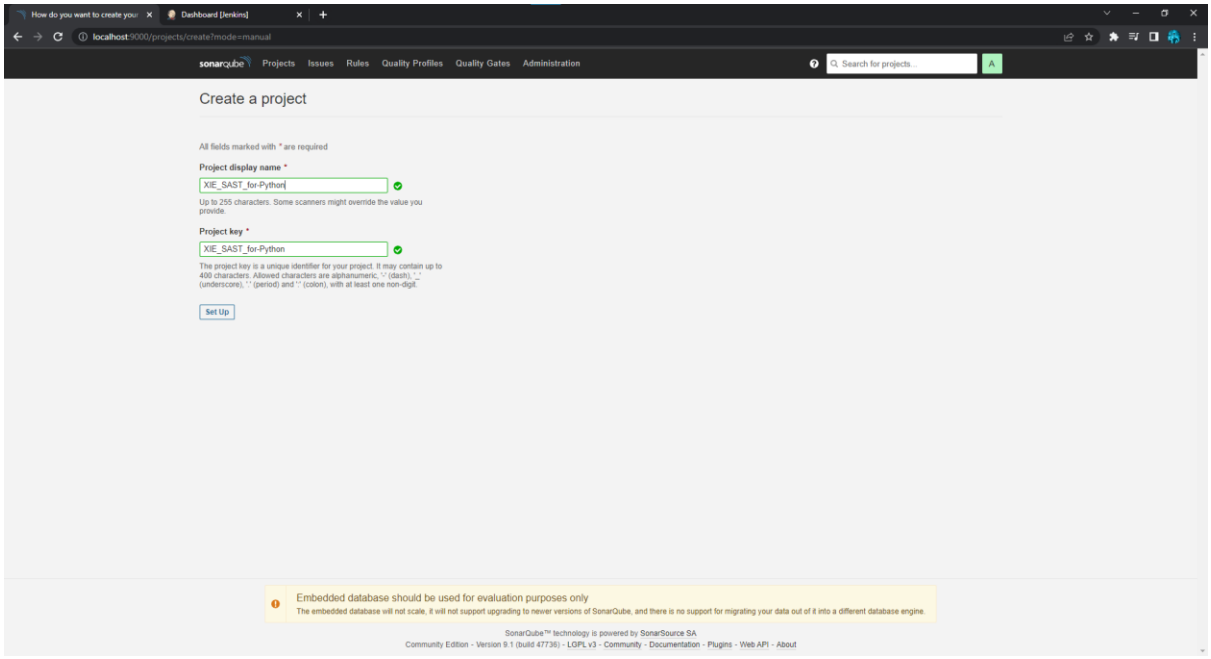

SonarQube
localhost:9000/sessions/new?return_to=%2F
Log In to SonarQube
admin
Log in
Cancel
SonarQube™ technology is powered by SonarSource SA
LGPL v3 - Community - Documentation - Plugins


SonarQube
localhost:9000/account/reset_password
Update your password
This account should not use the default password.
Enter a new password
All fields marked with * are required
Old Password *
New Password *
Confirm Password *
Update


How do you want to create your
localhost:9000/projects/create
sonarqube
Projects
Issues
Rules
Quality Profiles
Quality Gates
Administration
Search for projects...
How do you want to create your project?
Do you want to benefit from all of SonarQube's features (like repository import and Pull Request decoration)? Create your project from your favorite DevOps platform.
First, you need to set up a DevOps platform configuration.
From Azure DevOps
Set up global configuration
From Bitbucket
Set up global configuration
From GitHub
Set up global configuration
From GitLab
Set up global configuration
Are you just testing or have an advanced use-case? Create a project manually.
Manually
Embedded database should be used for evaluation purposes only
The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.
SonarQube™ technology is powered by SonarSource SA
Community Edition - Version 9.1 (build 47736) - LGPL v3 - Community - Documentation - Plugins - Web API - About


Jenkins
Search
Aakash Dhotre
log out
Dashboard
Plugin Manager
Back to Dashboard
Manage Jenkins
Plugin Manager
Updates
Available
Installed
Advanced
sonarqube
Install
Name
Released
SonarQube Scanner 2.14
External Site/Tool Integrations
Build Reports
11 mo ago
This plugin allows an easy integration of SonarQube, the open source platform for Continuous Inspection of code quality.
Sonar Gerrit 376.v67dc39df1298
External Site/Tool Integrations
3 mo 9 days ago
This plugin allows to submit issues from SonarQube to Gerrit as comments directly.
SonarQube Generic Coverage 1.0
TODO
3 yr 2 mo ago
Mashup Portlets 1.1.2
External Site/Tool Integrations
User Interface
2 yr 8 mo ago
Additional Dashboard Portlets: Generic JS Portlet (lets you pull in arbitrary content via JS), Recent Changes Portlet (shows the SCM changes for a given job), SonarQube Portlets (show SonarQube statistics directly in Jenkins) and Test Results Portlet (shows the test results for a given job).
Install without restart
Download now and install after restart
Update information obtained: 2 min 21 sec ago
Check now
REST API
Jenkins 2.346.2


sonarqube
Projects
Issues
Rules
Quality Profiles
Quality Gates
Administration
Search for projects...
How do you want to create your project?
Do you want to benefit from all of SonarQube's features (like repository import and Pull Request decoration)? Create your project from your favorite DevOps platform.
First, you need to set up a DevOps platform configuration.
From Azure DevOps
Set up global configuration
From Bitbucket
Set up global configuration
From GitHub
Set up global configuration
From GitLab
Set up global configuration
Are you just testing or have an advanced use-case? Create a project manually.
Manually
Embedded database should be used for evaluation purposes only
The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.
SonarQube™ technology is powered by SonarSource SA
Community Edition - Version 9.1 (build 47736) - LGPL v3 - Community - Documentation - Plugins - Web API - About



Conclusion: Hence, we have studied to understand Static Analysis SAST process and learn to integrate Jenkins SAST to SonarQube/GitLab.

Questionnaire:

Q 1) What is Security Testing?

Answer: Security testing can be considered as the most important in all types of software testing. Its main objective is to find vulnerabilities in any software (web or networking) based application and protect their data from possible attacks or intruders.
As many applications contain confidential data and need to be protected from being leaked. Software testing needs to be done periodically on such applications to identify threats and to take immediate action on them.

Q 2) What is "Vulnerability"?

Answer: Vulnerability can be defined as the weakness of any system through which intruders or bugs can attack the system.
If security testing has not been performed rigorously on the system, then chances of vulnerabilities get increased. Time to time patches or fixes is required to prevent a system from the vulnerabilities.

Q 3) What is Intrusion Detection?

Answer: Intrusion detection is a system which helps in determining possible attacks and deal with it. Intrusion detection includes collecting information from many systems and sources, analysis of the information and finding the possible ways of the attack on the system.
Intrusion detection checks the following:

- Possible attacks
- Any abnormal activity
- Auditing the system data
- Analysis of different collected data, etc.

Q 4) List the attributes of Security Testing?

Answer: There are following seven attributes of Security Testing:
Authentication, Authorization, Confidentiality, Availability, Integrity, Non-repudiation and Resilience.

Experiment 8

Aim: Create a Jenkins CICD Pipeline with SonarQube / GitLab Integration to perform a static analysis of the code to detect bugs, code smells, and security vulnerabilities on a sample Web / Java / Python application.

Theory:
Integrating Jenkins with SonarQube provides you with an automated platform for performing continuous inspection of code for quality and security assurance.

In this lab, you will launch a Jenkins and SonarQube CICD environment using Docker containers on a provided EC2 instance. You will then configure a Jenkins build pipeline to build, compile, and package a sample Java servlet web application. The build pipeline will publish the source code into SonarQube, which in turn will perform a static analysis of the code to detect bugs, code smells, and security vulnerabilities.

This lab is aimed at DevOps and CICD practitioners, and, in particular, build and release engineers interested in managing and configuring Jenkins together with SonarQube to perform automated static code analysis.

This lab will start with the following AWS resources being provisioned automatically for you:

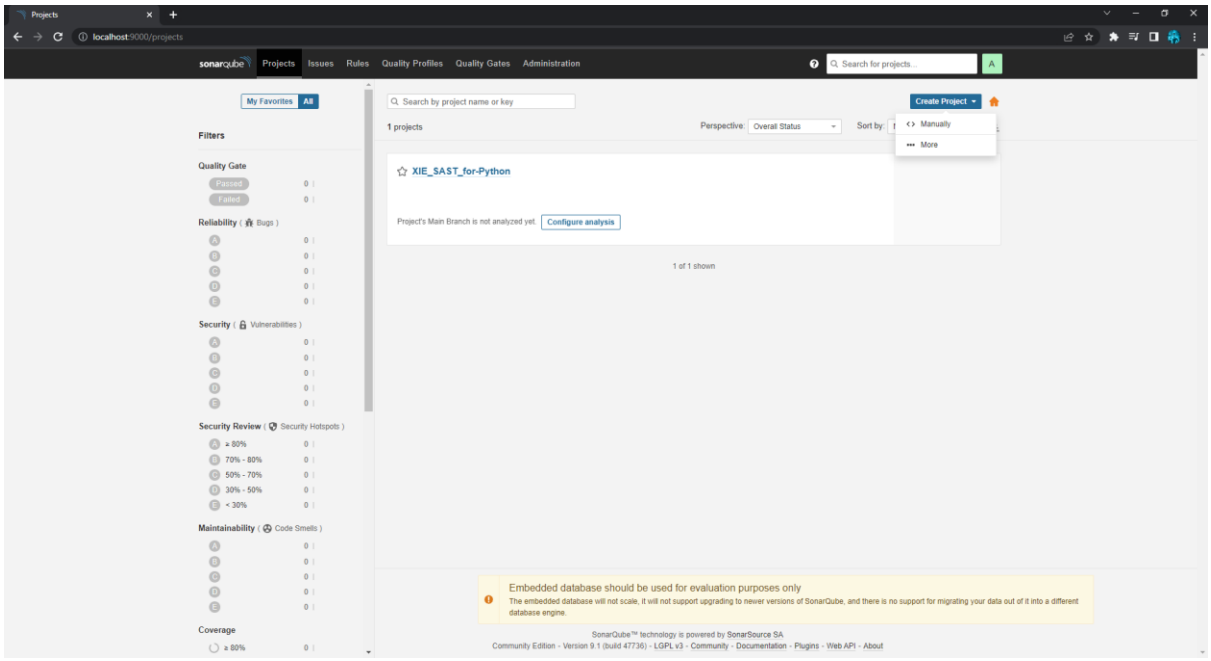
- A single EC2 instance, named cicd.platform.instance, which will have a public IP address attached

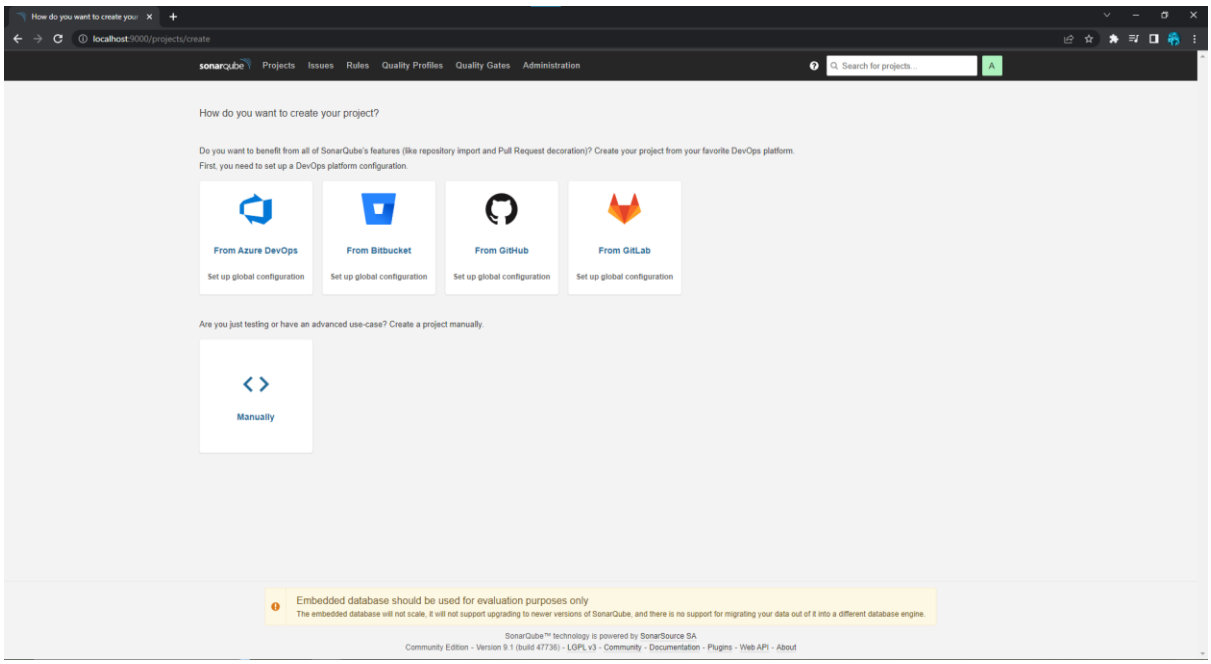
To achieve the Lab end state, you will be walked through the process of:

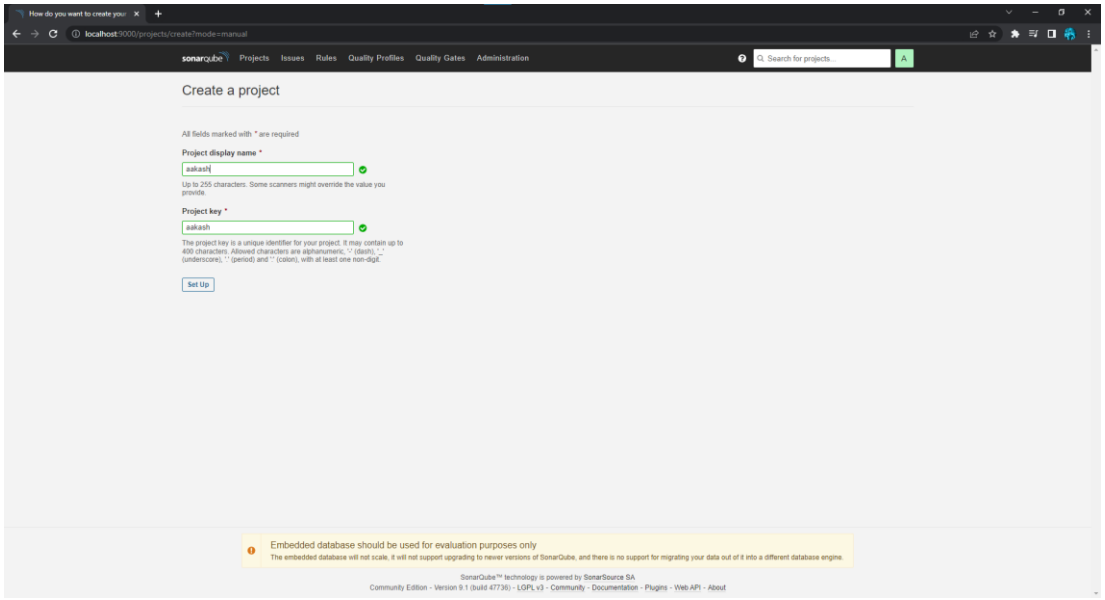
- SSHing into the EC2 instance, named cicd.platform.instance
    - Use Docker Compose to launch the following Docker containers:
        - Jenkins
        - SonarQube
        - Postgres
        - Socat
- Using a browser, administer and configure Jenkins - installing the required plugins. Connectivity to Jenkins will be done via the cicd.platform.instance Public IP address
- Using a browser, administer and configure SonarQube. Connectivity to SonarQube will be done via the cicd.platform.instance Public IP address
- Create a Jenkins build pipeline and configure it to build a sample Java servlet web application hosted on GitHub, with the source code later being forwarded into SonarQube for static code analysis
- Execute the Jenkins build pipeline and confirm that it has completed successfully, forwarding the source code over to SonarQube for static code analysis
- Confirm that SonarQube has received and performed static code analysis and generated a project report.

