Gautham Srinivasan

IT Analyst - DTCC

23 Nov 2024

DevSecOps Pipeline with AWS, Kubernetes, Jenkins, and Security Integrations

Project Overview

The project aims to implement a **DevSecOps pipeline** that integrates key tools such as **Jenkins**, **Terraform**, **AWS**, **Kubernetes**, **SonarCloud**, **Snyk**, and **OWASP ZAP** to automate the CI/CD process with a focus on **security**. The goal is to establish a seamless integration between infrastructure provisioning, code quality checks, security scanning, and deployment, all while ensuring that vulnerabilities are tracked and addressed efficiently through **Jira** automation.

Prerequisites

- **AWS Account**: Active AWS account with IAM user permissions to manage resources.
- **Terraform**: Installed locally for provisioning AWS infrastructure.
- **Jenkins**: Installed with necessary plugins (AWS, Docker, Kubernetes, etc.).
- **GitHub**: Repositories for application code and Jenkinsfiles.
- **SonarCloud**: Account and API token for static code analysis.
- **Snyk**: Account and API token for vulnerability scanning.
- **Docker**: Installed for container image management and deployment.
- **Kubernetes**: kubectl installed for EKS cluster management.
- **Jira**: Account for issue tracking and API access for automation..

Activities Outline

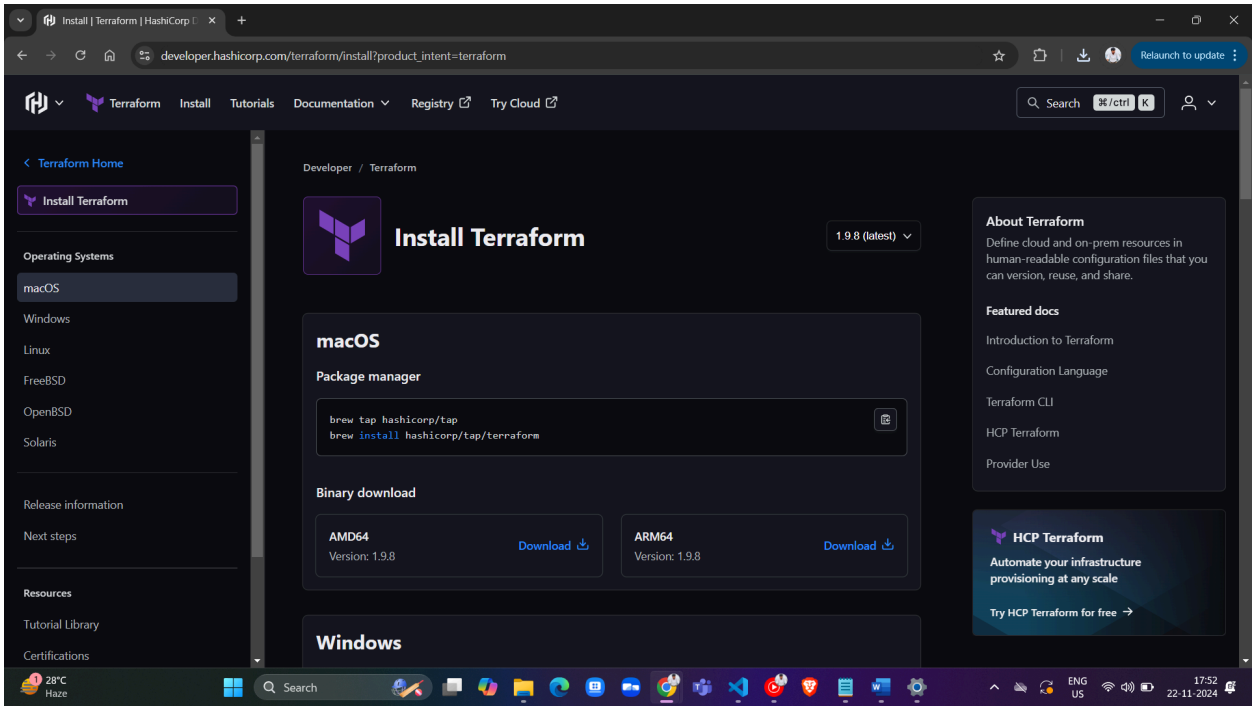
1. **Infrastructure Setup**: Provision AWS resources using Terraform.
2. **Jenkins Setup**: Configure Jenkins as the CI/CD orchestrator.
3. **SAST Integration**: Static Application Security Testing using SonarQube.
4. **SCA Integration**: Software Composition Analysis with Snyk.
5. **Dockerization & ECR**: Containerization and image management with AWS Elastic Container Registry (ECR).
6. **Kubernetes Deployment (EKS)**: Deploy and manage applications on EKS.
7. **DAST Integration**: Dynamic Application Security Testing with OWASP ZAP.
8. **Jira Integration**: Automate ticket creation for vulnerabilities.

Procedures

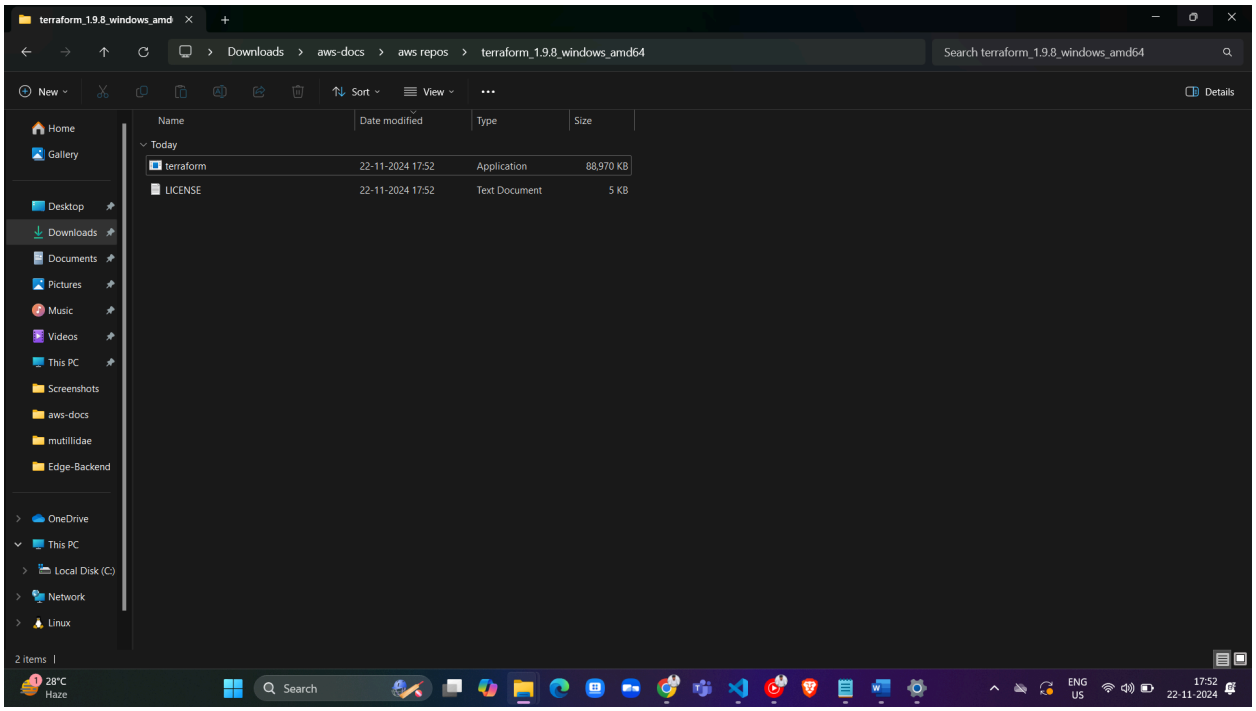
Step 1. Initial Setup

Install Terraform

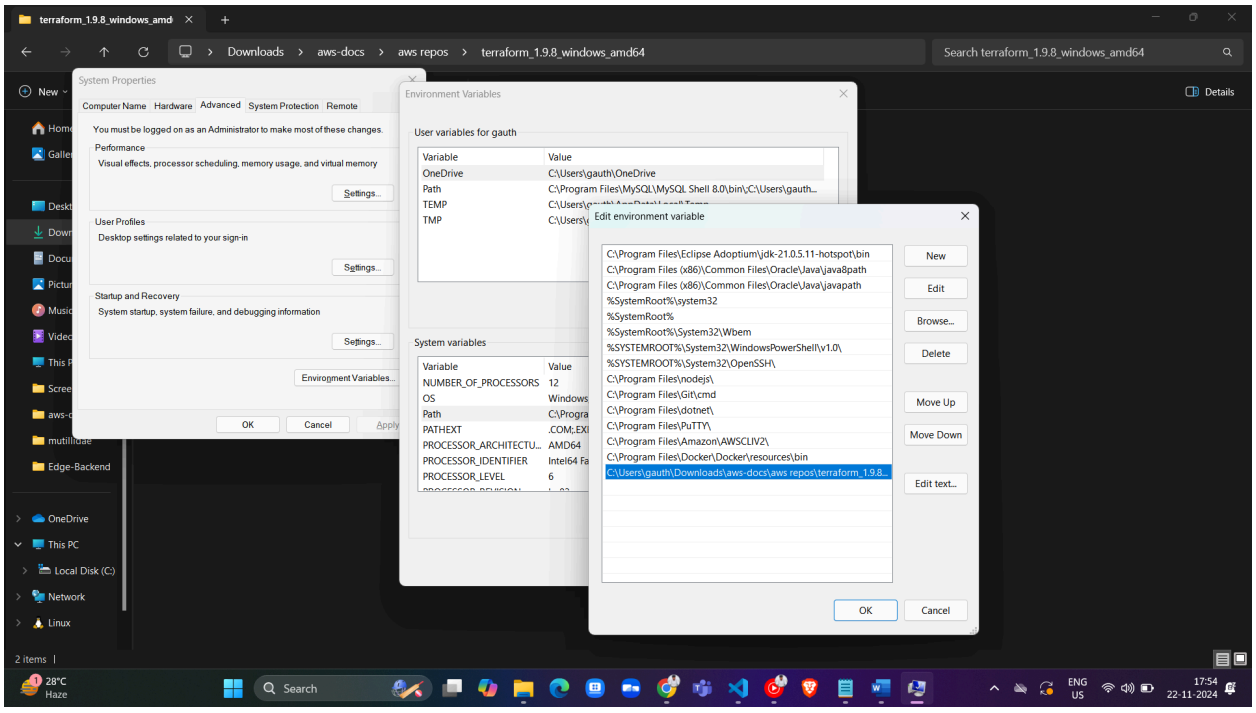
1. Download Terraform from the official Terraform website.(https://developer.hashicorp.com/terraform/install?product_intent=terraform)