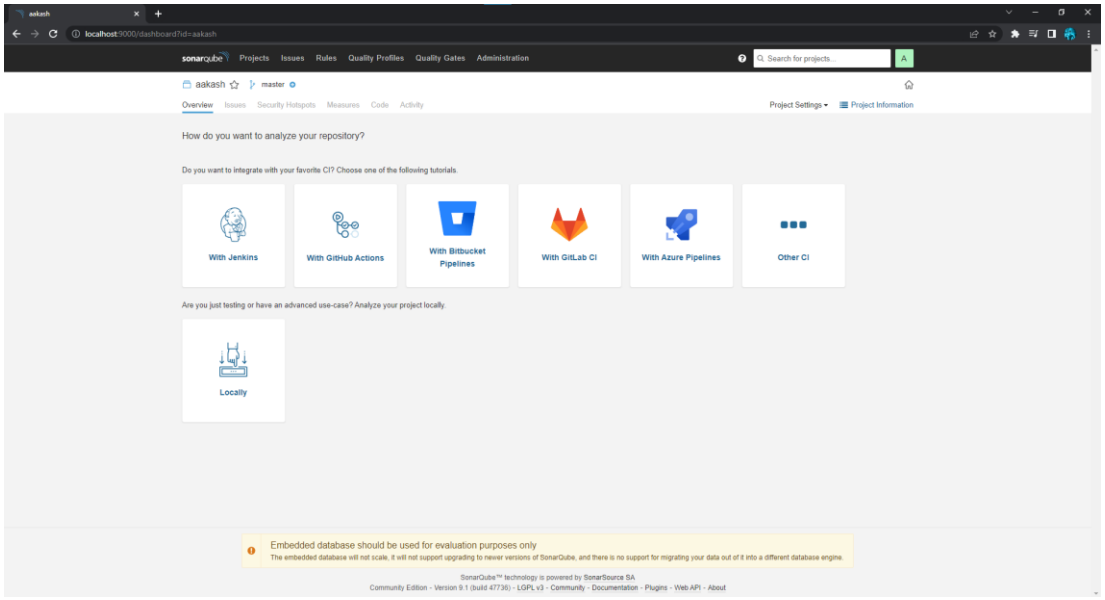
Result:

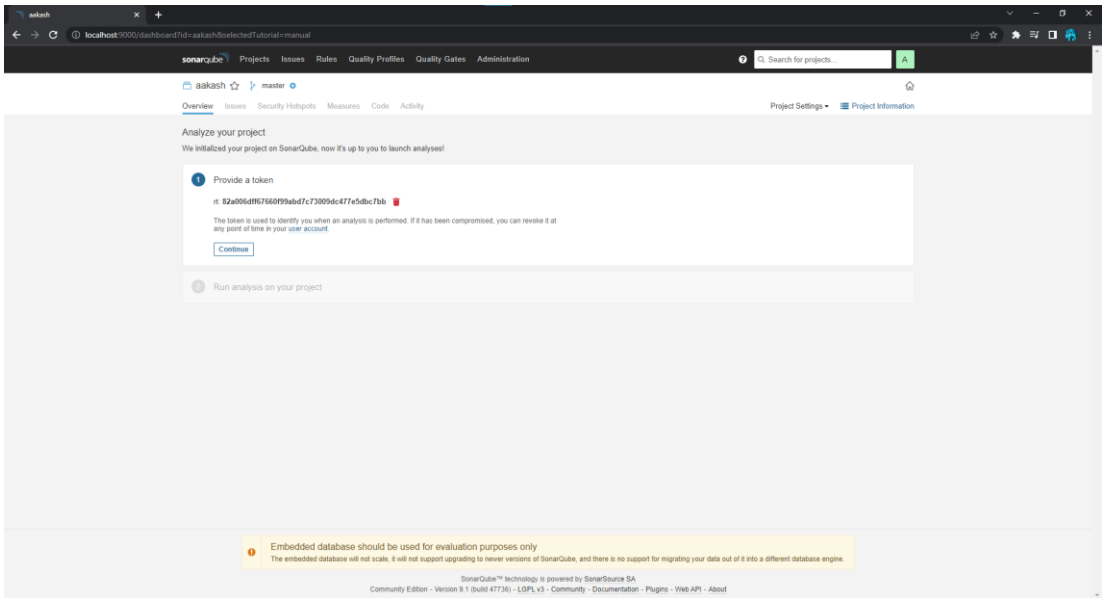
1. Start sonarqube
2. Create a folder in c drive name it as XieJob. Inside it create one more folder and name it as Py Scripts
3. Go to sonarqube dashboard and click on new project
4. Generate token and click on continue
5. Open sonar scanner folder
6. Open conf folder
7. Open sonar-scanner properties file and fill the details and save it.
8. Open cmd and paste the path
9. You will see details of project on dashboard

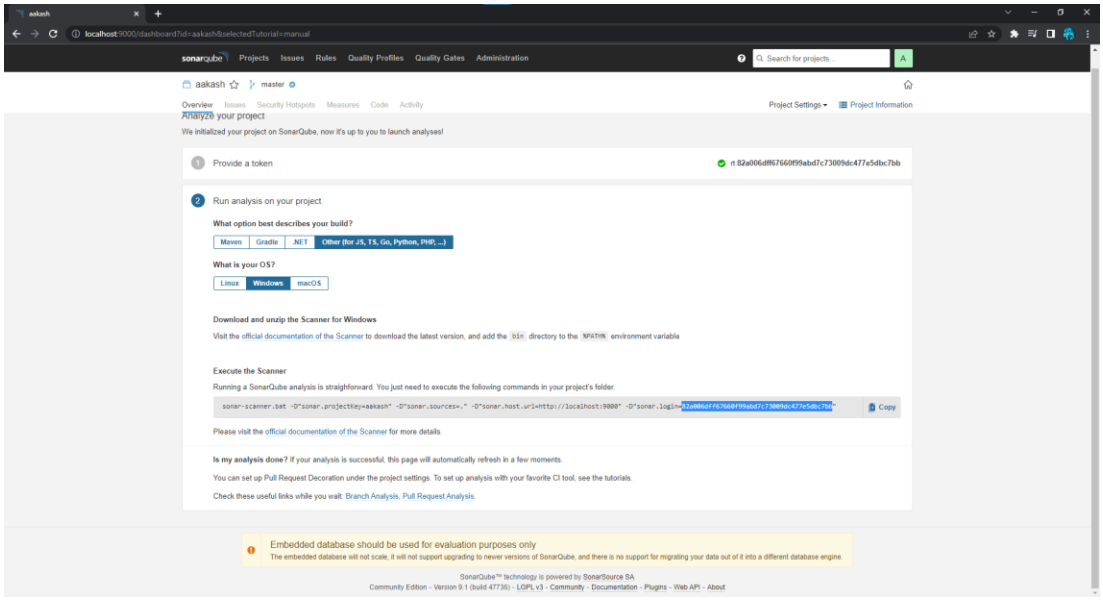





How do you want to create your
localhost:9000/projects/create?mode=manual
sonarqube
Projects
Issues
Rules
Quality Profiles
Quality Gates
Administration
Search for projects...
A
Create a project
All fields marked with * are required
Project display name *
aakash
Up to 255 characters. Some scanners might override the value you provide.
Project key *
aakash
The project key is a unique identifier for your project. It may contain up to 400 characters. Allowed characters are alphanumeric, '-' (dash), '_' (underscore), '.' (period) and ':' (colon), with at least one non-digit.
Set Up
Embedded database should be used for evaluation purposes only
The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.
SonarQube™ technology is powered by SonarSource SA
Community Edition - Version 9.1 (build 47736) - LGPL v3 - Community - Documentation - Plugins - Web API - About


aakash
localhost:9000/dashboard?id=aakash
sonarqube
Projects
Issues
Rules
Quality Profiles
Quality Gates
Administration
Search for projects...
A
aakash
master
Overview
Issues
Security Hotspots
Measures
Code
Activity
Project Settings
Project Information
How do you want to analyze your repository?
Do you want to integrate with your favorite CI? Choose one of the following tutorials.
With Jenkins
With GitHub Actions
With Bitbucket Pipelines
With GitLab CI
With Azure Pipelines
Other CI
Are you just testing or have an advanced use-case? Analyze your project locally.
Locally
Embedded database should be used for evaluation purposes only
The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.
SonarQube™ technology is powered by SonarSource SA
Community Edition - Version 9.1 (build 47736) - LGPL v3 - Community - Documentation - Plugins - Web API - About


aakash
localhost:9000/dashboard?id=aakash&selectedTutorial=manual
sonarqube
Projects
Issues
Rules
Quality Profiles
Quality Gates
Administration
Search for projects...
A
aakash
master
Overview
Issues
Security Hotspots
Measures
Code
Activity
Project Settings
Project Information
Analyze your project
We initialized your project on SonarQube, now it's up to you to launch analyses!
1
Provide a token
82a006dff67660f99abd7c73009dc477e5dbc7bb
The token is used to identify you when an analysis is performed. If it has been compromised, you can revoke it at any point of time in your user account.
Continue
2
Run analysis on your project
Embedded database should be used for evaluation purposes only
The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.
SonarQube™ technology is powered by SonarSource SA
Community Edition - Version 9.1 (build 47736) - LGPL v3 - Community - Documentation - Plugins - Web API - About



sonar-scanner - Notepad

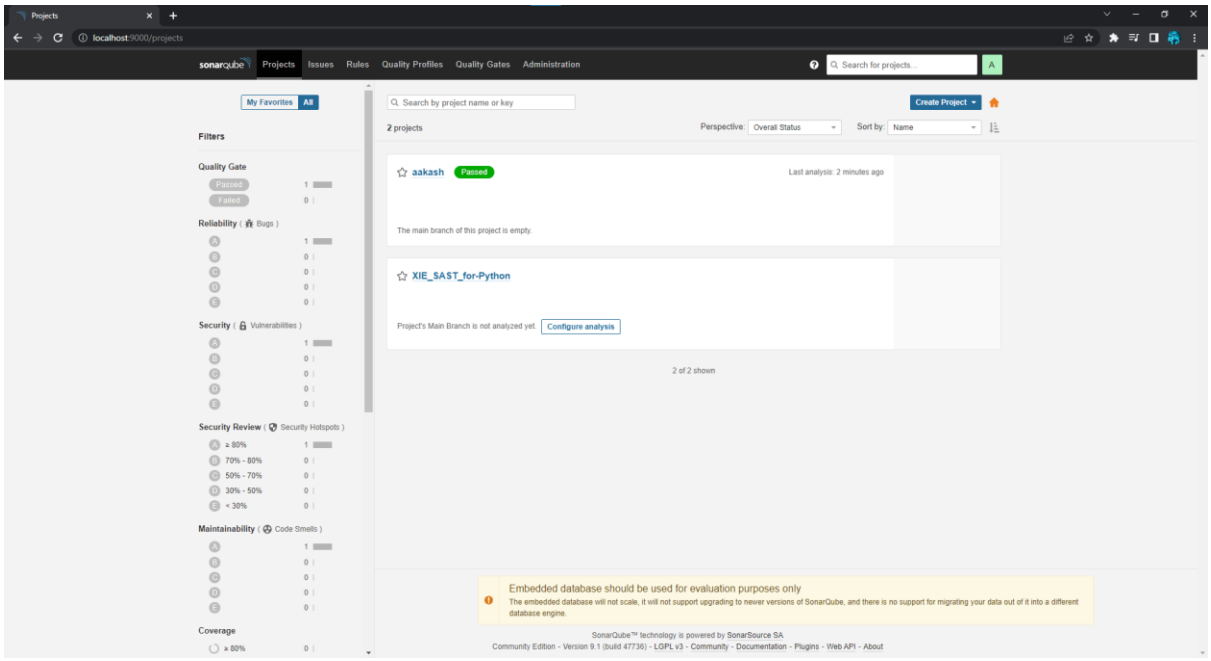
File Edit Format View Help

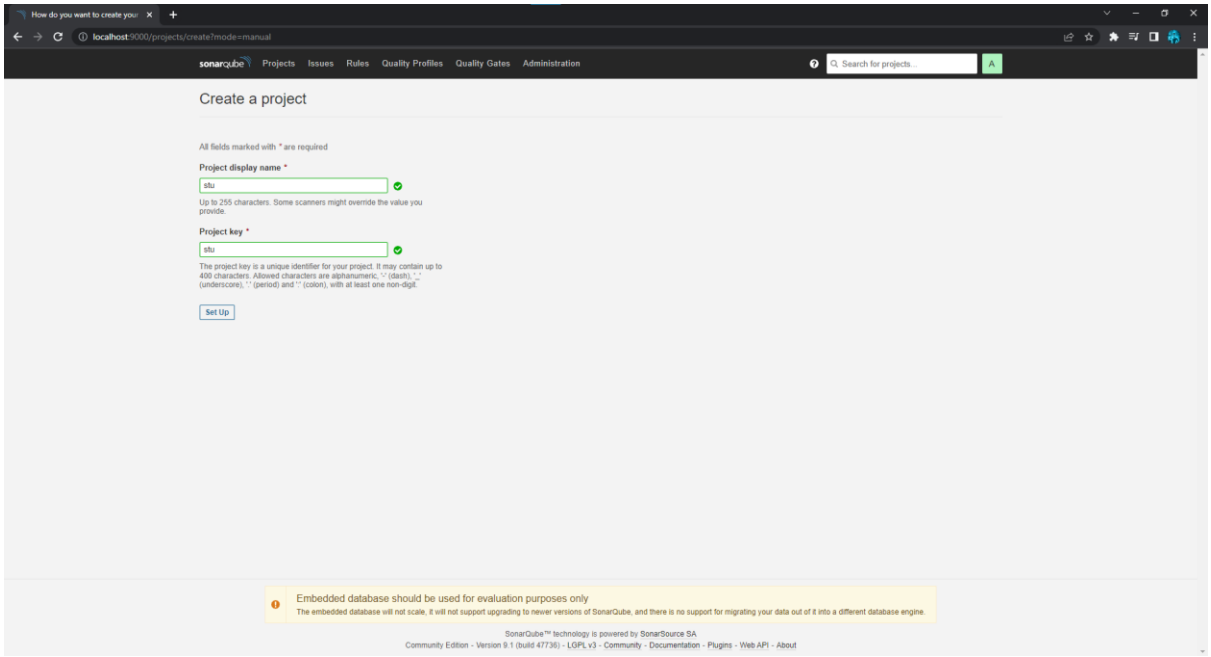
#Configure here general information about the environment, such as SonarQube server connection details for example
#No information about specific project should appear here

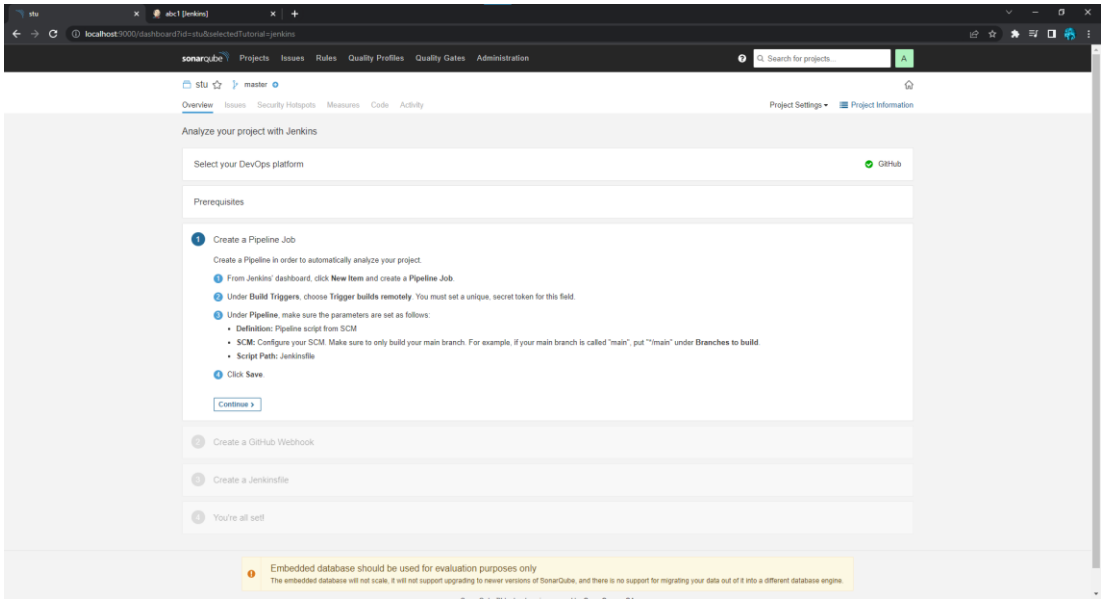
#----- Default SonarQube server
#sonar.host.url=http://localhost:9000