2. Install it on your local machine.

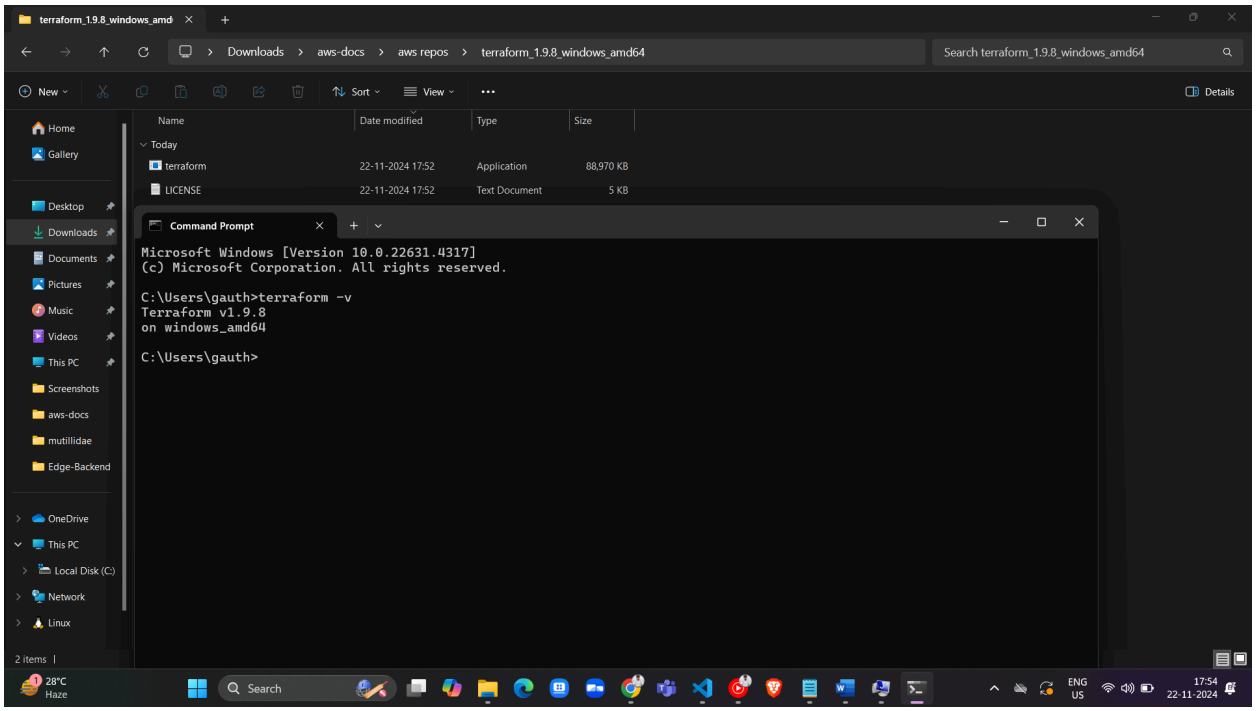


3. Add the Terraform executable to your system's environment path.



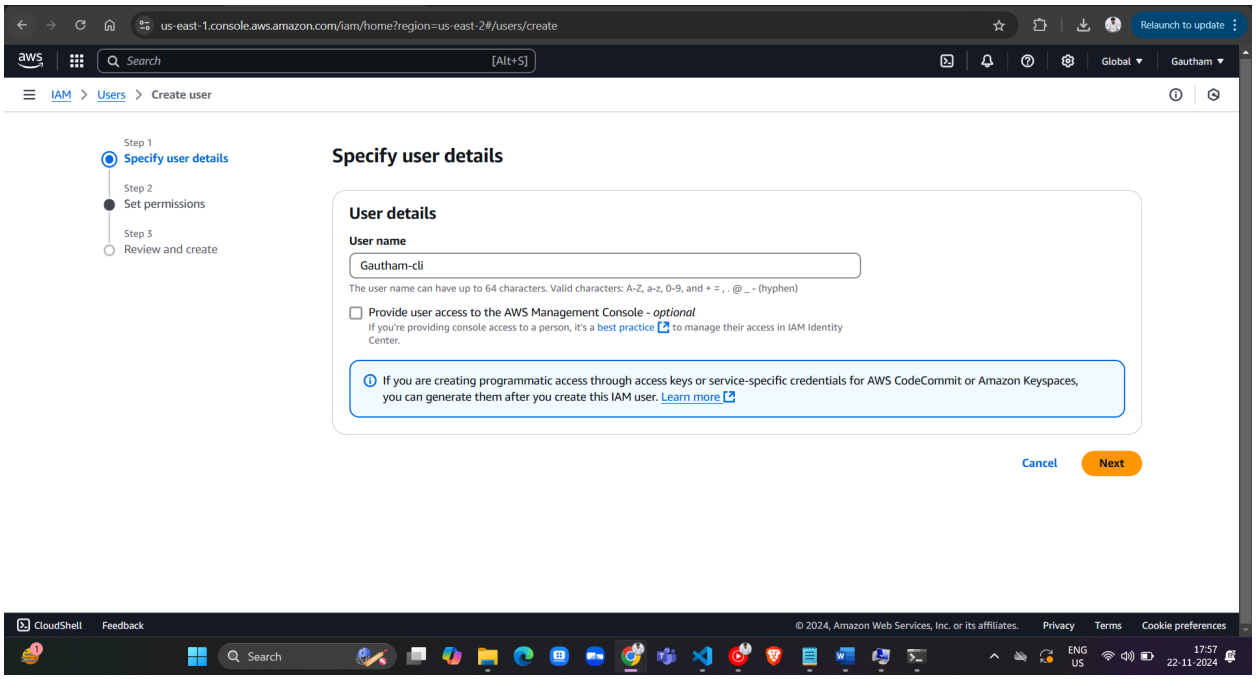
4. Verify the installation:

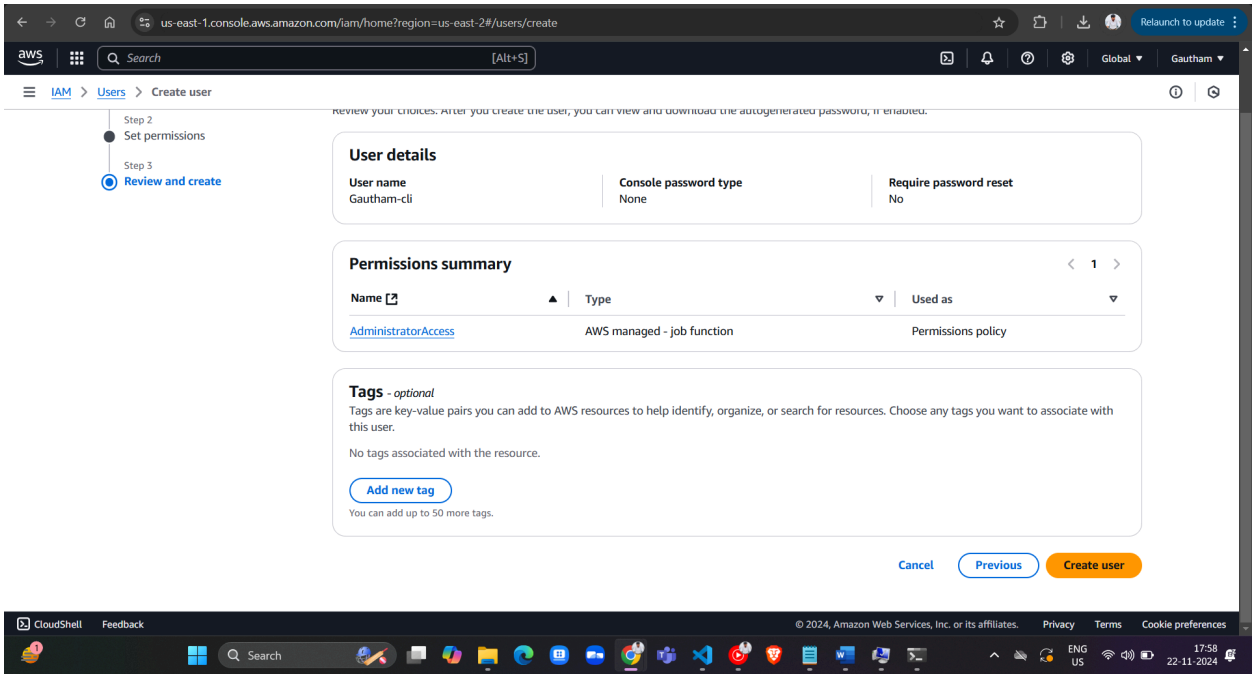
```
terraform -v
```



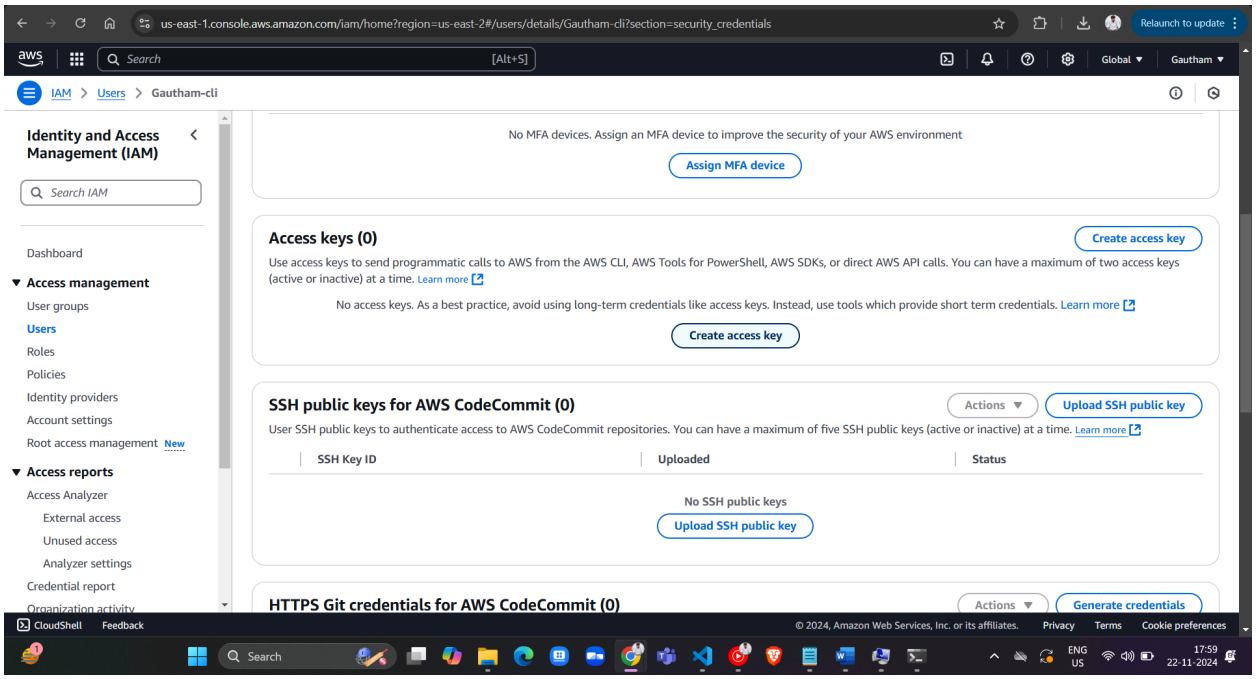
Step 2. Configure AWS CLI

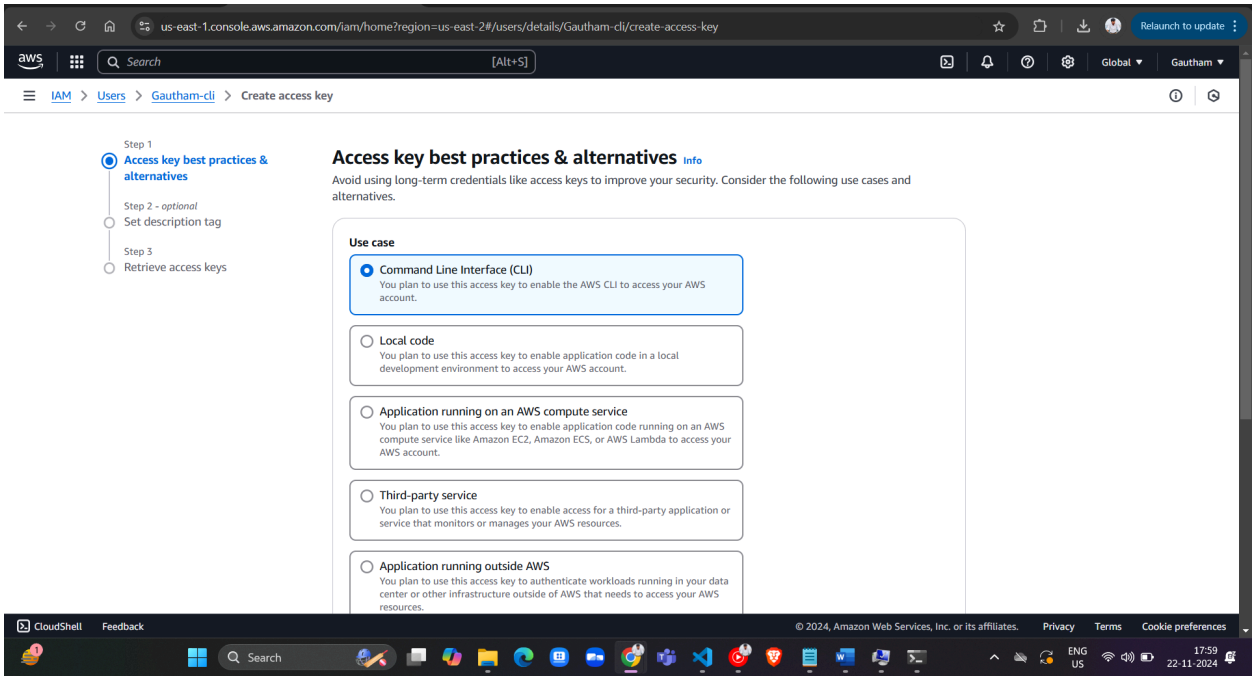
1. Create an IAM user with administrator privileges and CLI access.