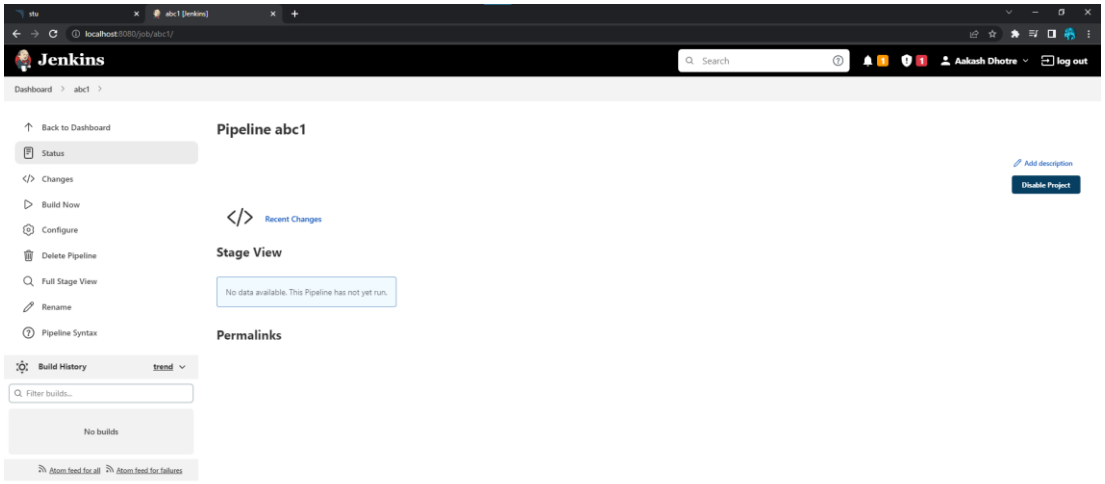
#----- Default source code encoding
#sonar.sourceEncoding=UTF-8
sonar.projectKey=aakash
sonar.projectName=aakash
sonar.projectVersion=1.0
sonar.projectBaseDir=C:/XieJob
sonar. sources=PY Scripts
sonar.login=82a006dff67660f99abd7c73009dc477e5dbc7bb

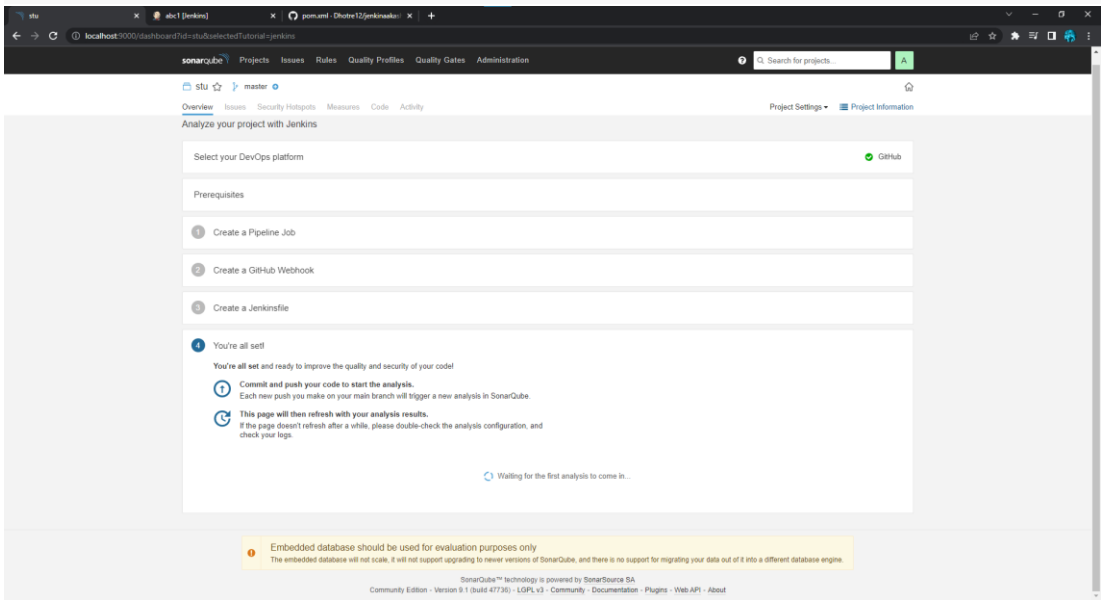
```
Command Prompt
INFO: ------------- Run sensors on project
INFO: Sensor Zero Coverage Sensor
INFO: Sensor Zero Coverage Sensor (done) | time=0ms
INFO: SCM Publisher No SCM system was detected. You can use the 'sonar.scm.provider' property to explicitly specify it.
INFO: CPD Executor Calculating CPD for 0 files
INFO: CPD Executor CPD calculation finished (done) | time=0ms
INFO: Analysis report generated in 138ms, dir size=101.6 kB
INFO: Analysis report compressed in 26ms, zip size=12.4 kB
INFO: Analysis report uploaded in 142ms
INFO: ANALYSIS SUCCESSFUL, you can browse http://localhost:9000/dashboard?id=aakash
INFO: Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
INFO: More about the report processing at http://localhost:9000/api/ce/task?id=AYPMb21L5HlbQdJ7jnHP
INFO: Analysis total time: 15.132 s
INFO: ------------------------------------------------------------------------
INFO: EXECUTION SUCCESS
INFO: ------------------------------------------------------------------------
INFO: Total time: 27.985s
INFO: Final Memory: 7M/27M
INFO: ------------------------------------------------------------------------
```






localhost:9000/dashboard?id=stu&selectedTutorial=jenkins
sonarqube
Projects
Issues
Rules
Quality Profiles
Quality Gates
Administration
Search for projects...
stu
master
Overview
Issues
Security Hotspots
Measures
Code
Activity
Project Settings
Project Information
Analyze your project with Jenkins
Select your DevOps platform
GitHub
Prerequisites
Create a Pipeline Job
Create a Pipeline in order to automatically analyze your project.
From Jenkins' dashboard, click New Item and create a Pipeline Job.
Under Build Triggers, choose Trigger builds remotely. You must set a unique, secret token for this field.
Under Pipeline, make sure the parameters are set as follows:
Definition: Pipeline script from SCM
SCM: Configure your SCM. Make sure to only build your main branch. For example, if your main branch is called "main", put "*/main" under Branches to build.
Script Path: Jenkinsfile
Click Save.
Continue
Create a GitHub Webhook
Create a Jenkinsfile
You're all set!
Embedded database should be used for evaluation purposes only
The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.


localhost:8080/job/abc1/
Jenkins
Search
Aakash Dhotre
log out
Dashboard
abc1
Back to Dashboard
Status
Changes
Build Now
Configure
Delete Pipeline
Full Stage View
Rename
Pipeline Syntax
Build History
trend
Filter builds...
No builds
Atom feed for all
Atom feed for failures
Pipeline abc1
Add description
Disable Project
Recent Changes
Stage View
No data available. This Pipeline has not yet run.
Permalinks
REST API
Jenkins 2.346.2


localhost:9000/dashboard?id=stu&selectedTutorial=jenkins
sonarqube
Projects
Issues
Rules
Quality Profiles
Quality Gates
Administration
Search for projects...
stu
master
Overview
Issues
Security Hotspots
Measures
Code
Activity
Project Settings
Project Information
Analyze your project with Jenkins
Select your DevOps platform
GitHub
Prerequisites
Create a Pipeline Job
Create a GitHub Webhook
Create a Jenkinsfile
You're all set!
You're all set and ready to improve the quality and security of your code!
Commit and push your code to start the analysis.
Each new push you make on your main branch will trigger a new analysis in SonarQube.
This page will then refresh with your analysis results.
If the page doesn't refresh after a while, please double-check the analysis configuration, and check your logs.
Waiting for the first analysis to come in...
Embedded database should be used for evaluation purposes only
The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.
SonarQube™ technology is powered by SonarSource SA
Community Edition - Version 9.1 (build 47736) - LGPL v3 - Community - Documentation - Plugins - Web API - About

Conclusion: Hence, we have studied to create a Jenkins CICD Pipeline with SonarQube / GitLab Integration to perform a static analysis of the code to detect bugs, code smells, and security vulnerabilities on a sample Web / Java / Python application.

Questionnaire:

Q1. What is SonarQube?

Ans: SonarQube is an open-source framework developed by SonarSource for continuous inspection of code quality to conduct automated reviews of 20 + programming languages with static code analysis to identify bugs, code bad smells and security vulnerabilities.

Q2 Why to use SonarQube?

Ans: SonarQube increases productivity by allowing development teams to detect and muzzle duplication and redundancy of code. SonarQube makes it easier for team members to reduce application size, code complexity, time and cost of maintenance, and make code easier to read and understand.

Q3. What is difference between SonarQube and SonarLint?

Ans: SonarLint:

- SonarLint exists only in the IDE (IntelliJ, Visual Studio and Eclipse).
- Its aim is to provide immediate feedback as you type in your code.
- It focuses on what code you add or update for this function.
- SonarLint is an agent that allow us to connect with this SonarQube and execute the analysis remotely.

SonarQube:

- SonarQube is a central server which performs full analysis (triggered by the different SonarQube scanners).
- The purpose is to give your code base a 360 Â° view of the quality. To this end, it periodically analyzes all of the source lines of your project.

Both SonarLint and SonarQube depend on the same analyzers for static source code-most of which are written using SonarSource technology.

Q4. Is SonarQube Replacement for Checkstyle, PMD, FindBugs?

Ans: By default for Java projects, Sonar will run CheckStyle, FindBugs and PMD, as well as a few other "plugins" such as Cobertura . The main added advantage is that it stores the history in a database.

These 3 tools are used by Sonar as plugins and the data from all three of these tools is applied with a value that displays graphs.

Q5. What is difference between Sonar Runner and Sonar Scanner?

Ans: The old name for "Scanner" is "Runner."

All you need to know about the different SonarQube Scanners is available in the Scanners section of the official documentation.

You can use below option,if you are stuck to Java 7:

- SonarQube Runner (sonar-runner) up to version 5.5 of SonarQube
- SonarQube Scanner (sonar-scanner) 2.6.1

Experiment 9

Aim: To Understand Continuous monitoring and Installation and configuration of Nagios Core, Nagios Plugins and NRPE (Nagios Remote Plugin Executor) on Linux Machine.

Theory:
Nagios is a popular monitoring tool many DevOps teams use to ensure thorough and efficient tracking of systems, devices, apps, and services. However, the tool does have a steep learning curve, so setting up and starting to use Nagios can be tricky without prior experience or a good tutorial.

What Is Continuous Monitoring?
Continuous monitoring is a process of constant detecting, reporting, and responding to risks and events within an IT system. This process is a vital DevOps security practice and has multiple goals:

- Provide real-time insight into system performance.
- Offer feedback on the overall health and security of IT infrastructure.
- Enhance visibility across IT operations and the DevOps pipeline.
- Identify the cause of incidents and apply mitigation before the problem results in downtime or a data breach.

The need for continuous tracking comes from the issues of manual monitoring as traditional tracking is too prone to:

- Slowing down deployments in CI/CD pipelines.
- Causing performance issues in production.
- Lengthy and challenging root-cause analysis.

The ability to quickly detect, report, and respond to threats is vital to a company's overall cybersecurity. Continuous monitoring is also a standard practice within SecOps teams as reliable, real-time insights throughout environments improve:

- Threat intelligence.
- Root cause analysis.
- Incident responses.
- Post-incident forensics.

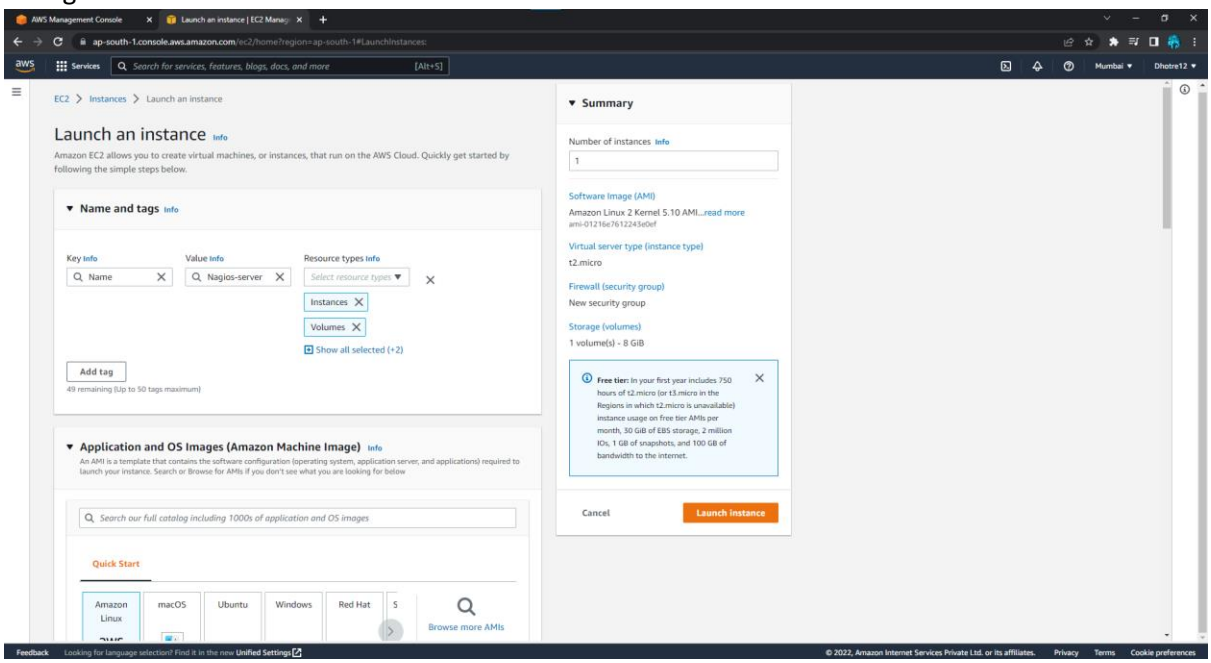
Other popular terms for continuous monitoring are ConMon and Continuous Control Monitoring (CCM).

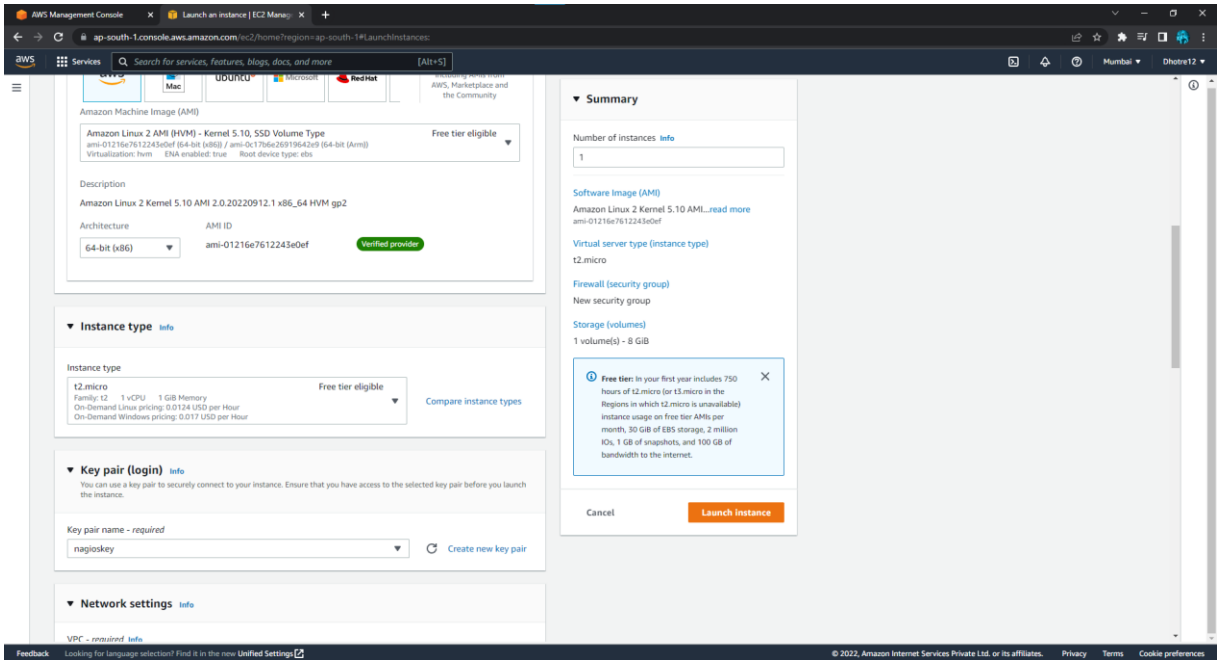
Why Is ConMon Important?
Continuous monitoring is a vital aspect of modern cybersecurity. A sound ConMon solution allows a security team to:

- Quickly detect system issues (network errors, low memory, port failures, system crashes, unreachable servers, etc.).
- Resolve problems before they impact users or business productivity.
- Identify security and compliance risks within the infrastructure.
- Lower the risk of cyberattacks with a timely alert system and automatic incident responses.
- Precisely identify the root cause of an issue.
- Maintain high levels of system uptime and availability.
- Use precise historical analysis to plan infrastructure upgrades.

Result:

To start Nagios Core installation you must have your EC2 instance up and run and have already configured SSH access to the instance.





Create key pair

Key pairs allow you to connect to your instance securely.

Enter the name of the key pair below. When prompted, store the private key in a secure and accessible location on your computer. **You will need it later to connect to your instance.** Learn more

Key pair name

nagioskey

The name can include upto 255 ASCII characters. It can't include leading or trailing spaces.

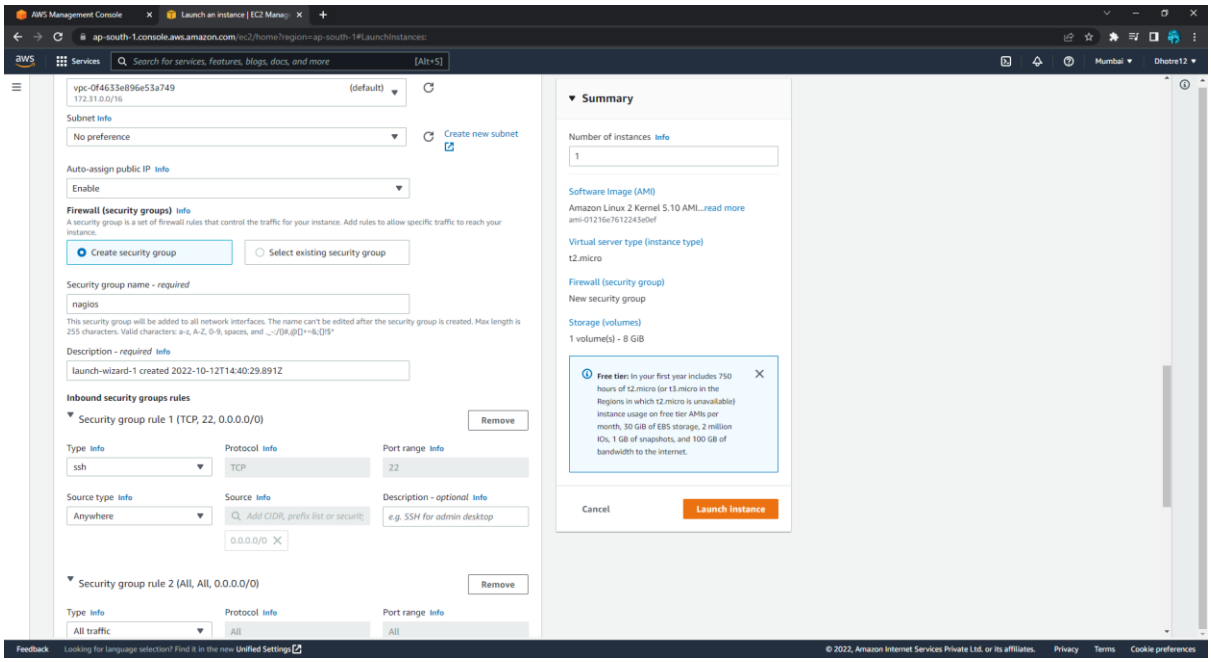
Key pair type

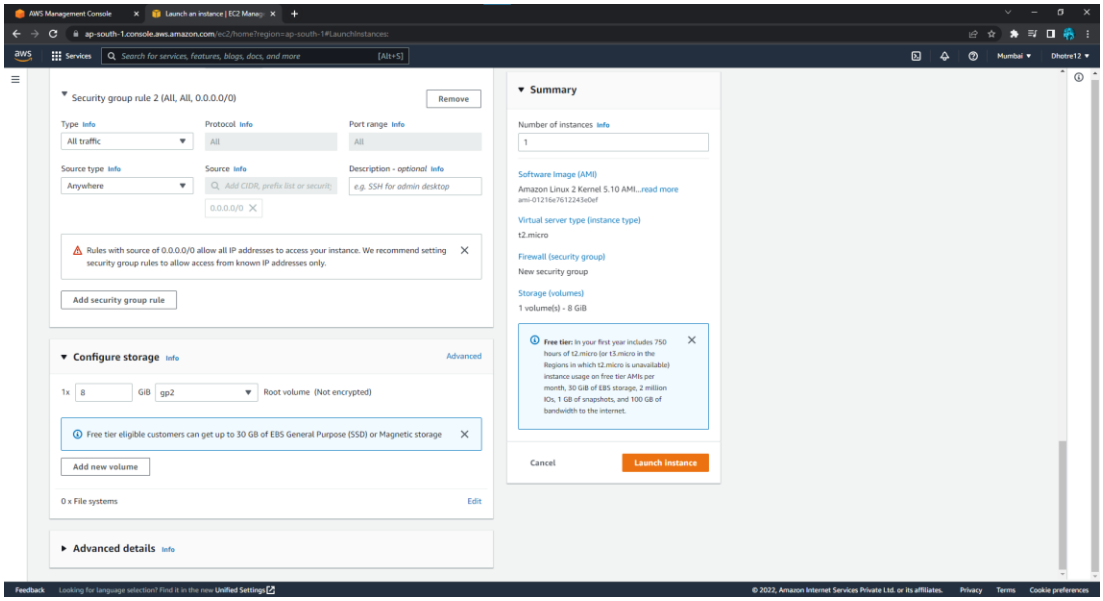
- (●) RSA
  RSA encrypted private and public key pair
- ( ) ED25519
  ED25519 encrypted private and public key pair (Not supported for Windows instances)

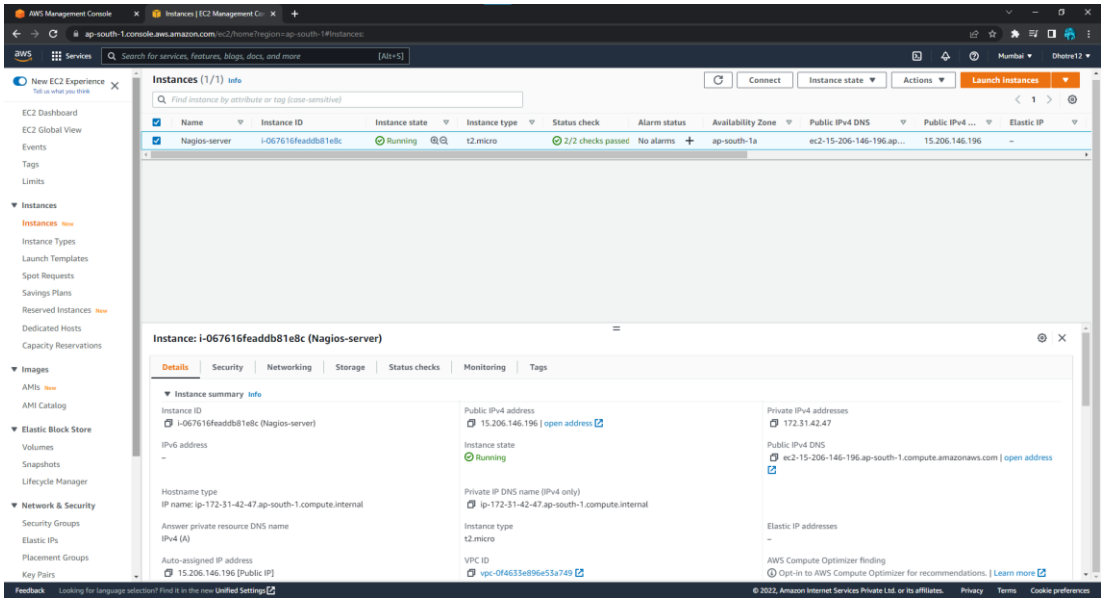
Private key file format

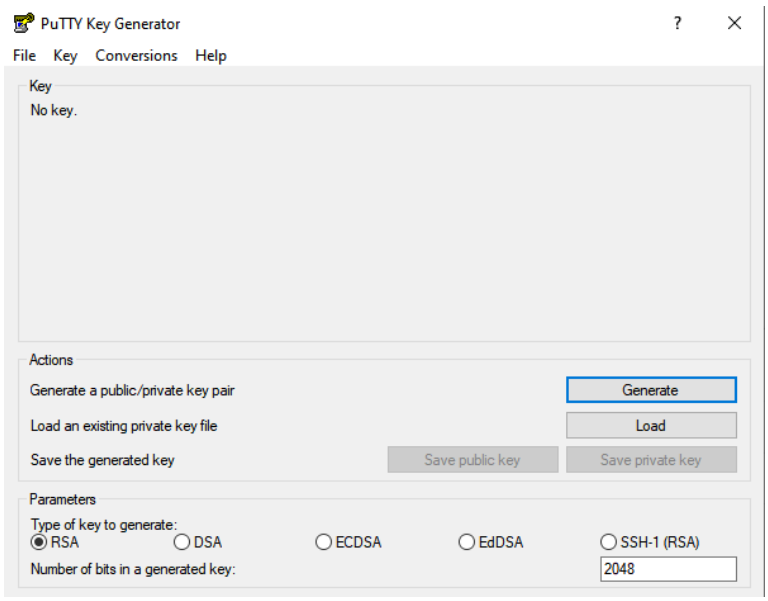
- (●) .pem
  For use with OpenSSH
- ( ) .ppk
  For use with PuTTY

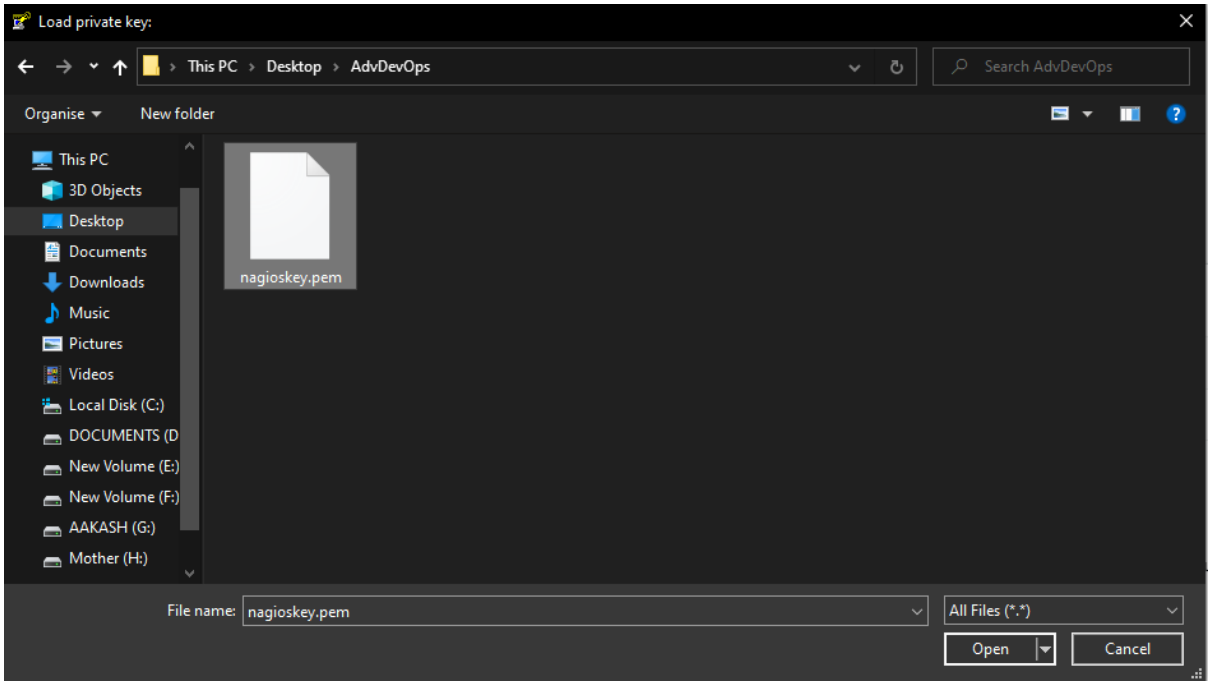
Cancel | Create key pair

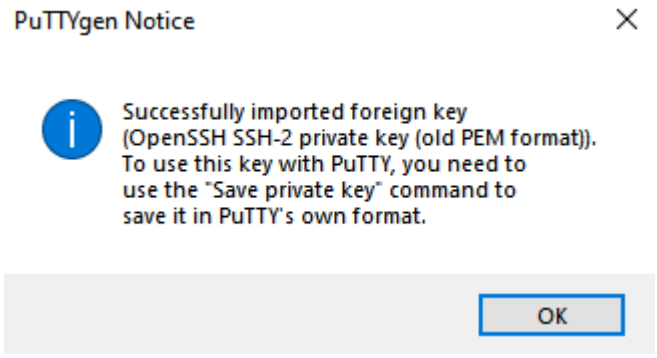


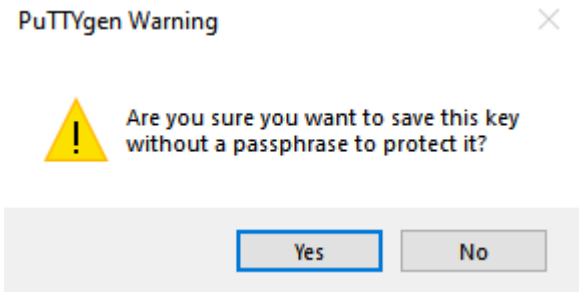

Security group rule 2 (All, All, 0.0.0.0/0)
Remove
Type Info
All traffic
Protocol Info
All
Port range Info
All
Source type Info
Anywhere
Source Info
Add CIDR, prefix list or securit
Description - optional Info
e.g. SSH for admin desktop
0.0.0.0/0
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.
Add security group rule
Configure storage Info
Advanced
1x 8 GiB gp2 Root volume (Not encrypted)
Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage
Add new volume
0 x File systems
Edit
Advanced details Info
Summary
Number of instances Info
1
Software Image (AMI)
Amazon Linux 2 Kernel 5.10 AMI...read more
Virtual server type (instance type)
t2.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.
Cancel
Launch instance