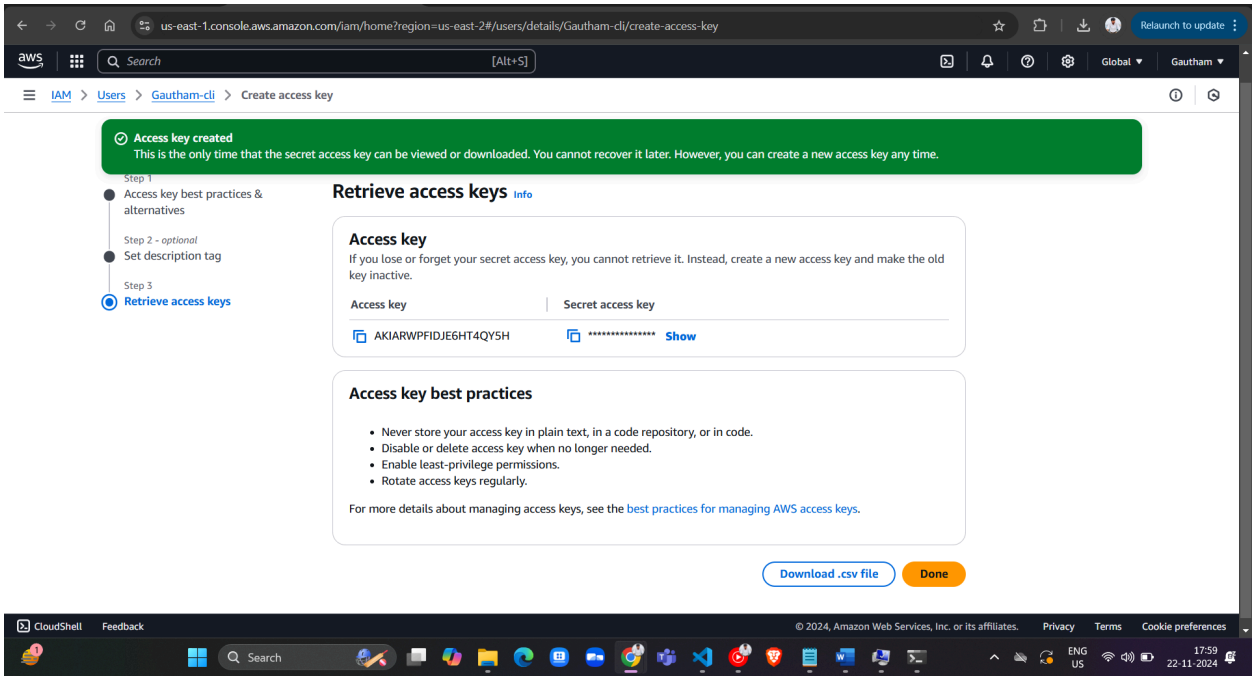


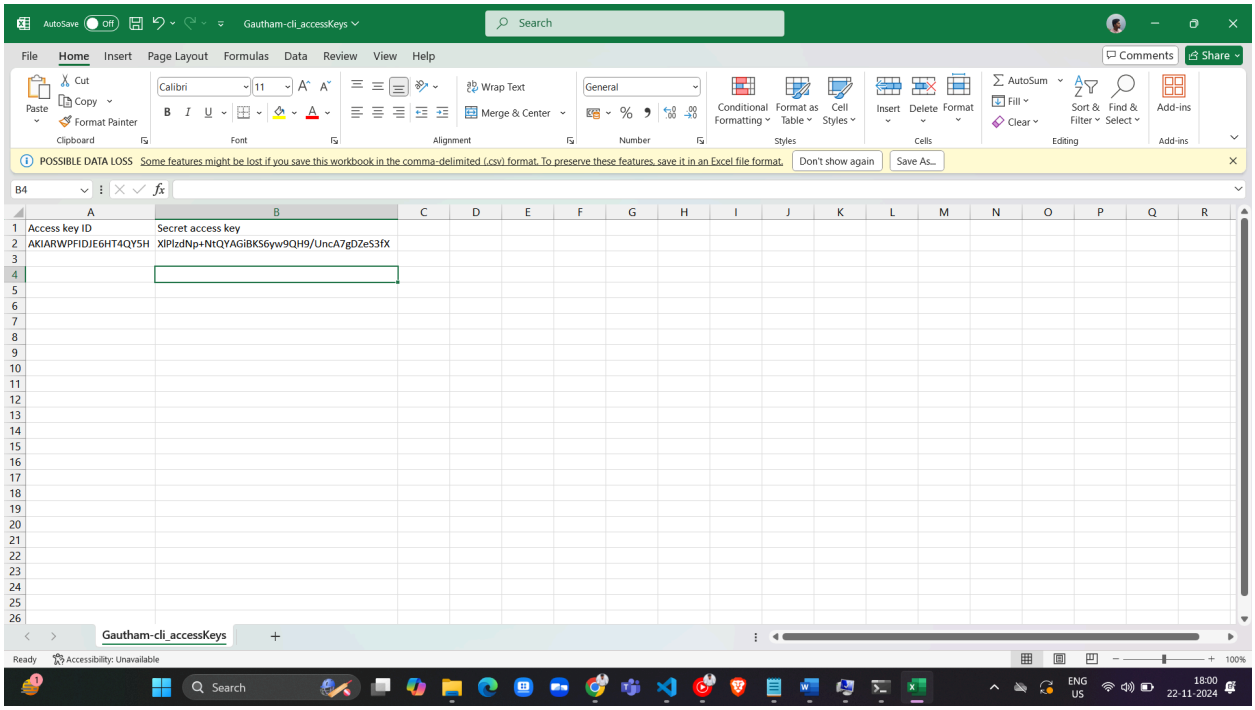
2. Click create access key for the created user



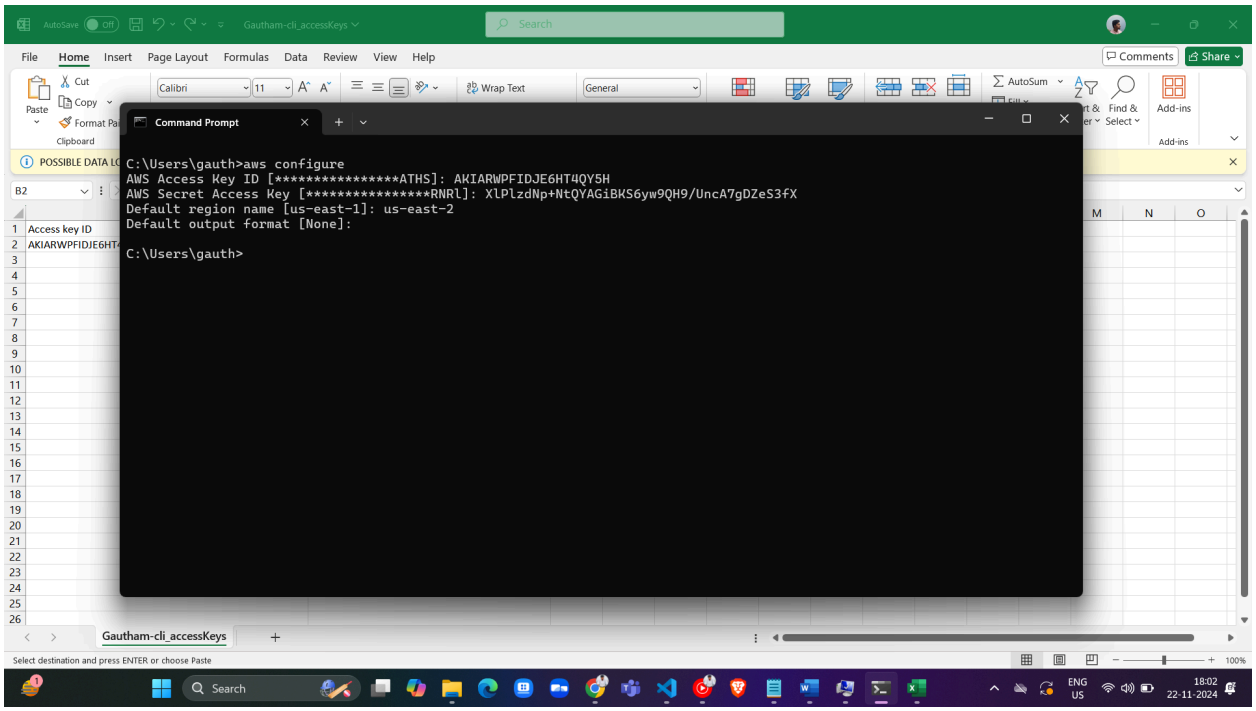


3. Download the credentials file (.csv) containing the Access Key ID and Secret Access Key.





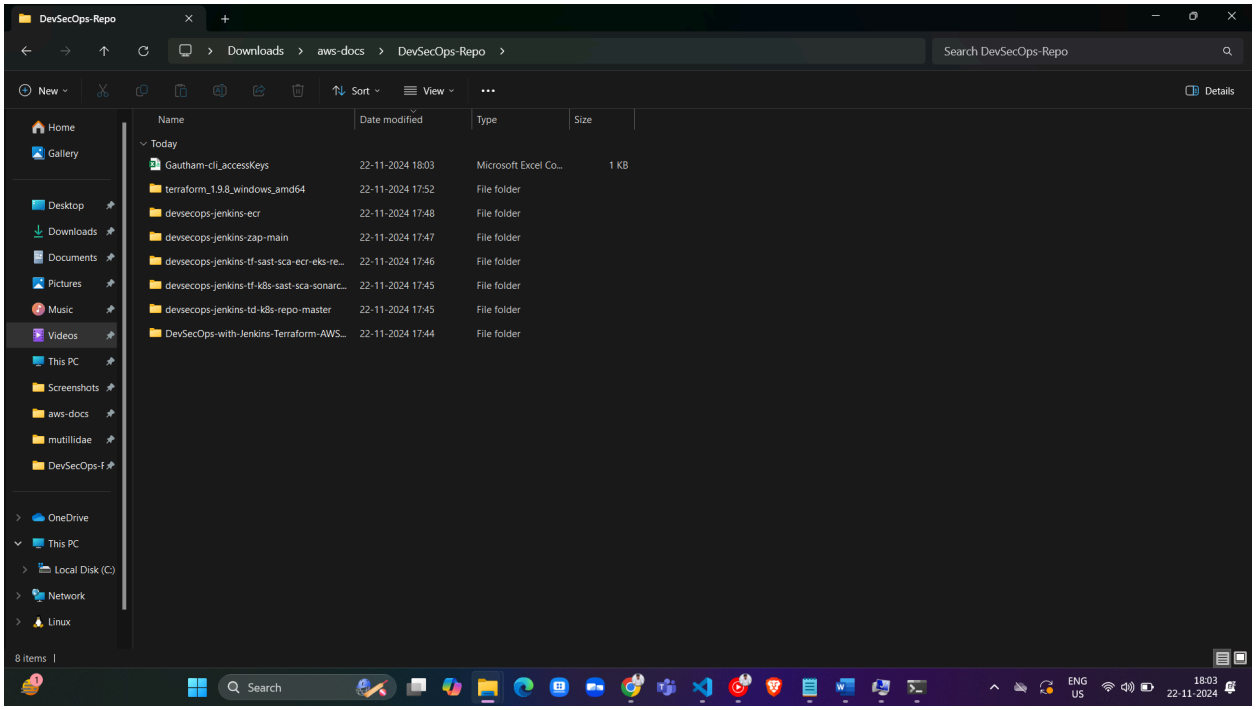
4. Configure AWS CLI: **aws configure**
5. Enter the Access Key ID.
6. Enter the Secret Access Key.
7. Set the default region: `us-east-2`.



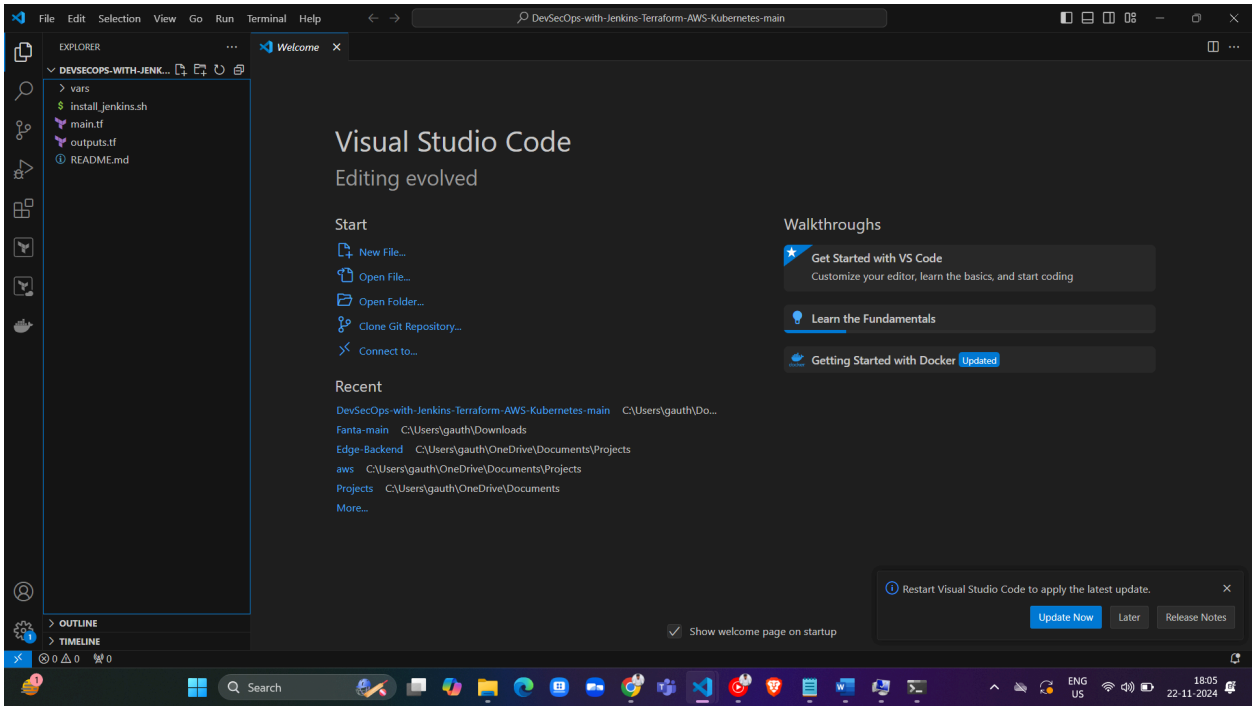
Step 3. Infrastructure Setup with Terraform