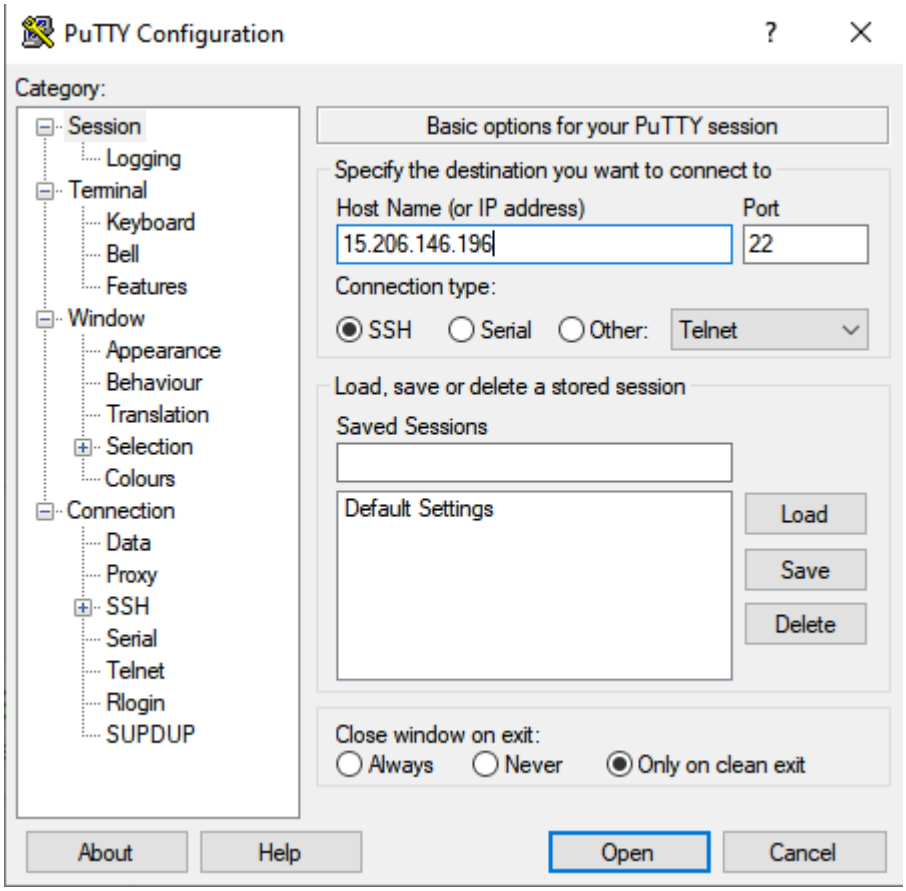

Instances (1/1) Info
Connect
Instance state
Actions
Launch instances
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
Public IPv4 DNS
Public IPv4 ...
Elastic IP
Nagios-server
i-067616feaddb81e8c
Running
t2.micro
2/2 checks passed
No alarms
ap-south-1a
ec2-15-206-146-196.ap...
15.206.146.196
Instance: i-067616feaddb81e8c (Nagios-server)
Details
Security
Networking
Storage
Status checks
Monitoring
Tags
Instance summary Info
Instance ID
i-067616feaddb81e8c (Nagios-server)
Public IPv4 address
15.206.146.196 | open address
Private IPv4 addresses
172.31.42.47
IPv6 address
Instance state
Running
Public IPv4 DNS
ec2-15-206-146-196.ap-south-1.compute.amazonaws.com | open address
Hostname type
IP name: ip-172-31-42-47.ap-south-1.compute.internal
Private IP DNS name (IPv4 only)
ip-172-31-42-47.ap-south-1.compute.internal
Answer private resource DNS name
IPv4 (A)
Instance type
t2.micro
Elastic IP addresses
Auto-assigned IP address
15.206.146.196 [Public IP]
VPC ID
vpc-0f4633e896e53a749
AWS Compute Optimizer finding
Opt-in to AWS Compute Optimizer for recommendations. | Learn more

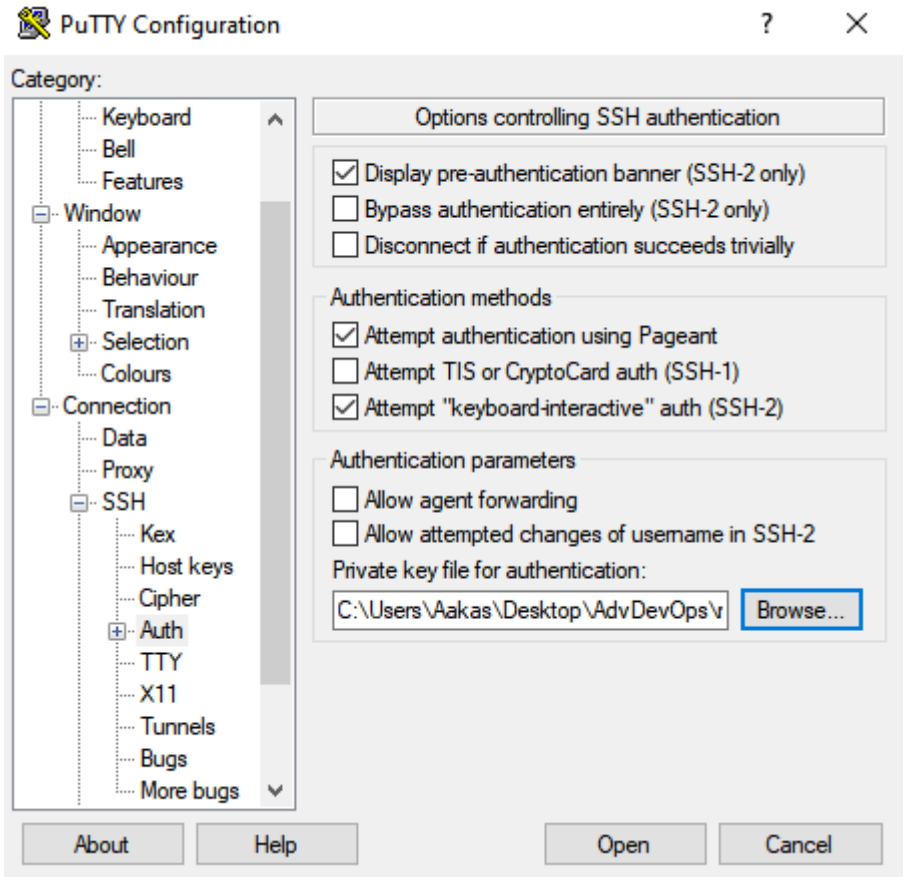

PuTTY Key Generator
File Key Conversions Help
Key
No key.
Actions
Generate a public/private key pair
Generate
Load an existing private key file
Load
Save the generated key
Save public key
Save private key
Parameters
Type of key to generate:
RSA
DSA
ECDSA
EdDSA
SSH-1 (RSA)
Number of bits in a generated key:
2048


Load private key:
This PC › Desktop › AdvDevOps
Search AdvDevOps
Organise
New folder
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Local Disk (C:)
DOCUMENTS (D
New Volume (E:)
New Volume (F:)
AAKASH (G:)
Mother (H:)
nagioskey.pem
File name: nagioskey.pem
All Files (*.*)
Open
Cancel


PuTTYgen Notice
Successfully imported foreign key
(OpenSSH SSH-2 private key (old PEM format)).
To use this key with PuTTY, you need to
use the "Save private key" command to
save it in PuTTY's own format.
OK


PuTTYgen Warning
Are you sure you want to save this key
without a passphrase to protect it?
Yes
No


PuTTY Configuration
Category:
Session
Logging
Terminal
Keyboard
Bell
Features
Window
Appearance
Behaviour
Translation
Selection
Colours
Connection
Data
Proxy
SSH
Serial
Telnet
Rlogin
SUPDUP
Basic options for your PuTTY session
Specify the destination you want to connect to
Host Name (or IP address)
Port
15.206.146.196
22
Connection type:
SSH
Serial
Other:
Telnet
Load, save or delete a stored session
Saved Sessions
Default Settings
Load
Save
Delete
Close window on exit:
Always
Never
Only on clean exit
About
Help
Open
Cancel


PuTTY Configuration
Category:
Keyboard
Bell
Features
Window
Appearance
Behaviour
Translation
Selection
Colours
Connection
Data
Proxy
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)
Disconnect if authentication succeeds trivially
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:
C:\Users\Aakas\Desktop\AdvDevOps\v
Browse...
About
Help
Open
Cancel

Step 1: Install pre-requisite softwares on your EC2 machine prior to Nagios installation like apache, php, gcc compiler and gd development libraries.
sudo su
yum install httpd php
yum install gcc glibc glibc-common
yum install gd gd-devel
Step 2: Create account information you need to setup a nagios user. Run the following commands.
adduser -m nagios
passwd nagios

Now, it ask to enter new password give '12345' as password
groupadd nagioscmd
usermod -a -G nagioscmd nagios
usermod -a -G nagioscmd apache

Step 3: Download nagios core and the plugins. Create a directory for storing the downloads.
mkdir ~/downloads
cd ~/downloads

Download the source code tarballs of both nagios and the nagios plugins.
wget http://prdownloads.sourceforge.net/sourceforge/nagios/nagios-4.0.8.tar.gz
wget http://nagios-plugins.org/download/nagios-plugins-2.0.3.tar.gz

Step 4: Compile and install Nagios extract the nagios source code tarball.
tar zxvf nagios-4.0.8.tar.gz
cd nagios-4.0.8

Run the configuration script with the name of the group which you have created in above step.
./configure --with-command-group=nagioscmd

Compile the Nagios source code.
make all

Install Binaries, init script, sample Config files and set permissions on the external command directly.
make install
make install-init
make install-config
make install-commandmode

Step 5: Configure the Web interface.
make install-webconf

Step 6: Create a 'nagiosadmin' account for login into the nagios web interface. Set password as well.
htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin

Asking for password, set a new pwd.
service httpd restart

Step 7: Compile and install the Nagios plugins Extract the Nagios plugins source code tarball.
cd ~/downloads
tar zxvf nagios-plugins-2.0.3.tar.gz
cd nagios-plugins-2.0.3

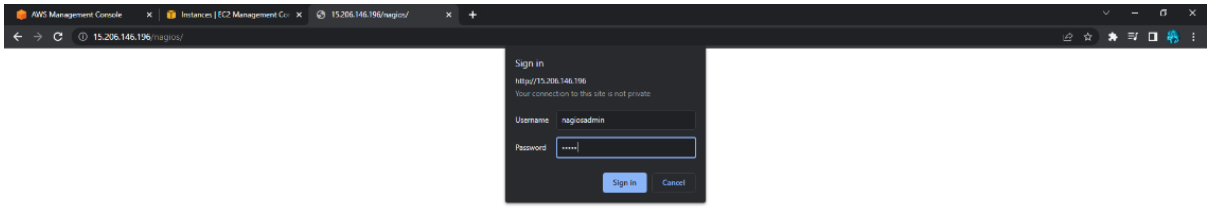
Compile and install the plugins.
./configure --with-nagios-user=nagios --with-nagios-group=nagios
make
make install

Step 8: Start Nagios. Add Nagios to the list of system services and have it automatically start when the system boots.
chkconfig --add nagios
chkconfig nagios on

Verify the Sample Nagios configuration files.
/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

If there are no errors, start nagios.
service nagios start
service httpd restart

Step 9: Copy publicip of EC2 instance and paste in google chrome, in given way
For eg. 20.1.1.1/nagios/
Ask for username: nagiosadmin
password: 12345



Conclusion: Hence, we have studied to understand Continuous monitoring and Installation and configuration of Nagios Core, Nagios Plugins and NRPE (Nagios Remote Plugin Executor) on Linux Machine.

Questionnaire:

Q1. What is Nagios?

Ans. Nagios is one of the monitoring tools that is used for Continuous monitoring of systems, applications, services, and business processes etc. in a DevOps culture. In the event of a failure, Nagios can alert technical staff of the problem, allowing them to begin remediation processes before outages affects business processes, end-users, or customers. With Nagios you don't have to explain why an unseen infrastructure outage affect your organization's bottom line.

Q2. How does Nagios work?

Ans. It is as follows:

- Nagios runs on a server, usually as a daemon or service.
- Nagios periodically runs plugins residing on the same server, they contact hosts or servers on your network or on the internet.
- One can view the status information using the web interface.
- You can also receive email or SMS notifications if something happens.
- The Nagios daemon behaves like a scheduler that runs certain scripts at certain moments.
- It stores the results of those scripts and will run other scripts if these results change.

Q3. What are Plugins in Nagios?

Ans. Plugins are scripts (Perl scripts, Shell scripts, etc.) that can run from a command line to check the status of a host or service. Nagios uses the results from the plugins to determine the current status of hosts and services on your network.

Nagios will execute a Plugin whenever there is a need to check the status of a host or service. The plugin will perform the check and then simply returns the result to Nagios. Nagios will process the results that it receives from the Plugin and take the necessary actions.

Q4. What is NRPE (Nagios Remote Plugin Executor) in Nagios?

Ans. The NRPE addon is designed to allow you to execute Nagios plugins on remote Linux/Unix machines. The main reason for doing this is to allow Nagios to monitor "local" resources (like CPU load, memory usage, etc.) on remote machines. Since these public resources are not usually exposed to external machines, an agent like NRPE must be installed on the remote Linux/Unix machines.

The NRPE addon consists of two pieces:

- The check_nrpe plugin, which resides on the local monitoring machine.
- The NRPE daemon, which runs on the remote Linux/Unix machine.

Q5. What is meant by Nagios backend?

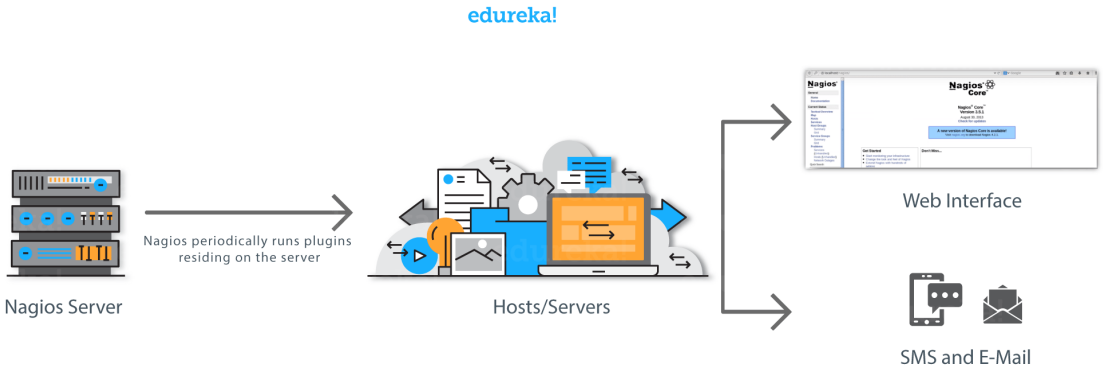
Ans. Both Configuration and Logs can be stored in a backend. Configurations are stored in backend using NagiosQL. Historical data are stored using ndoutils. In addition, you also have nagdb and opdb.

Experiment 10

Aim: To perform Port, Service monitoring, Windows/Linux server monitoring using Nagios.

Theory:
Nagios is used for Continuous monitoring of systems, applications, services, and business processes etc in a DevOps culture. In the event of a failure, Nagios can alert technical staff of the problem, allowing them to begin remediation processes before outages affect business processes, end-users, or customers. With Nagios, you don't have to explain why an unseen infrastructure outage affect your organization's bottom line.

Let me explain to you how Nagios works. Consider the diagram below:



Nagios Working - Nagios Tutorial - EdurekaNagios runs on a server, usually as a daemon or a service.

It periodically runs plugins residing on the same server, they contact hosts or servers on your network or on the internet. One can view the status information using the web interface. You can also receive email or SMS notifications if something happens.
The Nagios daemon behaves like a scheduler that runs certain scripts at certain moments. It stores the results of those scripts and will run other scripts if these results change.

Plugins: These are compiled executables or scripts (Perl scripts, shell scripts, etc.) that can be run from a command line to check the status or a host or service. Nagios uses the results from the plugins to determine the current status of the hosts and services on your network.