Clone Repository

1. Create a folder DevSecOps-Repo on your desktop.

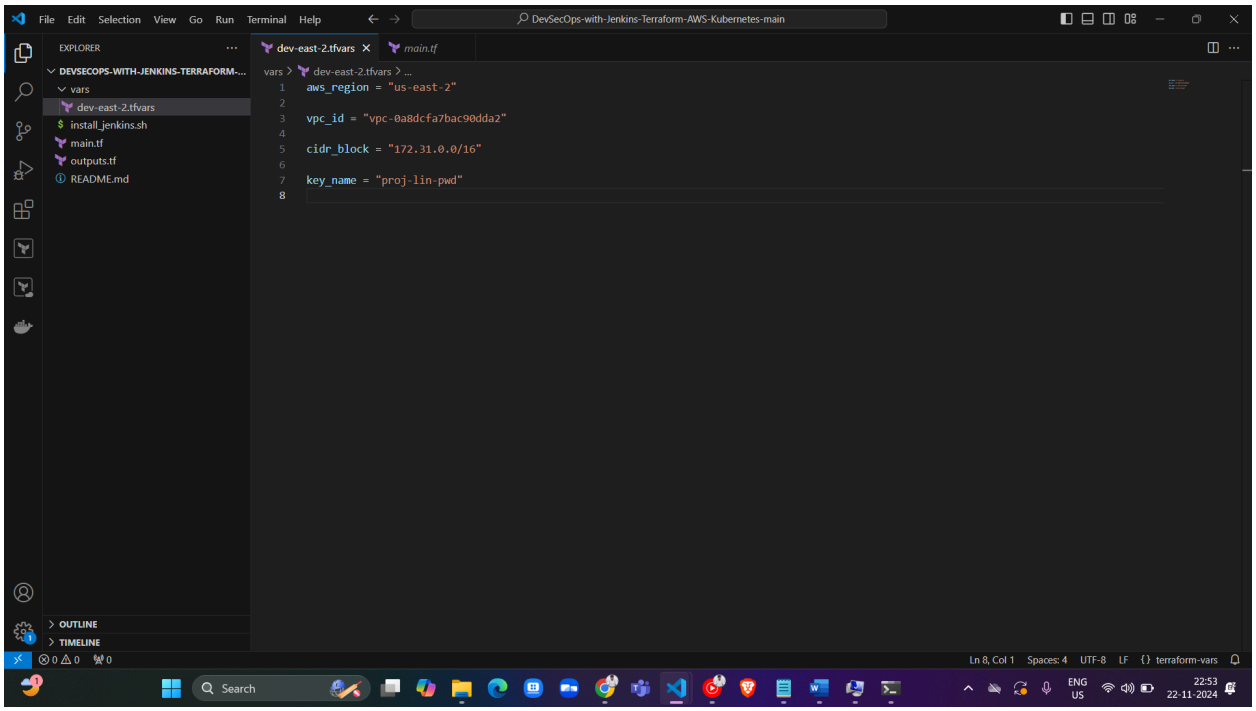


2. Open Git Bash in the folder and clone the repository:
   git clone
   https://github.com/Akshiv20/DevSecOps-with-Jenkins-Terraform-AWS-Kubernetes.git
3. Open the folder in Visual Studio Code.



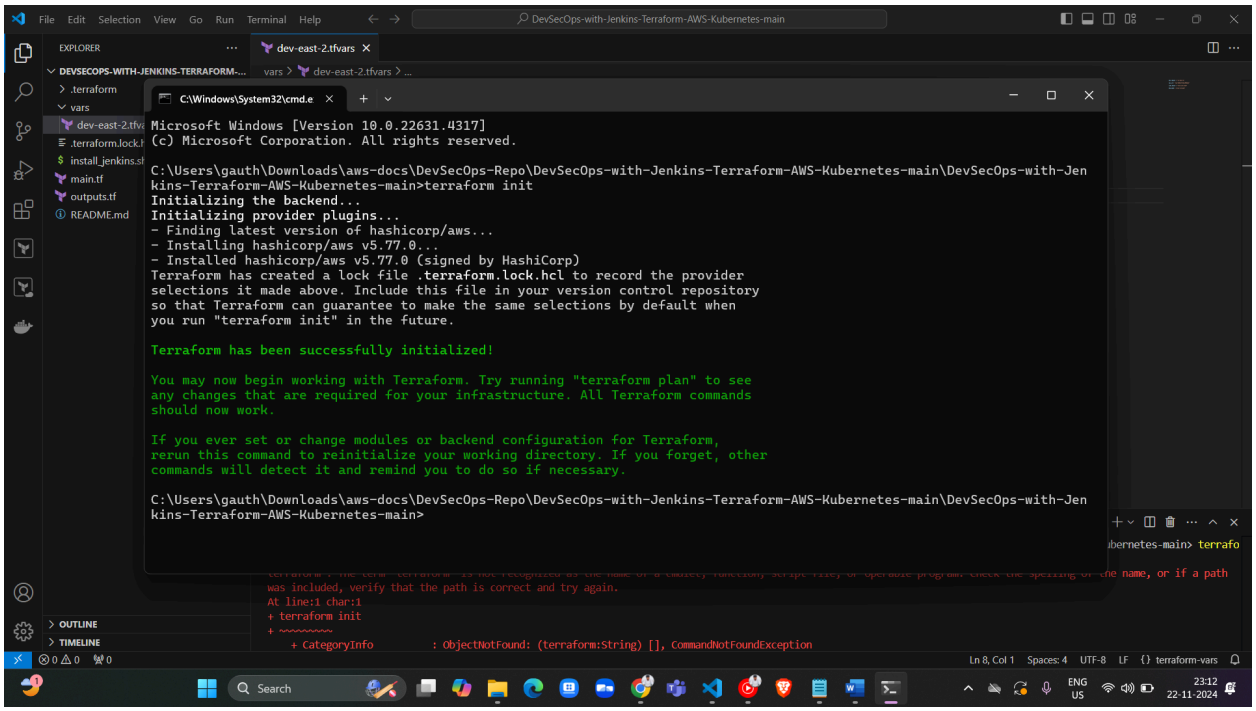
Configure Variables

1. Navigate to `vars/dev-east-2.tfvars` and set the following values:
   - VPC ID
   - IPv4 CIDR block
   - KeyPair value

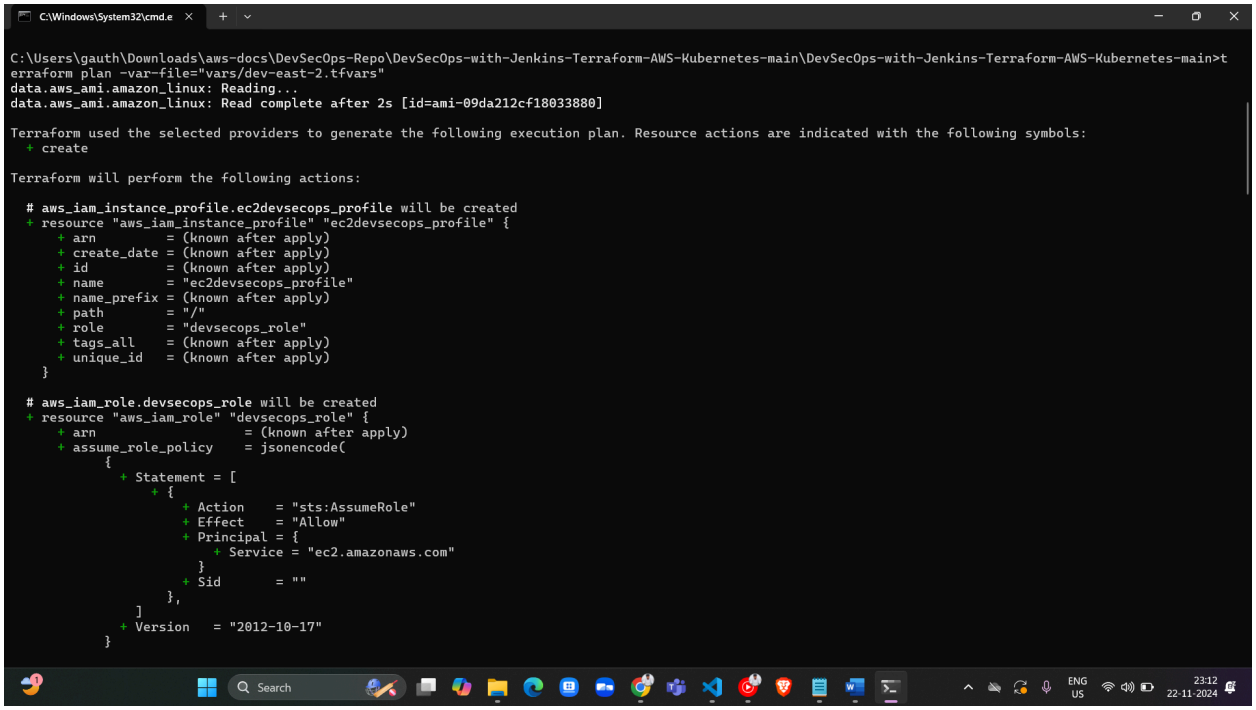


Deploy Terraform Resources

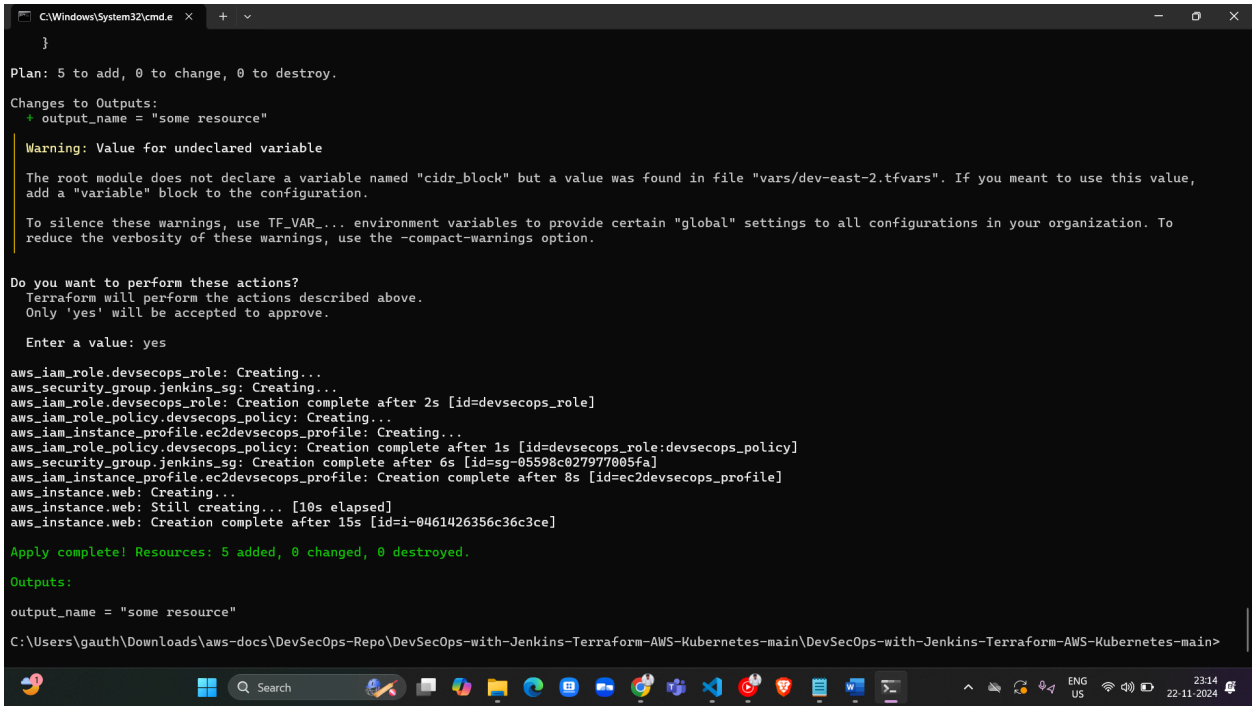
- Initialize Terraform:
  `terraform init`



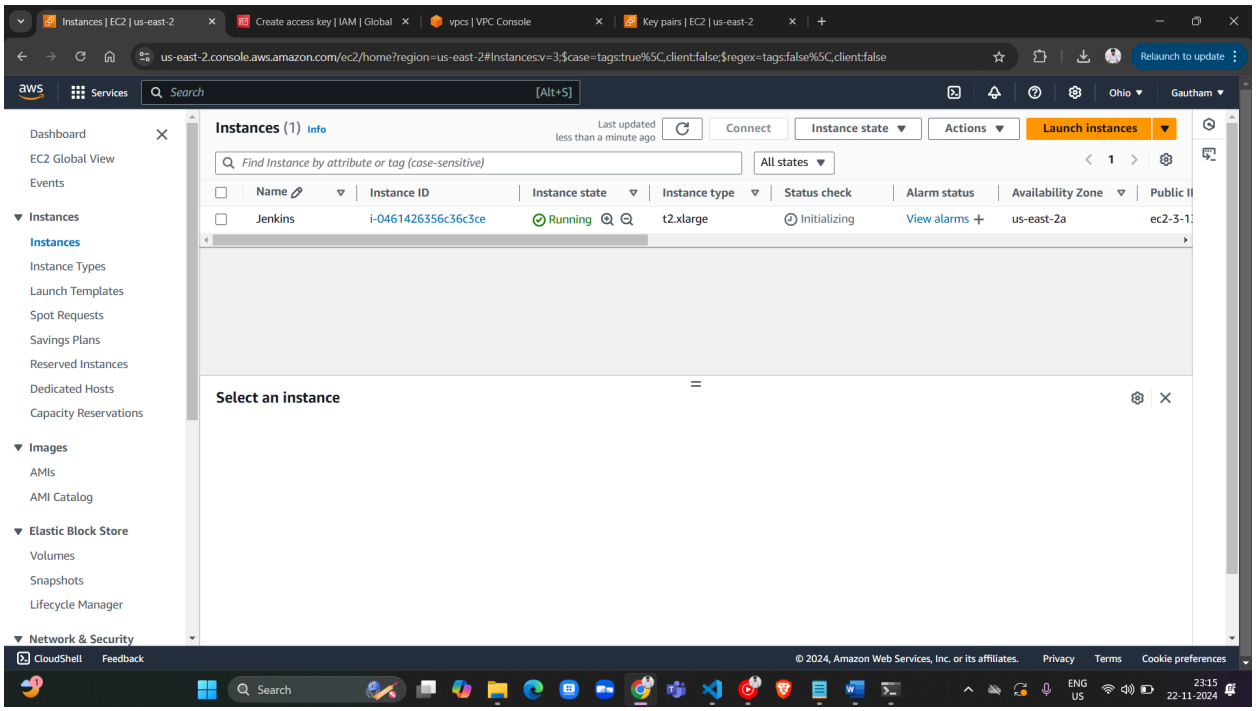
- Review the plan:
  `terraform plan -var-file="vars/dev-east-2.tfvars"`