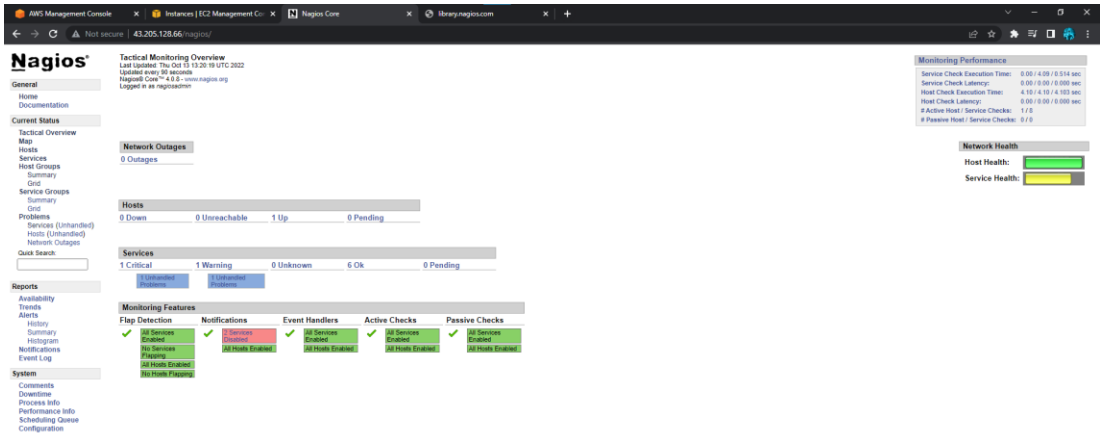
Let's now discuss its architecture.

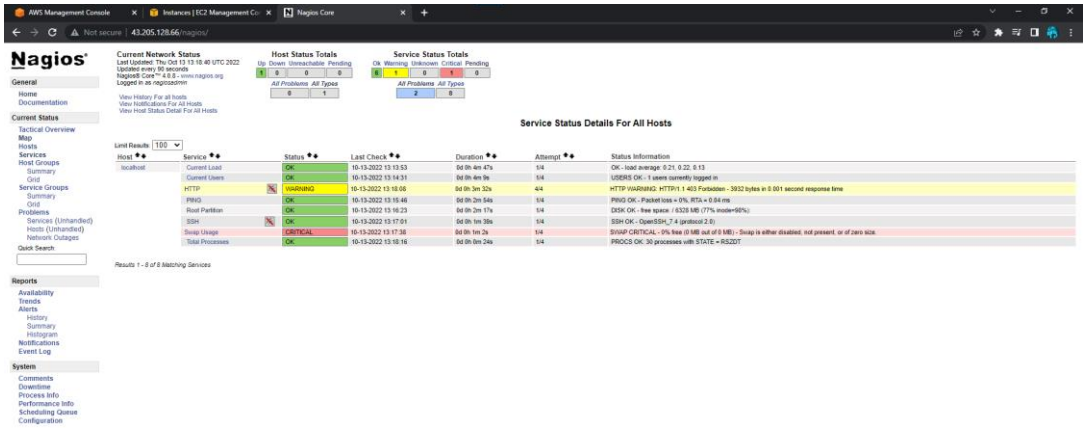
Nagios Architecture:

- Nagios is built on a server/agent's architecture.
- Usually, on a network, a Nagios server is running on a host, and Plugins interact with local and all the remote hosts that need to be monitored.
- These plugins will send information to the Scheduler, which displays that in a GUI.

Result:


Nagios
Nagios Core
Daemon running with PID 26154
Nagios® Core™
Version 4.0.8
August 12, 2014
Check for updates
A new version of Nagios Core is available!
Visit nagios.org to download Nagios 4.4.8.
Nagios XI
Easy Configuration, Advanced Reporting
Download
Nagios Log Server
Monitor and analyze logs from anywhere
Download
Nagios Network Analyzer
Real-time netflow and bandwidth analysis
Download
Get Started
Start monitoring your infrastructure
Change the look and feel of Nagios
Extend Nagios with hundreds of addons
Get support
Get training
Get certified
Quick Links
Nagios Library (tutorials and docs)
Nagios Labs (development blog)
Nagios Exchange (plugins and addons)
Nagios Support (tech support)
Nagios.com (company)
Nagios.org (project)
Latest News
Nagios Update: XI 5.6.6
Nagios Update: XI 5.6.5
Nagios Update: XI 5.6.4
More news...
Don't Miss...
Monitoring Log Data with Nagios - Nagios Log Server can handle all log data in one central location.
Can Nagios monitor netflow? - Yes! Nagios Network Analyzer can take in a variety of flow data. Learn More
Nagios XI 5 Available Now! - Easier configuration, Advanced Reporting. Download Today!


Tactical Monitoring Overview
Monitoring Performance
Network Outages
0 Outages
Network Health
Host Health:
Service Health:
Hosts
0 Down
0 Unreachable
1 Up
0 Pending
Services
1 Critical
1 Warning
0 Unknown
6 Ok
0 Pending
Monitoring Features
Flap Detection
Notifications
Event Handlers
Active Checks
Passive Checks


Current Network Status
Host Status Totals
Service Status Totals
Service Status Details For All Hosts
Host
Service
Status
Last Check
Duration
Attempt
Status Information

Conclusion: Hence we have studied to perform Port, Service monitoring, Windows/Linux server monitoring using Nagios and learned to implement them.

Questionnaire:

Q1. What do you mean by passive check in Nagios?

Ans. Passive checks are initiated and performed by external applications/processes and the Passive check results are submitted to Nagios for processing.

Passive checks are useful for monitoring services that are Asynchronous in nature and cannot be monitored effectively by polling their status on a regularly scheduled basis. It can also be used for monitoring services that are Located behind a firewall and cannot be checked actively from the monitoring host.

Q2. When Does Nagios Check for external commands?

Ans. Nagios check for external commands under the following conditions:

- At regular intervals specified by the command_check_interval option in the main configuration file or,
- Immediately after event handlers are executed. This is in addition to the regular cycle of external command checks and is done to provide immediate action if an event handler submits commands to Nagios.

Q3. What is the difference between Active and Passive check in Nagios?

Ans. The major difference between Active and Passive checks is that Active checks are initiated and performed by Nagios, while passive checks are performed by external applications.

Passive checks are useful for monitoring services that are:

- Asynchronous in nature and cannot be monitored effectively by polling their status on a regularly scheduled basis.
- Located behind a firewall and cannot be checked actively from the monitoring host.

The main features of Actives checks are as follows:

- Active checks are initiated by the Nagios process.
- Active checks are run on a regularly scheduled basis.

Q4. How does Nagios help with Distributed Monitoring?

Ans. With Nagios you can monitor your whole enterprise by using a distributed monitoring scheme in which local slave instances of Nagios perform monitoring tasks and report the results back to a single master. You manage all configuration, notification, and reporting from the master, while the slaves do all the work. This design takes advantage of Nagios's ability to utilize passive checks i.e. external applications or processes that send results back to Nagios. In a distributed configuration, these external applications are other instances of Nagios.

Q5. Explain Main Configuration file of Nagios and its location?

Ans. The main configuration file contains a number of directives that affect how the Nagios daemon operates. This config file is read by both the Nagios daemon and the CGIs (It specifies the location of your main configuration file).

Experiment 11

Aim: To understand AWS Lambda, its workflow, various functions and create your first Lambda functions using Python / Java / Nodejs.

Theory:
Lambda is a compute service that lets you run code without provisioning or managing servers. Lambda runs your code on a high-availability compute infrastructure and performs all of the administration of the compute resources, including server and operating system maintenance, capacity provisioning and automatic scaling, and logging. With Lambda, you can run code for virtually any type of application or backend service. All you need to do is supply your code in one of the languages that Lambda supports.

You organize your code into Lambda functions. Lambda runs your function only when needed and scales automatically, from a few requests per day to thousands per second. You pay only for the compute time that you consume—there is no charge when your code is not running.

You can invoke your Lambda functions using the Lambda API, or Lambda can run your functions in response to events from other AWS services. For example, you can use Lambda to:

- Build data-processing triggers for AWS services such as Amazon Simple Storage Service (Amazon S3) and Amazon DynamoDB.
- Process streaming data stored in Amazon Kinesis.
- Create your own backend that operates at AWS scale, performance, and security.

When should I use Lambda?
Lambda is an ideal compute service for many application scenarios, as long as you can run your application code using the Lambda standard runtime environment and within the resources that Lambda provides.

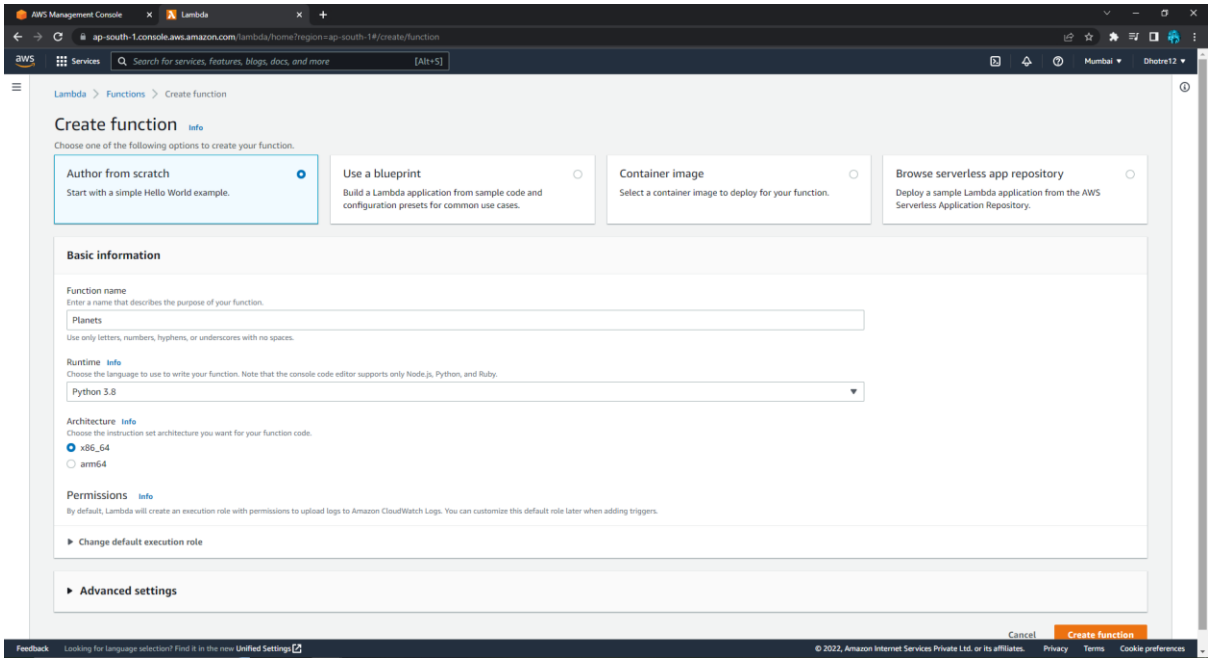
When using Lambda, you are responsible only for your code. Lambda manages the compute fleet that offers a balance of memory, CPU, network, and other resources to run your code. Because Lambda manages these resources, you cannot log in to compute instances or customize the operating system on provided runtimes. Lambda performs operational and administrative activities on your behalf, including managing capacity, monitoring, and logging your Lambda functions.

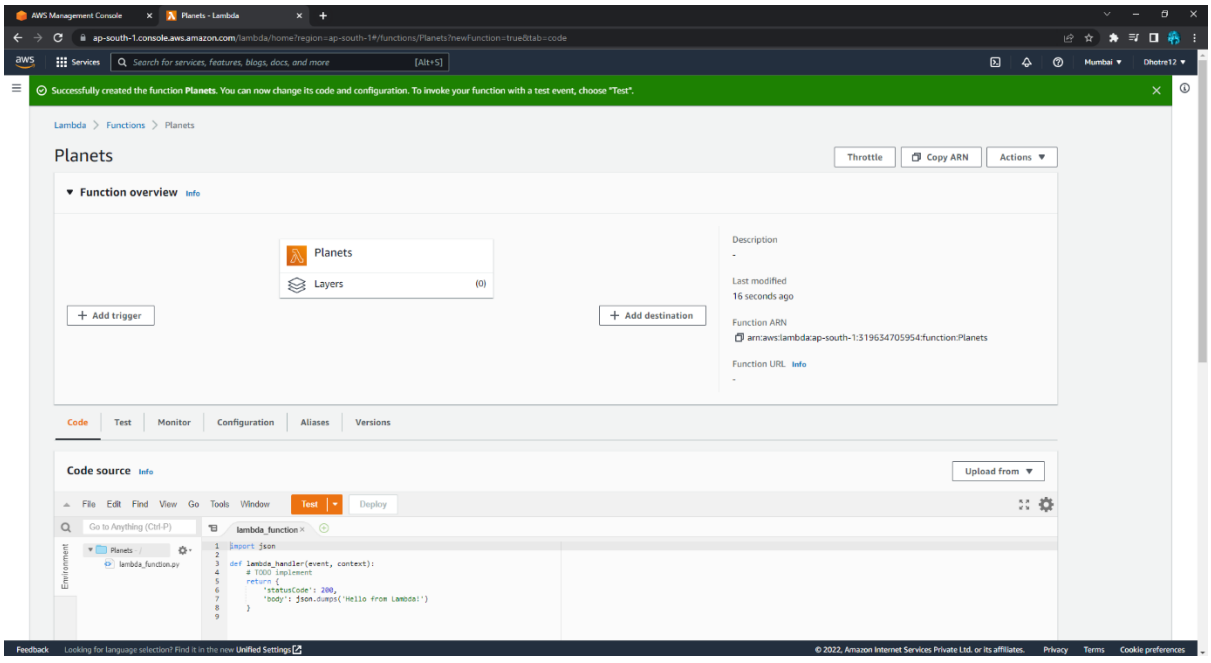
If you need to manage your own compute resources, AWS has other compute services to meet your needs. For example:

- Amazon Elastic Compute Cloud (Amazon EC2) offers a wide range of EC2 instance types to choose from. It lets you customize operating systems, network and security settings, and the entire software stack. You are responsible for provisioning capacity, monitoring fleet health and performance, and using Availability Zones for fault tolerance.
- AWS Elastic Beanstalk enables you to deploy and scale applications onto Amazon EC2. You retain ownership and full control over the underlying EC2 instances.