- Apply the configuration:
  `terraform apply -var-file="vars/dev-east-2.tfvars"`

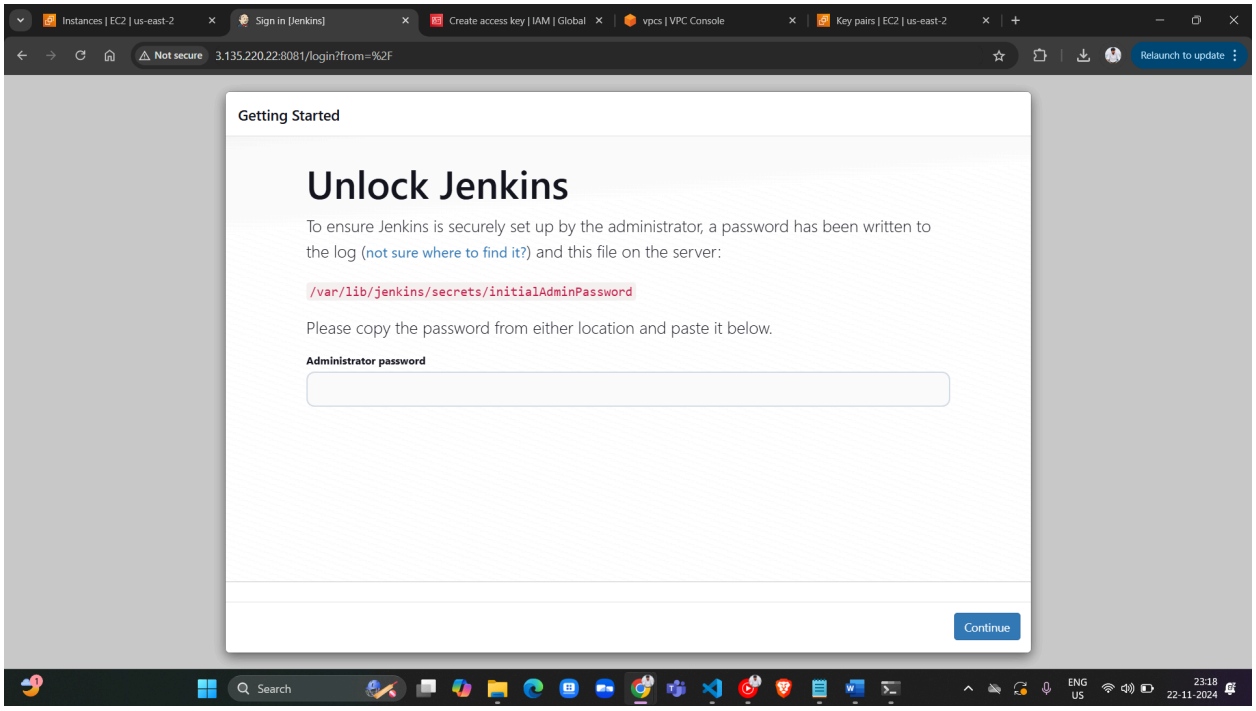


- Verify the created resources in the AWS Console.

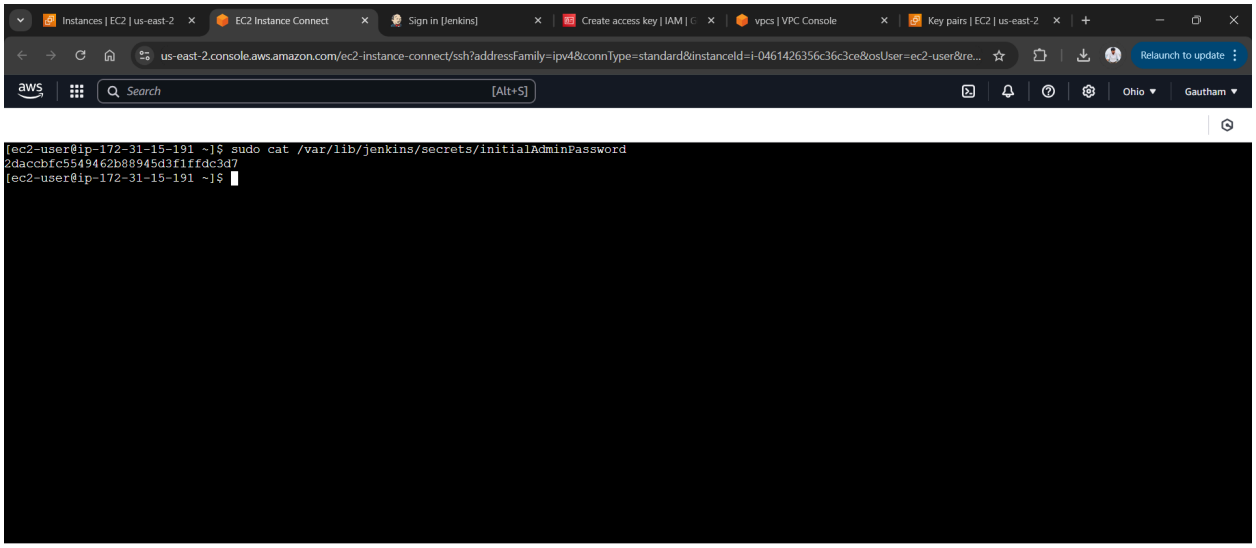


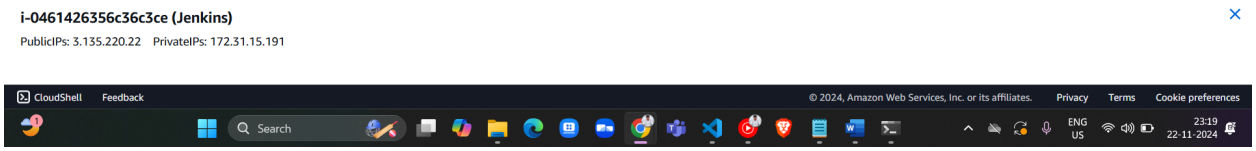
Access Jenkins

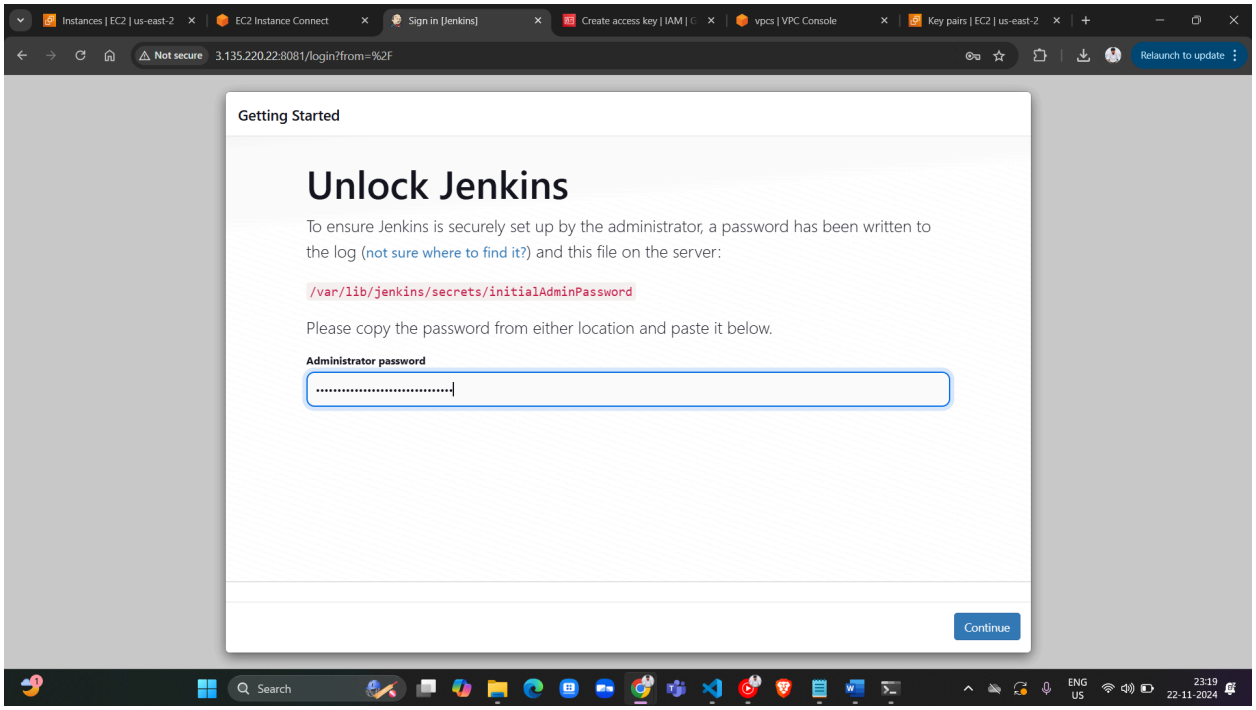
1. Open the Jenkins instance via `<public_ip>:8081`.



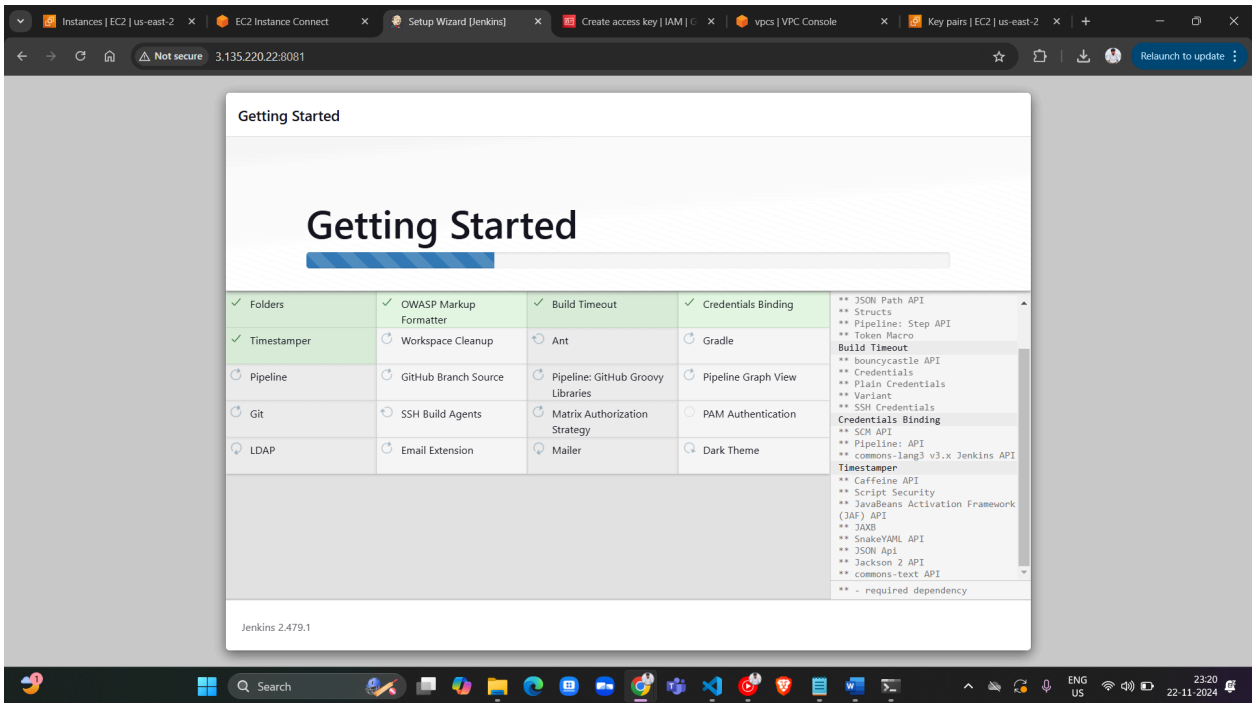
2. Retrieve the Jenkins password from the instance:
   `sudo cat /var/lib/jenkins/secrets/initialAdminPassword`