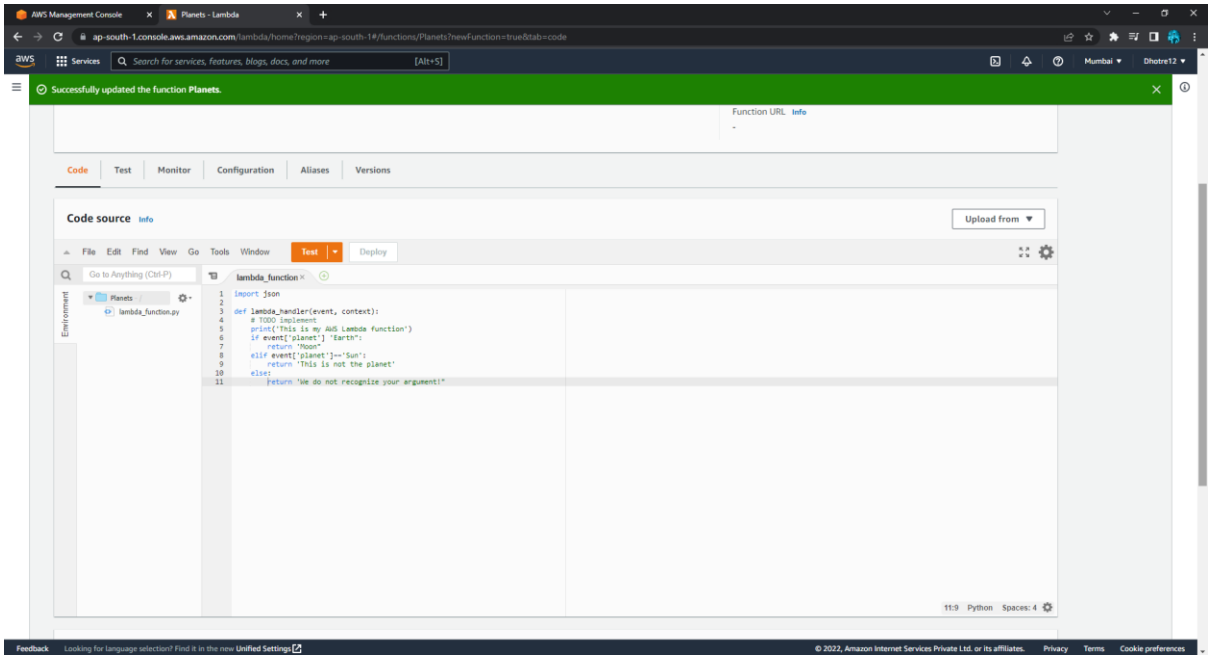
Result:

Step 1: Create function in lambda

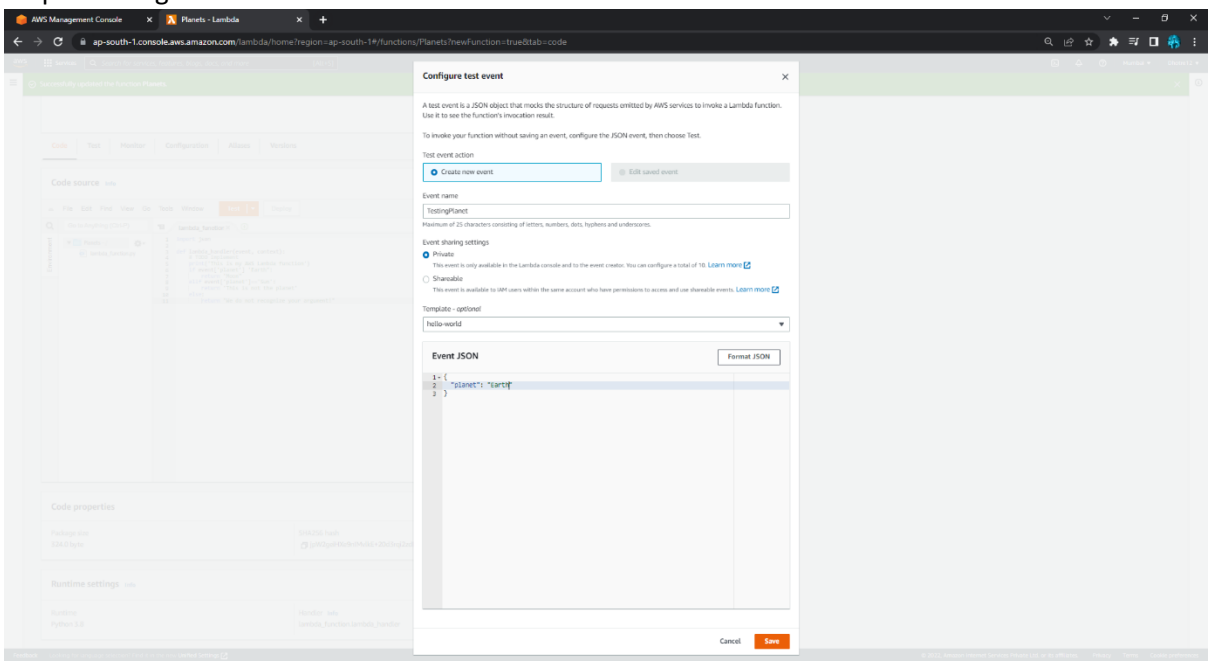




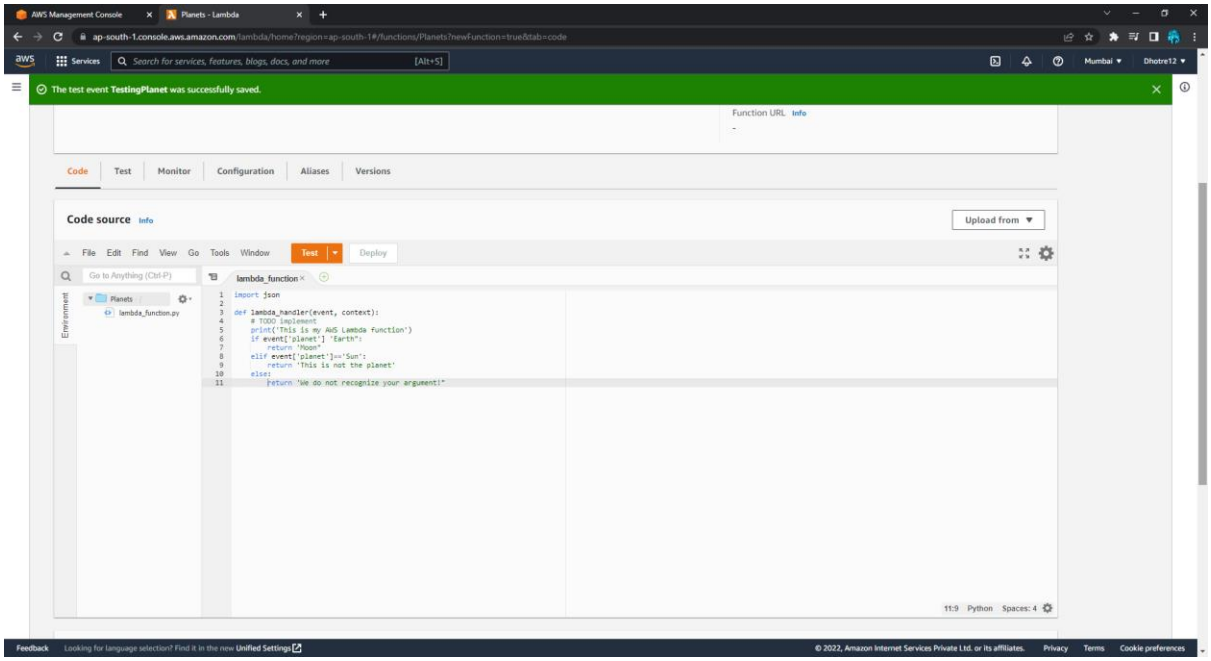
Step 2: Written code for planets.



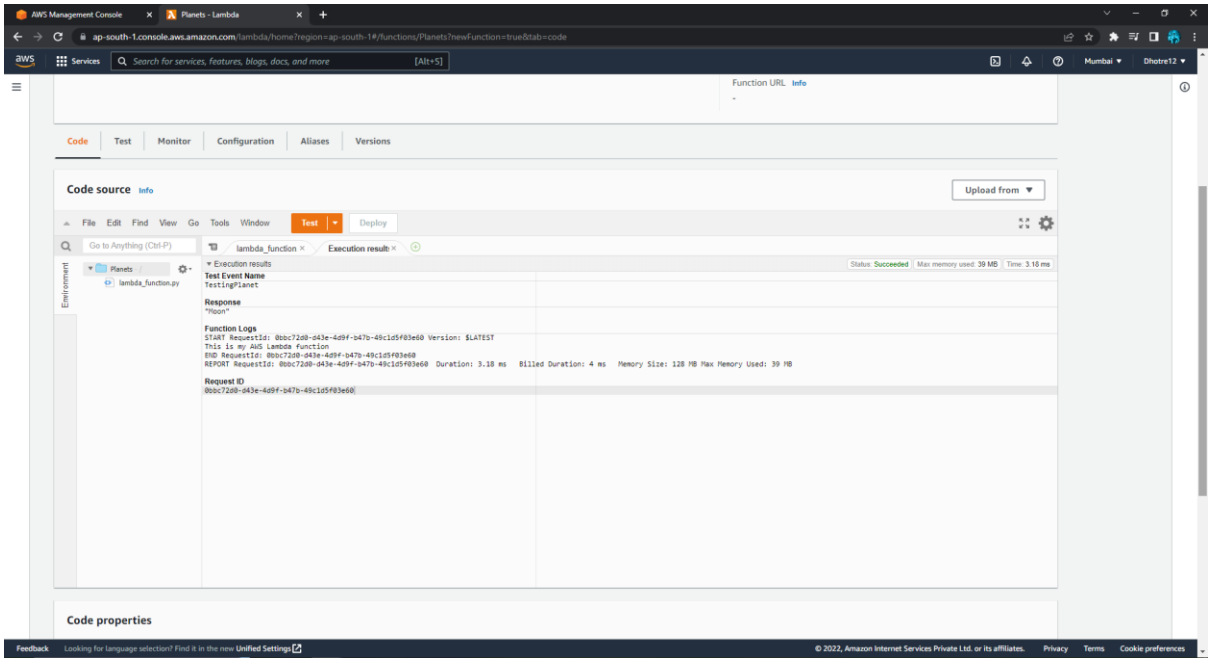
Step 3: Configure test event

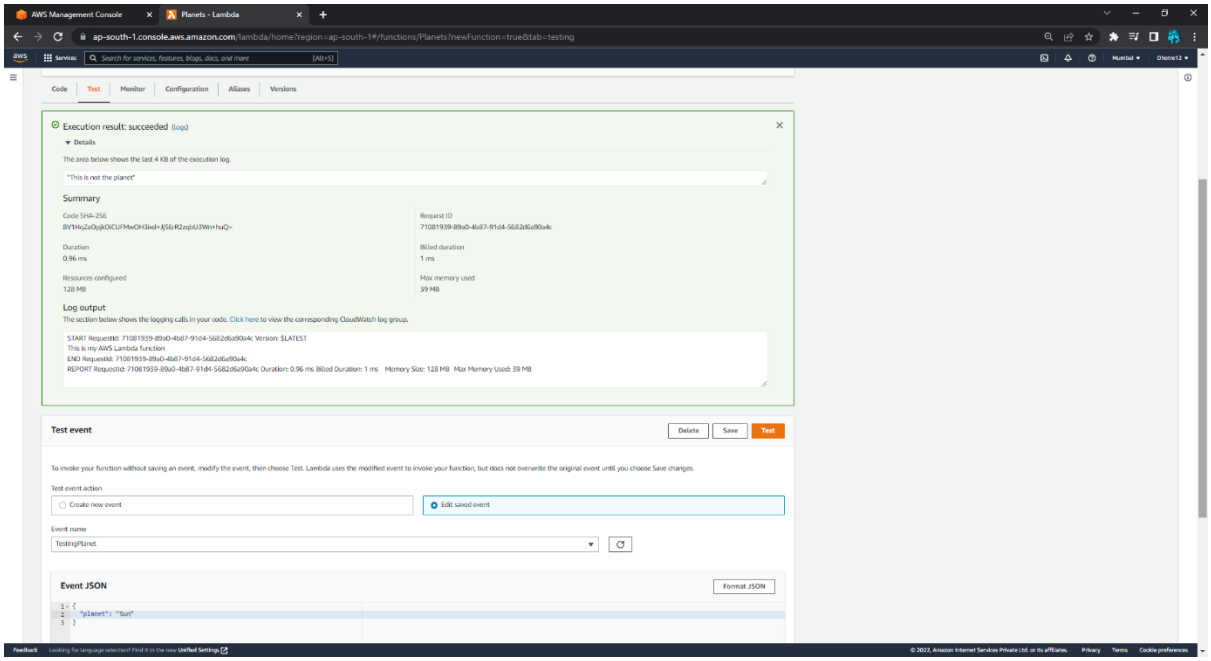


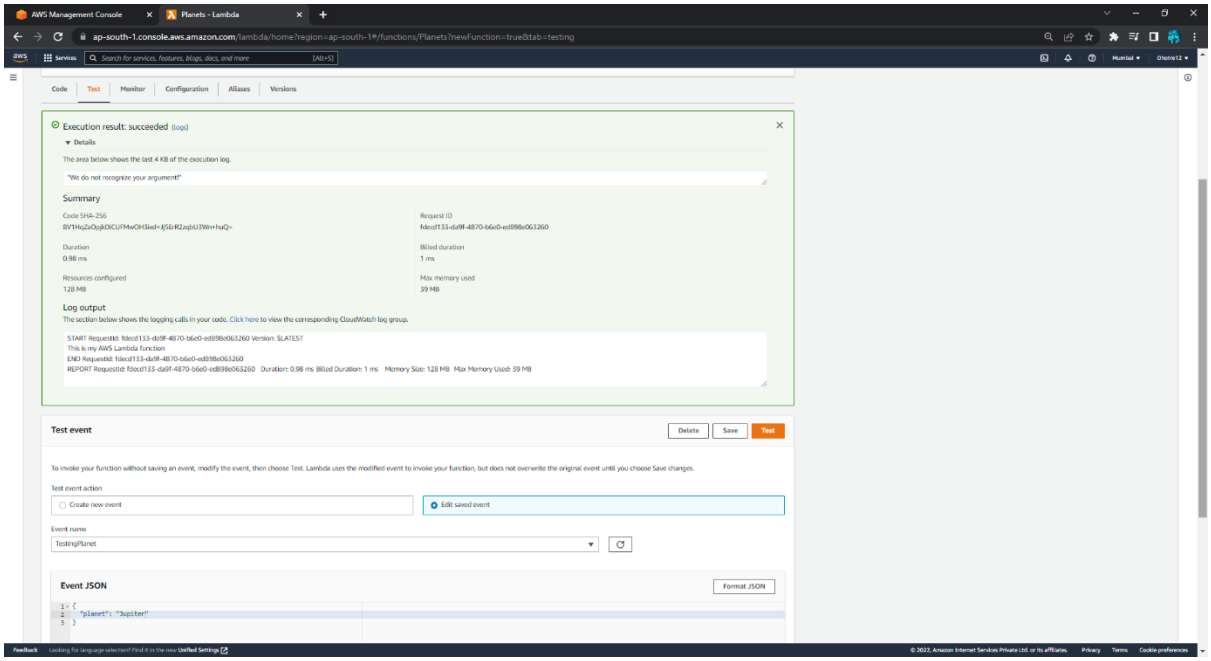
Step 4: Deploy



Step 5: Test the code.







Conclusion: Hence, we have studied to understand AWS Lambda, its workflow, various functions and create your first Lambda functions using Python / Java / Nodejs.

Questionnaire:

1. What exactly automates deployment is?

Ans. It is quite similar to programming in other languages. However, it cut down a lot of challenges associated. The best thing is the deployment of a pipeline that can easily be created as one become more proficient. Automate Deployment cuts down human interference and help the organizations ensure outcomes that are quality-based and are best in every aspect.

2. What are the features in AWS lambda that automate the deployment?

Ans. There are environmental variables that are supported by AWS lambda. They can be used for data and several other credentials when it comes to modifying the deployment package. As it's a serverless approach, it also supports aliases. There are certain types in fact that you can easily consider such as stage production and dev. Thus, functions can easily be considered for testing and without actually interrupting the production code. The end-point doesn't change easily and thus one can keep up the pace with the task.

3. What are the various ways to access EC2?

Ans. This can be done with Command Line Interface and Web-Based Interface. Also, there are tools for PowerShell in Windows through which it can simply be done.

4. Tell us about the frameworks which are available for serverless?

Ans. There are several frameworks and serverless is extremely powerful. Its great support to Lambda and open whisk, as well as azure functions, makes it simply the best in every aspect. When it comes to extending the cloud formation, the Serverless Application Model can easily be considered. Scripting the changes to API becomes extremely simple with this approach and the best thing is the task is very quick and reliable.

5. What are the advantages of using the Serverless approach?

Ans. The very first thing about this approach is simple operations which mean quick time to market and better sales. Users only need to pay when the code is running and thus a lot o cost can simply be saved which enhances profits. Also, managing the larger application components is not a big deal. In addition to this, there is no need to have the additional infrastructure. The biggest thing is users need not worry about the servers on which the code runs.

6. Is it possible to debug and troubleshoot the micro or small services?

Ans. Yes, it's possible. It can be performed even when the function is running and appropriate tasks are being performed.hat

7. Is there any disadvantage of using this approach too? What do you think?

Ans. Well, everything has its own pros and cons depending on the work we perform through it. When it comes to the serverless approach, the fact is applicable here too. In a few cases, there is a strict upper limit on vendor control in a serverless approach and this clearly means more downtime and thus losses. The loss of functionality and system limits are the other issues. Also, there is no dedicated hardware available for the serverless approach. Thus, performance and security become challenges at several stages. Sometimes customer errors can also create problems. The new deployment, as well as monitoring tools, become the only option when it comes to switching to Google Cloud functions.

Experiment 12

Aim: To create a Lambda function which will log “An Image has been added” once you add an object to a specific bucket in S3.