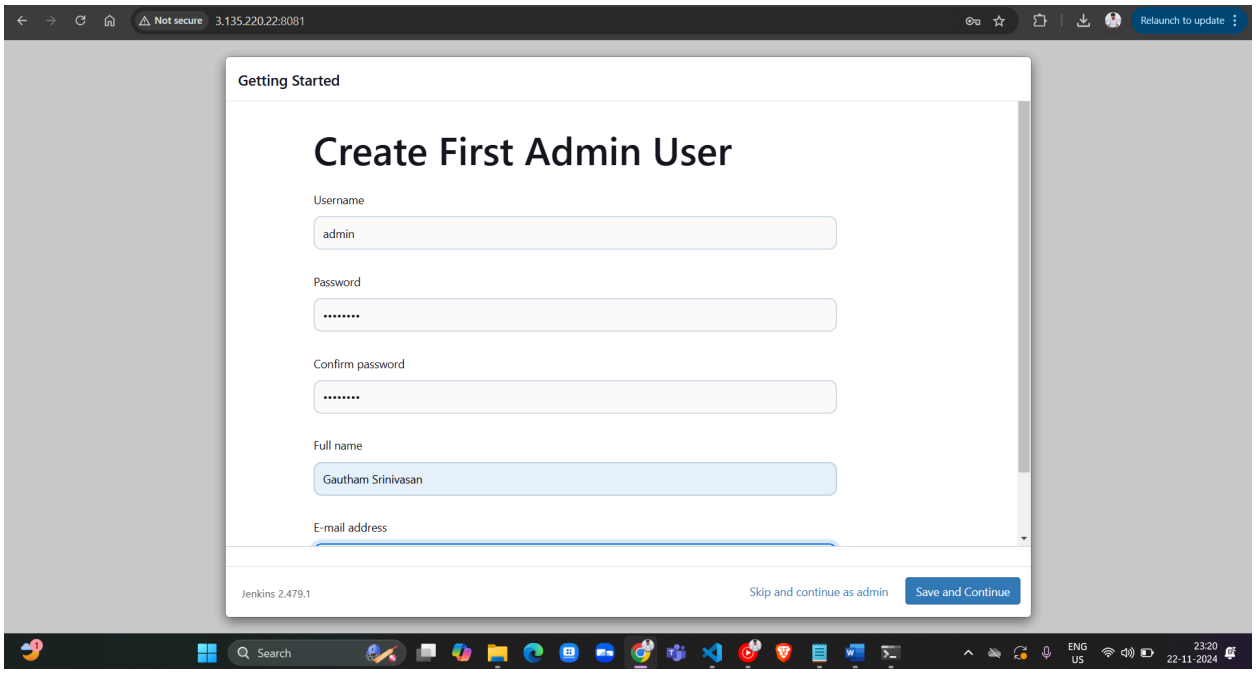


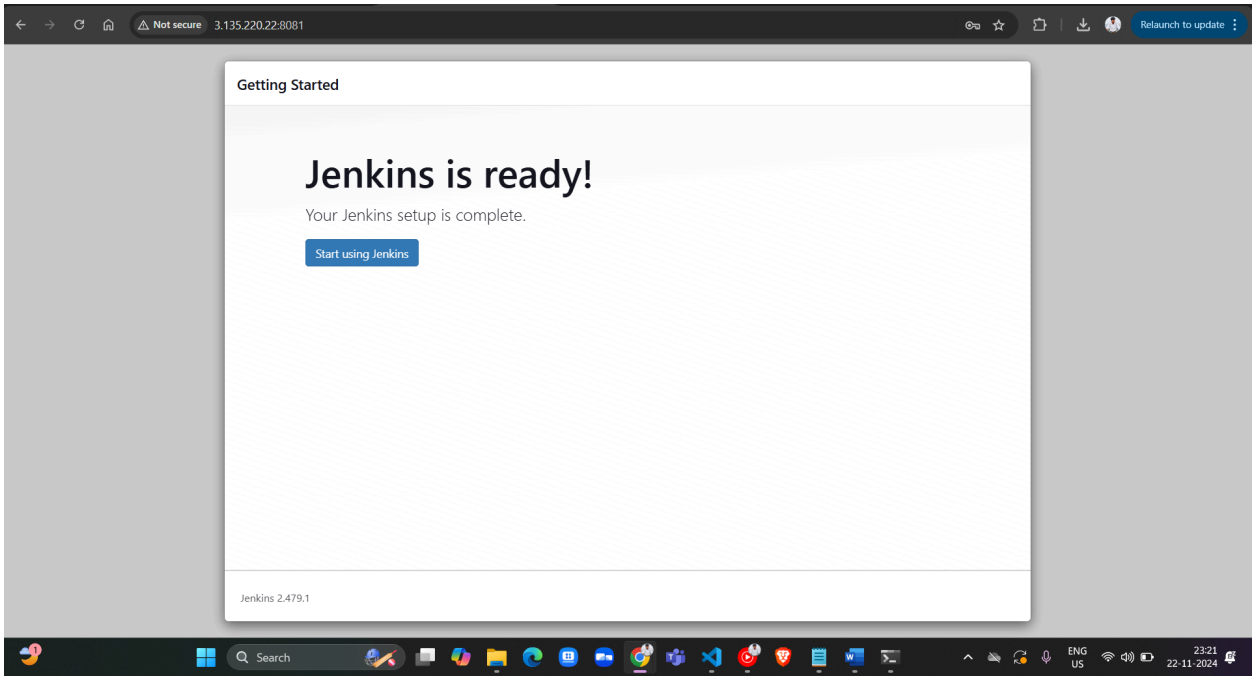


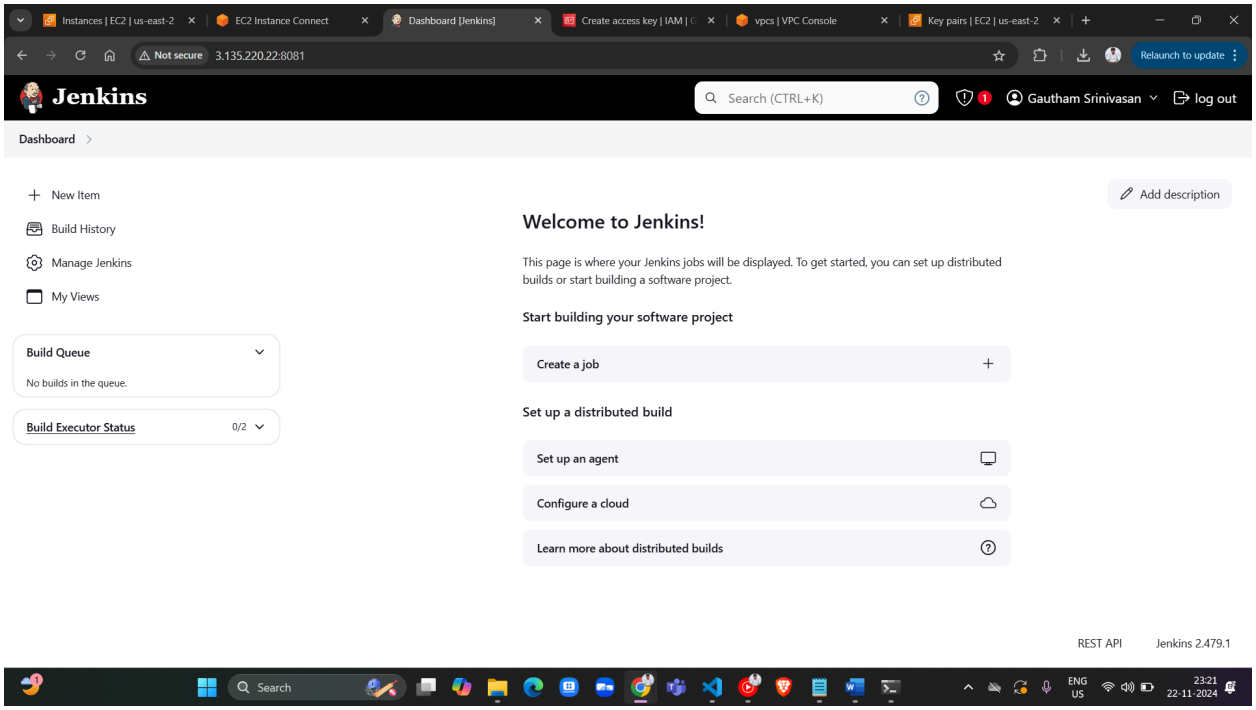


3. Complete Jenkins setup with suggested plugins.


Instances | EC2 | us-east-2
EC2 Instance Connect
Setup Wizard [Jenkins]
Create access key | IAM |
vpcs | VPC Console
Key pairs | EC2 | us-east-2
Not secure
3.135.220.22:8081
Relaunch to update
Getting Started
Getting Started
Folders
OWASP Markup Formatter
Build Timeout
Credentials Binding
Timestamper
Workspace Cleanup
Ant
Gradle
Pipeline
GitHub Branch Source
Pipeline: GitHub Groovy Libraries
Pipeline Graph View
Git
SSH Build Agents
Matrix Authorization Strategy
PAM Authentication
LDAP
Email Extension
Mailer
Dark Theme
** JSON Path API
** Structs
** Pipeline: Step API
** Token Macro
Build Timeout
** bouncycastle API
** Credentials
** Plain Credentials
** Variant
** SSH Credentials
Credentials Binding
** SCM API
** Pipeline: API
** commons-lang3 v3.x Jenkins API
Timestamper
** Caffeine API
** Script Security
** JavaBeans Activation Framework (JAF) API
** JAXB
** SnakeYAML API
** JSON Api
** Jackson 2 API
** commons-text API
** - required dependency
Jenkins 2.479.1
Search
ENG
US
23:20
22-11-2024


Not secure
3.135.220.22:8081
Relaunch to update
Getting Started
Create First Admin User
Username
admin
Password
Confirm password
Full name
Gautham Srinivasan
E-mail address
Jenkins 2.479.1
Skip and continue as admin
Save and Continue
Search
ENG
US
23:20
22-11-2024