Theory:
AWS Lambda is a serverless, event-driven compute service that lets you run code for virtually any type of application or backend service without provisioning or managing servers. You can trigger Lambda from over 200 AWS services and software as a service (SaaS) application, and only pay for what you use.

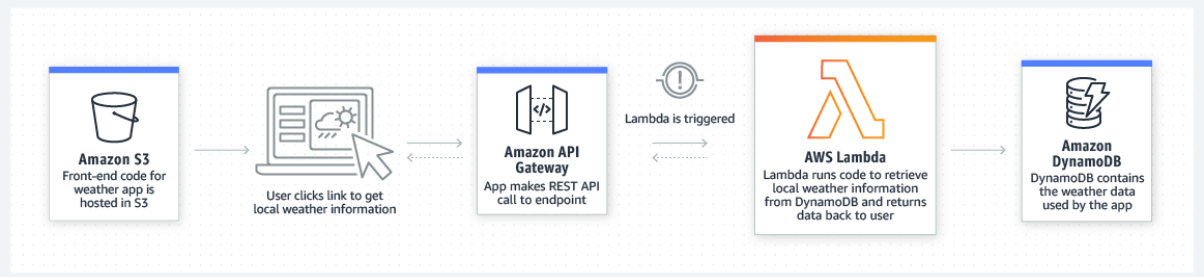
File processing

Use Amazon Simple Storage Service (Amazon S3) to trigger AWS Lambda data processing in real time after an upload, or connect to an existing Amazon EFS file system to enable massively parallel shared access for large-scale file processing.

Stream processing

Use AWS Lambda and Amazon Kinesis to process real-time streaming data for application activity tracking, transaction order processing, clickstream analysis, data cleansing, log filtering, indexing, social media analysis, IoT device data telemetry, and metering.

Web applications



Result:

Required Tools

- Node JS 12
- Serverless
- AWS CLI

Step 1: Install AWS CLI and Serverless Framework.

Step 2: Run the following command to generate sample code with serverless.

Step 3: Install dependencies

C:\Windows\system32\cmd.exe

```
C:\Users\ACER\Desktop\React\imageupload-lambda>sls create --template hello-world
Serverless: Generating boilerplate...
Serverless: Successfully generated boilerplate for template: "hello-world"
Serverless: NOTE: Please update the "service" property in serverless.yml with your service name

C:\Users\ACER\Desktop\React\imageupload-lambda>npm init -y
Wrote to C:\Users\ACER\Desktop\React\imageupload-lambda\package.json:

{
  "name": "imageupload-lambda",
  "version": "1.0.0",
  "description": "",
  "main": "handler.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC"
}



C:\Users\ACER\Desktop\React\imageupload-lambda>npm install busboy && uuid && jimp && aws-sdk
npm notice created a lockfile as package-lock.json. You should commit this file.
npm WARN imageupload-lambda@1.0.0 No description
npm WARN imageupload-lambda@1.0.0 No repository field.

+ busboy@0.3.1
added 3 packages from 1 contributor and audited 3 packages in 0.865s
found 0 vulnerabilities

'uuid' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\ACER\Desktop\React\imageupload-lambda>
```

Step 4: Create the function to decode the multipart/form-data

formParser.js - Notepad

File Edit Format View Help

```
const Busboy = require('busboy');

module.exports.parser = (event, fileZise) =>
    new Promise((resolve, reject) => {
    const busboy = new Busboy({
        headers: {
            'content-type':
            event.headers['content-type'] || event.headers['Content-Type']
        },
        limits: {
            fileZise
        }
    });

    const result = {
        files: []
    };

    busboy.on('file', (fieldname, file, filename, encoding, mimetype) => {
        const uploadFile = {}
        file.on('data', data => {
            uploadFile.content = data
        });
        file.on('end', () => {
            if (uploadFile.content) {
                uploadFile.filename = filename
                uploadFile.contentType = mimetype
                uploadFile.encoding = encoding
                uploadFile.fieldname = fieldname
                result.files.push(uploadFile)
             }
        })
    })

    busboy.on('field', (fieldname, value) => {
        result[fieldname] = value
```

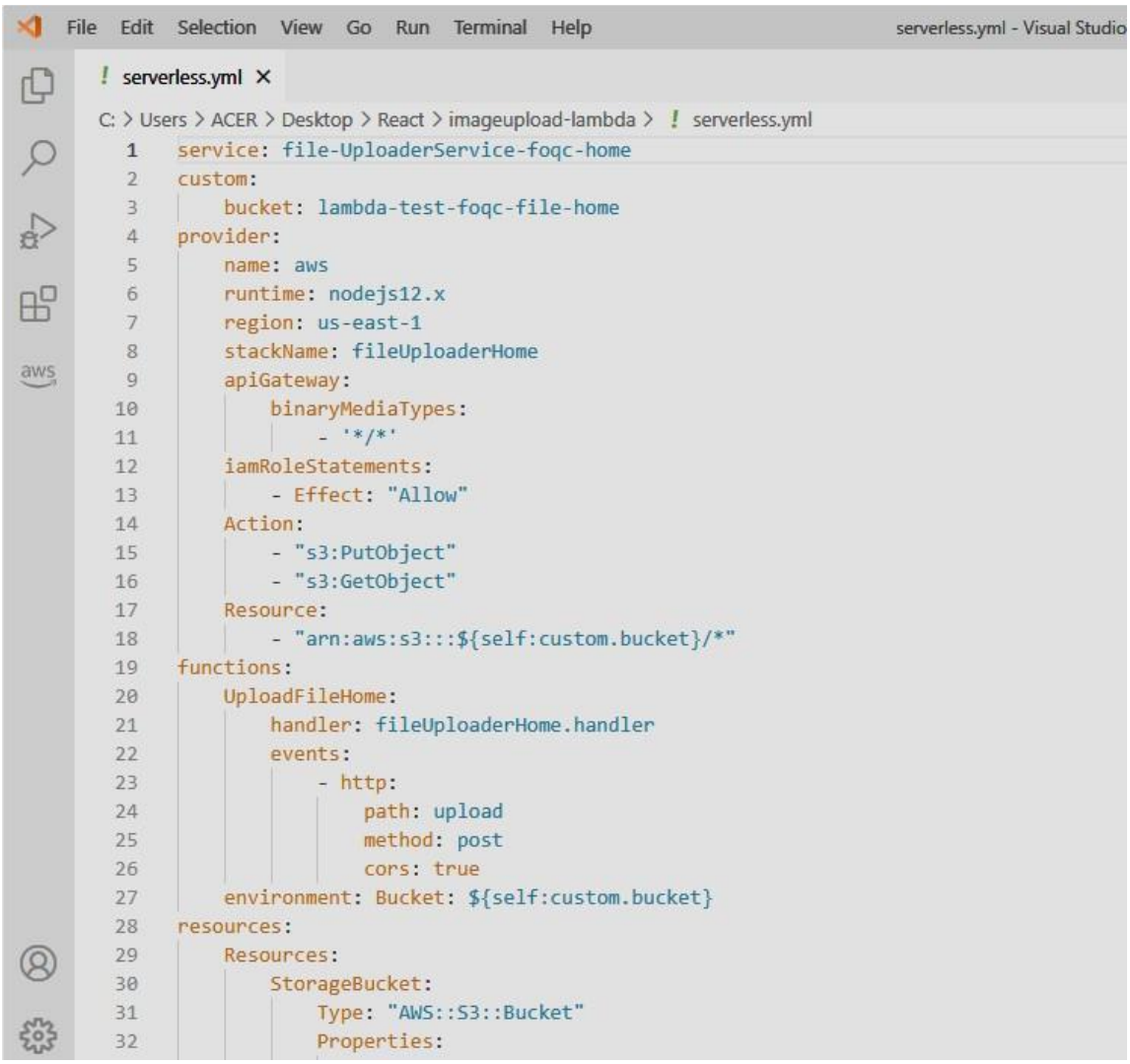
Step 5: Function that will process and upload the images to S3

fileUploaderHome.js - Notepad

File Edit Format View Help

```
"use strict";
const AWS = require("aws-sdk")
const uuid = require("uuid/v4")
const Jimp = require("jimp")
const s3 = new AWS.S3()
const formParser = require("./formParser")

const bucket = process.env.Bucket
const MAX_SIZE = 4000000 // 4MB
const PNG_MIME_TYPE = "image/png"
const JPEG_MIME_TYPE = "image/jpeg"
const JPG_MIME_TYPE = "image/jpg"
const MIME_TYPES = [PNG_MIME_TYPE, JPEG_MIME_TYPE, JPG_MIME_TYPE]

module.exports.handler = async event => {
const getErrorMessage = message => ({ statusCode: 500, body: JSON.stringify( message })})

const isAllowedFile = (size, mimeType) => { // some validation code }

const uploadToS3 = (bucket, key, buffer, mimeType) =>
    new Promise((resolve, reject) => {
        s3.upload(
            { Bucket: bucket, Key: key, Body: buffer, ContentType: mimeType },
            function(err, data) {
                if (err) reject(err);
                resolve(data)
            })
    })

const resize = (buffer, mimeType, width) =>
    new Promise((resolve, reject) => {
        Jimp.read(buffer)
        .then(image => image.resize(width, Jimp.AUTO).quality(70).getBufferAsync(mimeType))
        .then(resizedBuffer => resolve(resizedBuffer))
        .catch(error => reject(error))
    })
```

Step 6: Build serverless.yml file.

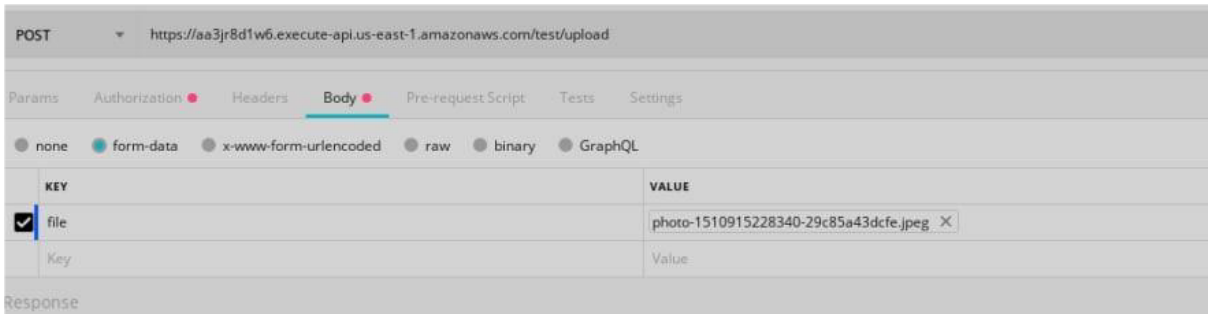


Step 7: Deploy the code using the following command.

- sls deploy --stage=test

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19043.1288]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ACER\Desktop\React\imageupload-lambda>sls deploy --stage=test

 Serverless Error ---------------------------------------

  Cannot parse "serverless.yml": bad indentation of a mapping entry in "C:\Users\ACER\Desktop\React\imageupload-lambda\serverless.yml" (27:24)

    24 |                path: upload
    25 |                method: post
    26 |                cors: true
    27 |     environment: Bucket: ${self:custom.bucket}
  -----------------------------^
    28 | resources:
    29 |     Resources:

  Get Support --------------------------------------------
     Docs:          docs.serverless.com
     Bugs:          github.com/serverless/serverless/issues
     Issues:        forum.serverless.com

  Your Environment Information -----------------------------
     Operating System:          win32
     Node Version:              14.17.5
     Framework Version:         2.64.1
     Plugin Version:            5.5.0
     SDK Version:               4.3.0
     Components Version:        3.17.1
```

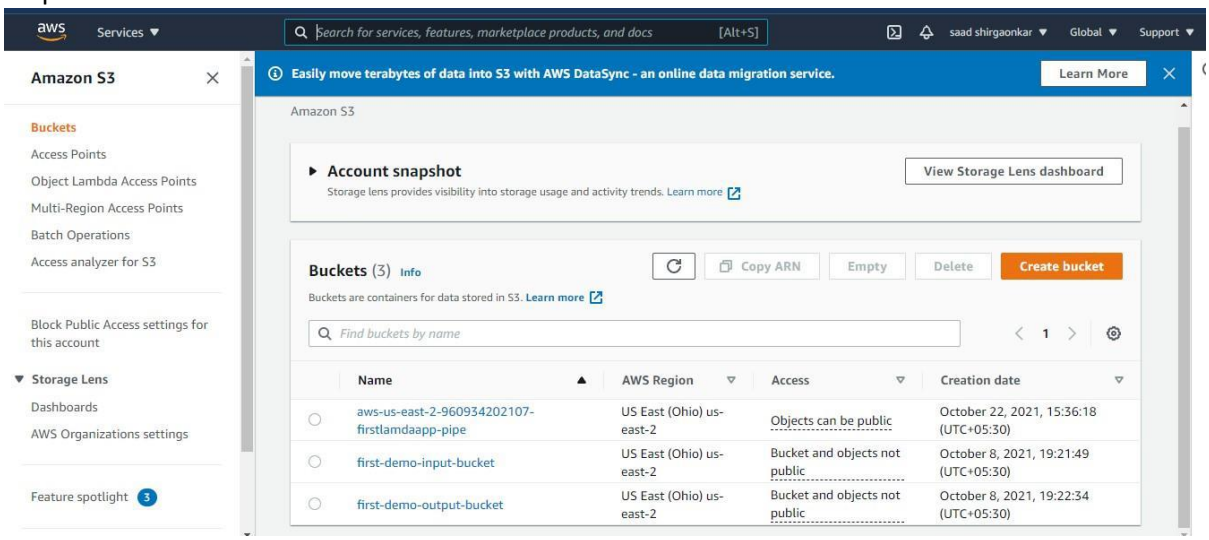
Step 8: Test Your API



Step 9: To conclude, in case you need to remove the service, run the following command.

• sls remove --stage=test

Step 10: Check the result in S3 Bucket.



Conclusion: Hence we have studied to create a Lambda function which will log “An Image has been added” once you add an object to a specific bucket in S3 and learned to implement them.

Questionnaire:

Q1. What makes Lambda a time-saving approach?

Ans. There are certain reasons for this. The one is it's possible to simply store everything in the local server memory. Also, data can be stored directly into the database without affecting the performance. In addition to this, testing is not much complicated. Integration testing can simply be made powerful through multiple vendors.

Q2. What is the time limit for execution in Lambda when you perform DDOS?

Ans. The time limit is 5 minutes.

Q3. What do you know about Zero downtime deployment?

Ans. Deployments are generally considered in the form of functions. AWS Lambda divides it into units in case they are complex. The fact here is app remains in offline mode during such a time period. However, the results are always good.

Q4. There are some of the very complex querying capabilities that need to be handled without having a warehouse? Which database do you consider during such a case?

Ans. The Amazon RDS is a good option as others such as ElasticCache suffer from some issues.RDS makes it easy to set up and manage every task reliably. Also, it is compatible with all modern tools.

Q5. Among On-demand and Reserved Instances, which one is better to impose a limit on expenditure when it comes to optimizing the speed on Lambda?

Ans. Reserved Instances are a better option.

Q6. What is EC2 service?

Ans. In Lambda, there is always a need to have scalable computing capacity while dealing with data in the cloud. EC2 is meant for the same purpose as a web service. Networking, as well as security, can easily be managed. Using minimal friction, configuring capacity can also be creased with EC2.

Q7. What do you know about AMI?

Ans. It stands for Amazon Machine Image and many times it is used in processed that is based on Lambda or even in conjunction with the same. Basically, it's a template that contains an application server, OS, or other applications. It is possible to create its copy in the cloud. It has several instances and running multiple instances is also possible. AMI can also run a virtual server in a cloud.

Q8. What do you know about Auto-Scaling?

Ans. It is basically a feature in the Amazon Web Services that simply enables you to automatically configure and spin the novel instances. The good thing is there is no need for you to interfere at any stage. However, users can monitor everything through metrics and thresholds. To enable this task, you simply need to cross a threshold and you can see without any interference, the instances have scaled horizontally.

Q9. Do you think there is a relation between Instance and AMI?

Ans. Yes, they are associated with each other. Lambda offers a query API that is good in terms of query parameters. Requests such as HTTPs can simply be handled and managed.