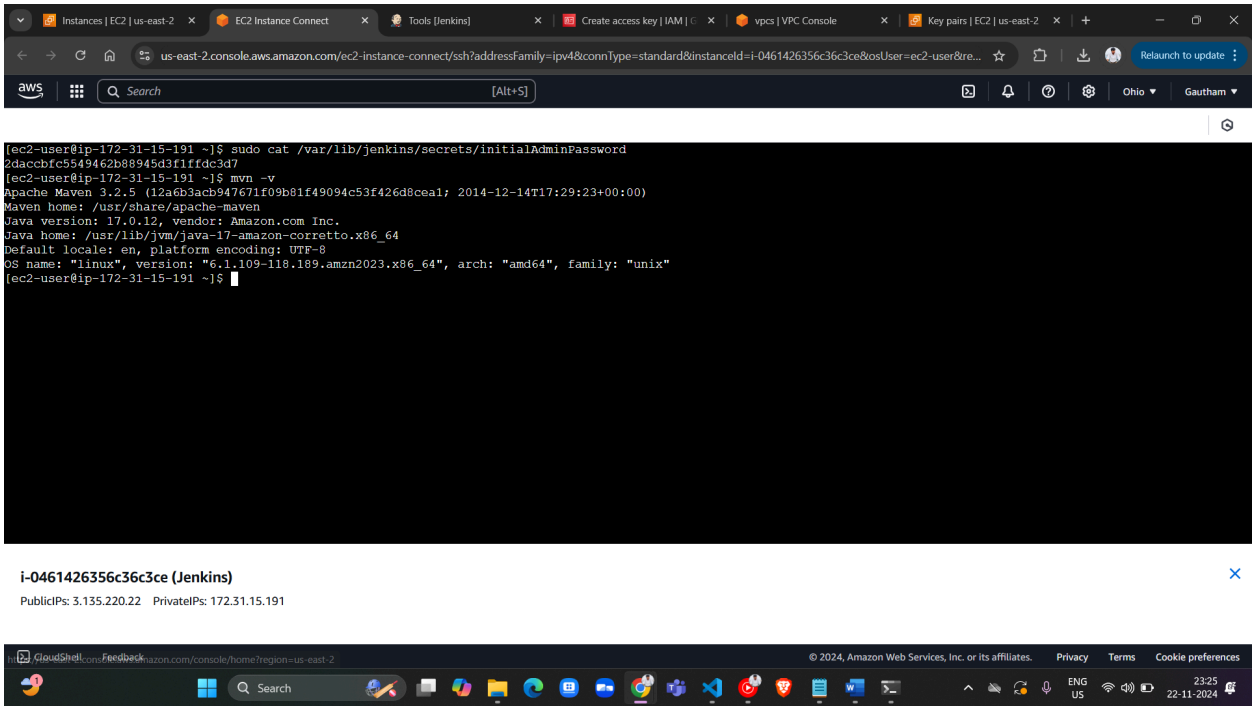


Step 4. Configure Jenkins

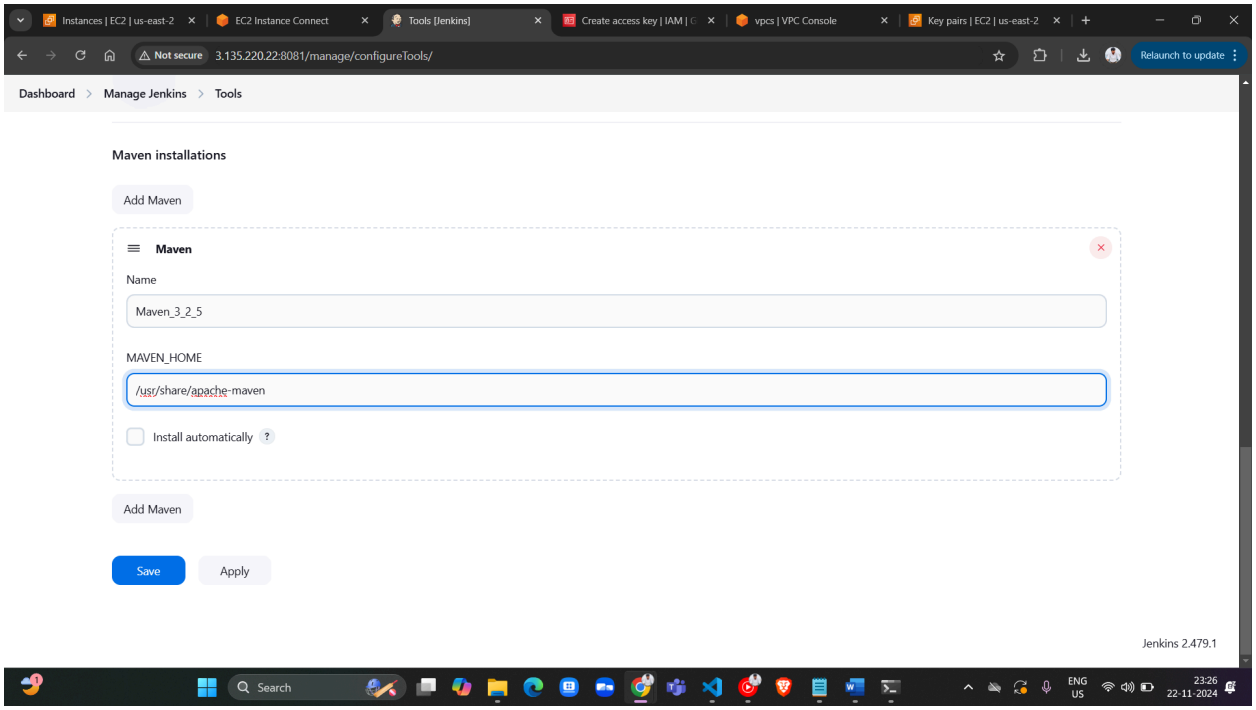
Set Up Maven

1. Go to Jenkins > Manage Jenkins > Global Tool Configuration.
2. Add a new Maven installation:

- Name: `Maven_3_2_5`
  Check for maven path using `maven -v`

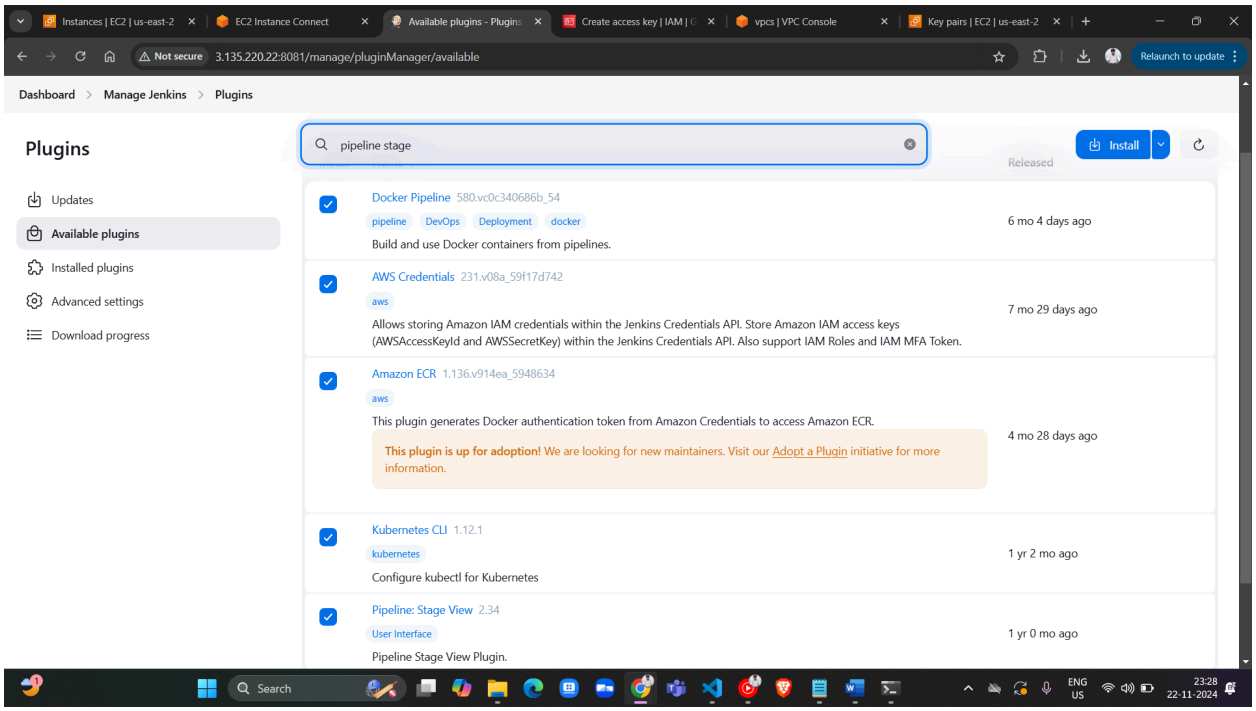


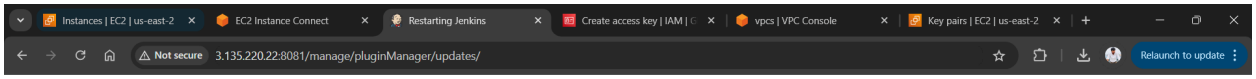
- Maven home path: `/usr/share/apache-maven`



Install Plugins

1. Install the following Jenkins plugins:
    - Docker Pipeline
    - AWS Credentials
    - Amazon ECR
    - Kubernetes CLI
    - Pipeline Stage View

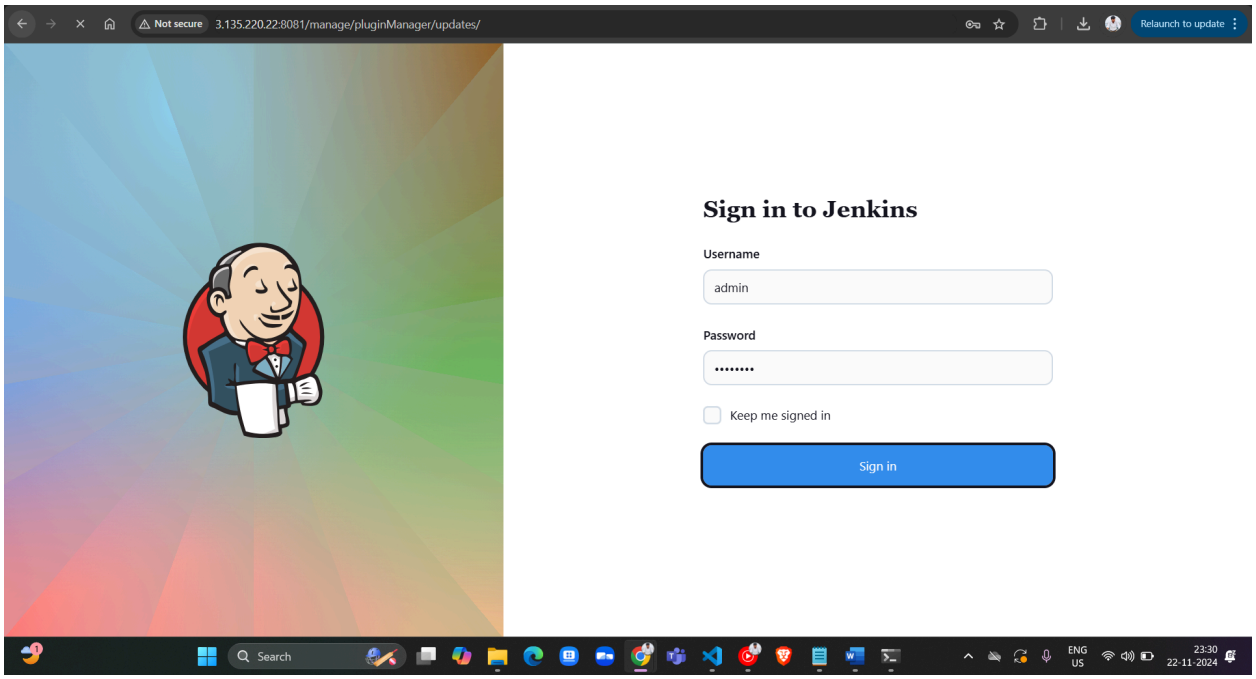




Jenkins is restarting

Your browser will reload automatically when Jenkins is ready

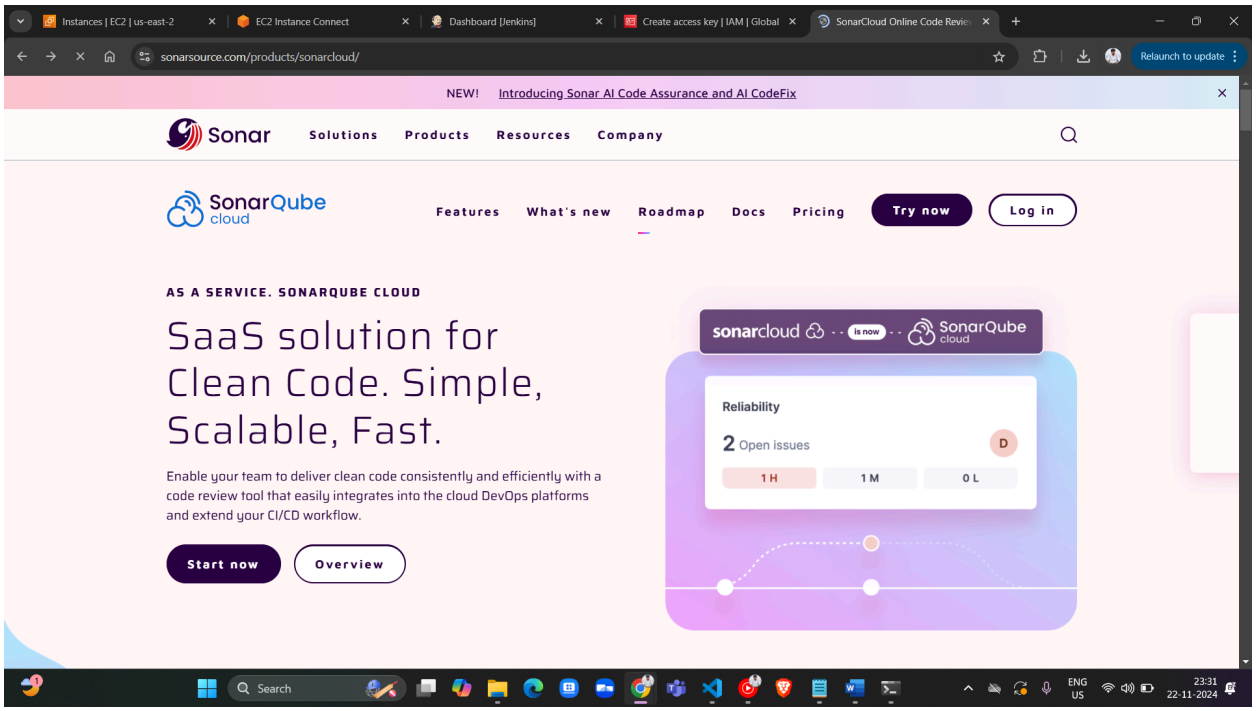
Safe Restart
Builds on agents can usually continue.

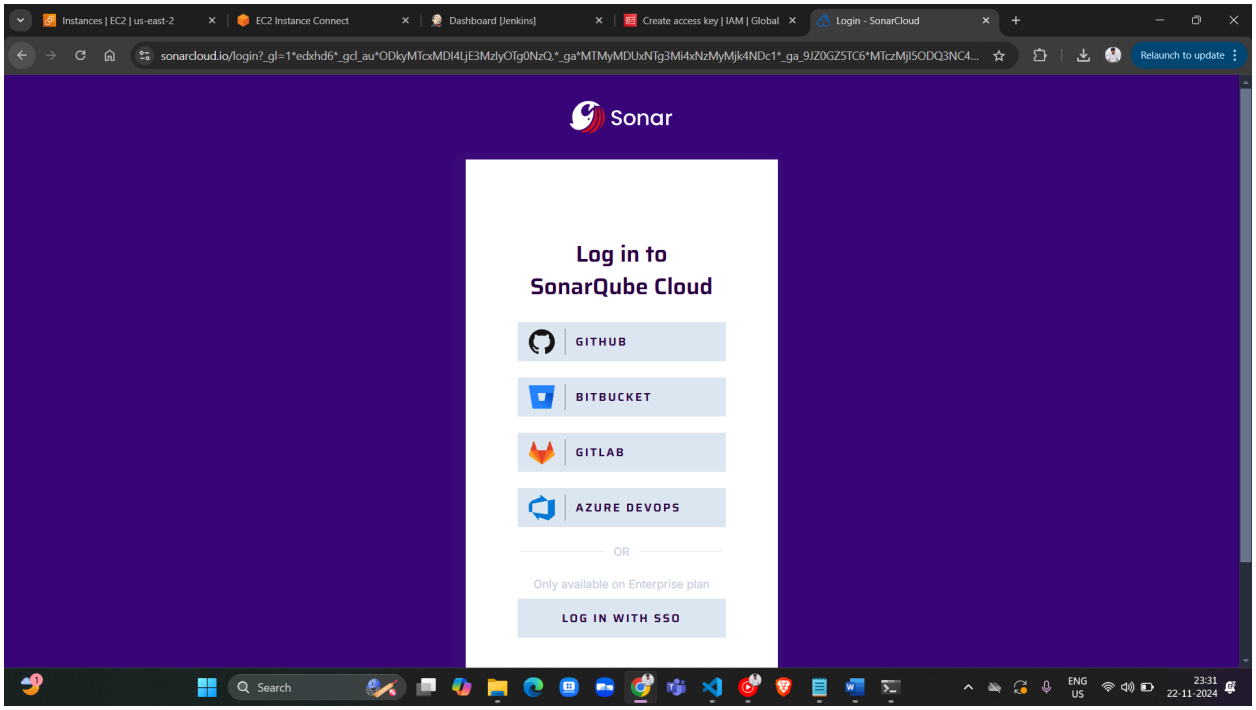


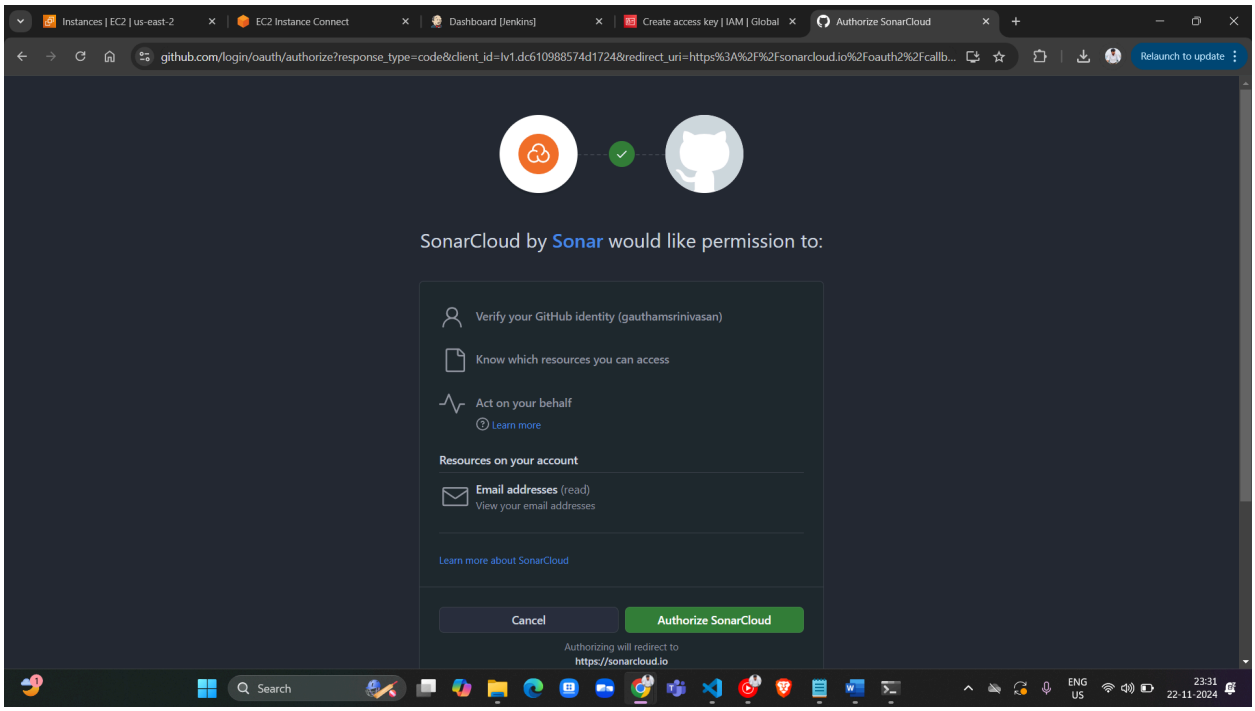
Step 5. SAST with SonarCloud

Configure SonarCloud

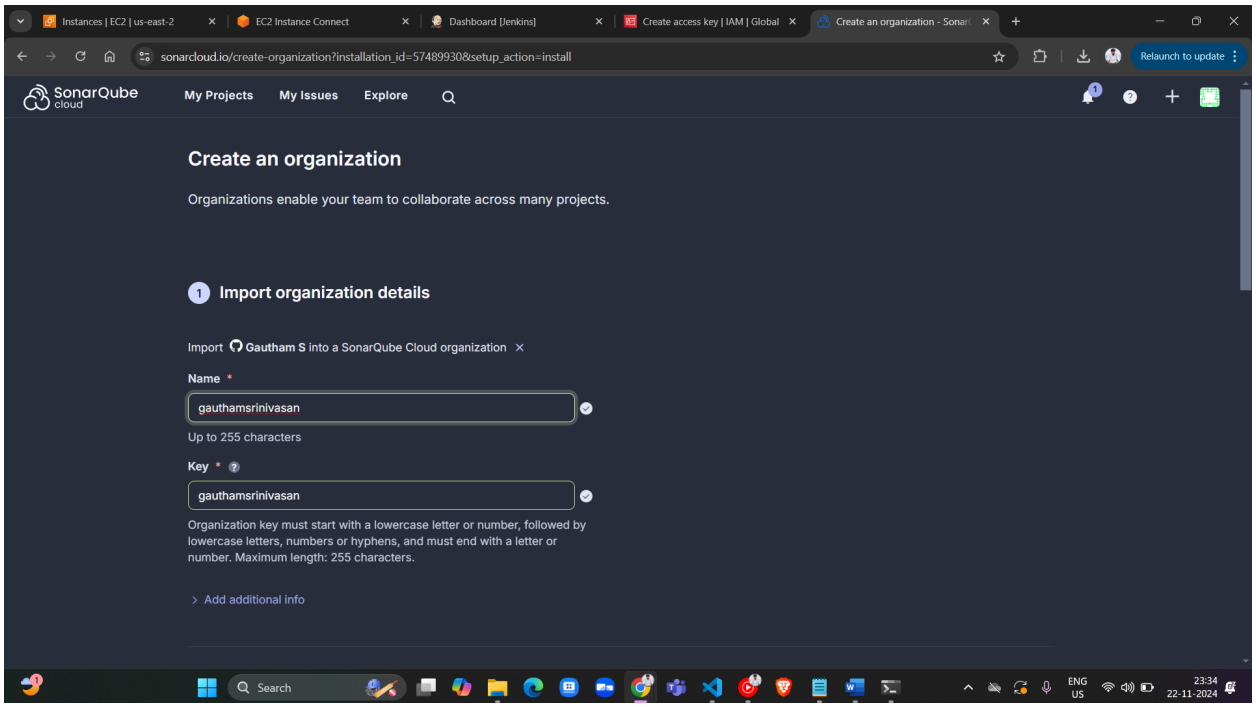
1. Sign in to SonarCloud using GitHub.

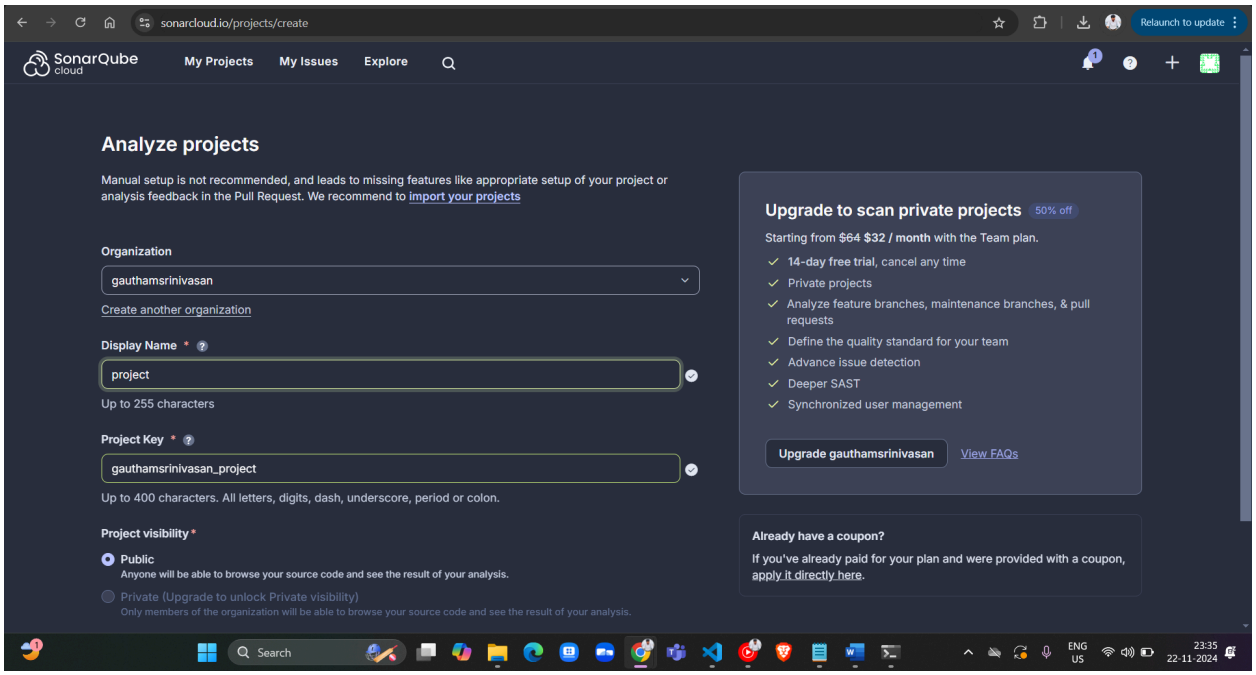

Instances | EC2 | us-east-2
EC2 Instance Connect
Dashboard [Jenkins]
Create access key | IAM | Global
SonarCloud Online Code Revie
sonarsource.com/products/sonarcloud/
Relaunch to update
NEW! Introducing Sonar AI Code Assurance and AI CodeFix
Sonar
Solutions
Products
Resources
Company
SonarQube cloud
Features
What's new
Roadmap
Docs
Pricing
Try now
Log in
AS A SERVICE. SONARQUBE CLOUD
SaaS solution for Clean Code. Simple, Scalable, Fast.
Enable your team to deliver clean code consistently and efficiently with a code review tool that easily integrates into the cloud DevOps platforms and extend your CI/CD workflow.
Start now
Overview
sonarcloud
is now
SonarQube cloud
Reliability
2 Open issues
D
1 H
1 M
0 L
Search
ENG US
23:31
22-11-2024

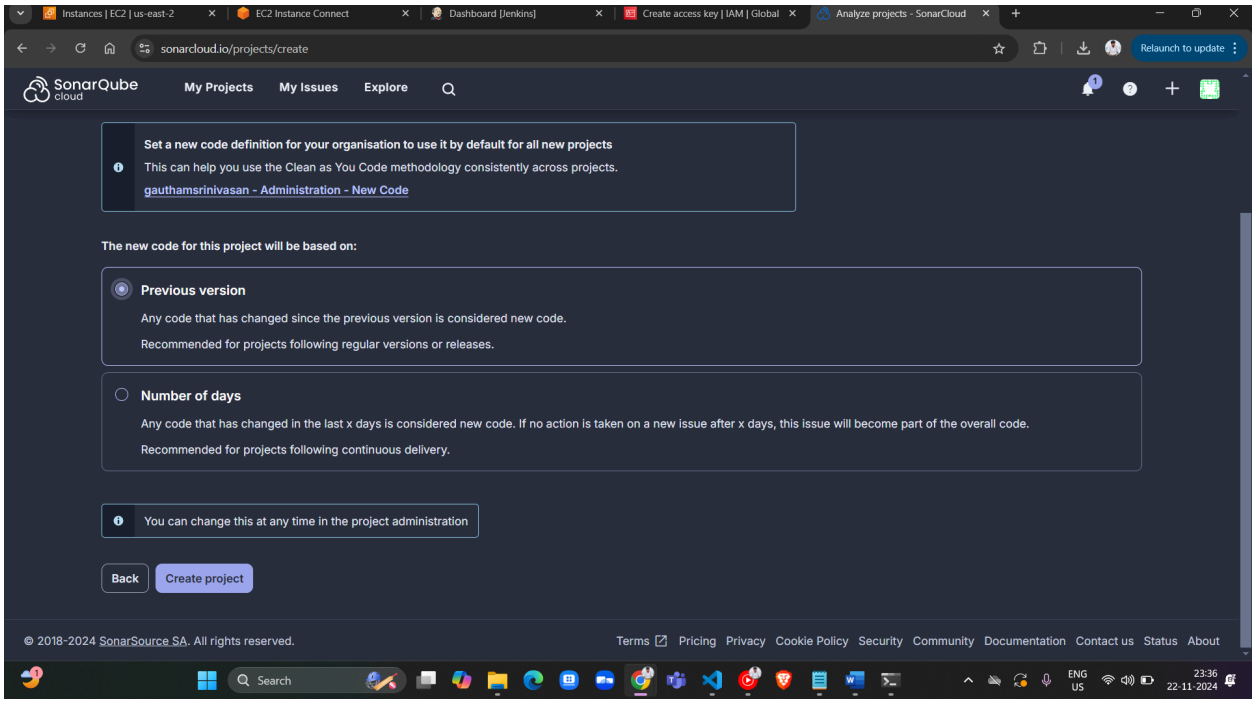

Instances | EC2 | us-east-2
EC2 Instance Connect
Dashboard [Jenkins]
Create access key | IAM | Global
Login - SonarCloud
sonarcloud.io/login?_gl=1*edxhd6*_gcl_au*ODkyMTcxMDI4LjE3MzIyOTg0NzQ.*_ga*MTMyMDUxNTg3Mi4xNzMyMjk4NDc1*_ga_9JZ0GZ5TC6*MTczMjI5ODQ3NC4...
Relaunch to update
Sonar
Log in to SonarQube Cloud
GITHUB
BITBUCKET
GITLAB
AZURE DEVOPS
OR
Only available on Enterprise plan
LOG IN WITH SSO
Search
ENG US
23:31
22-11-2024



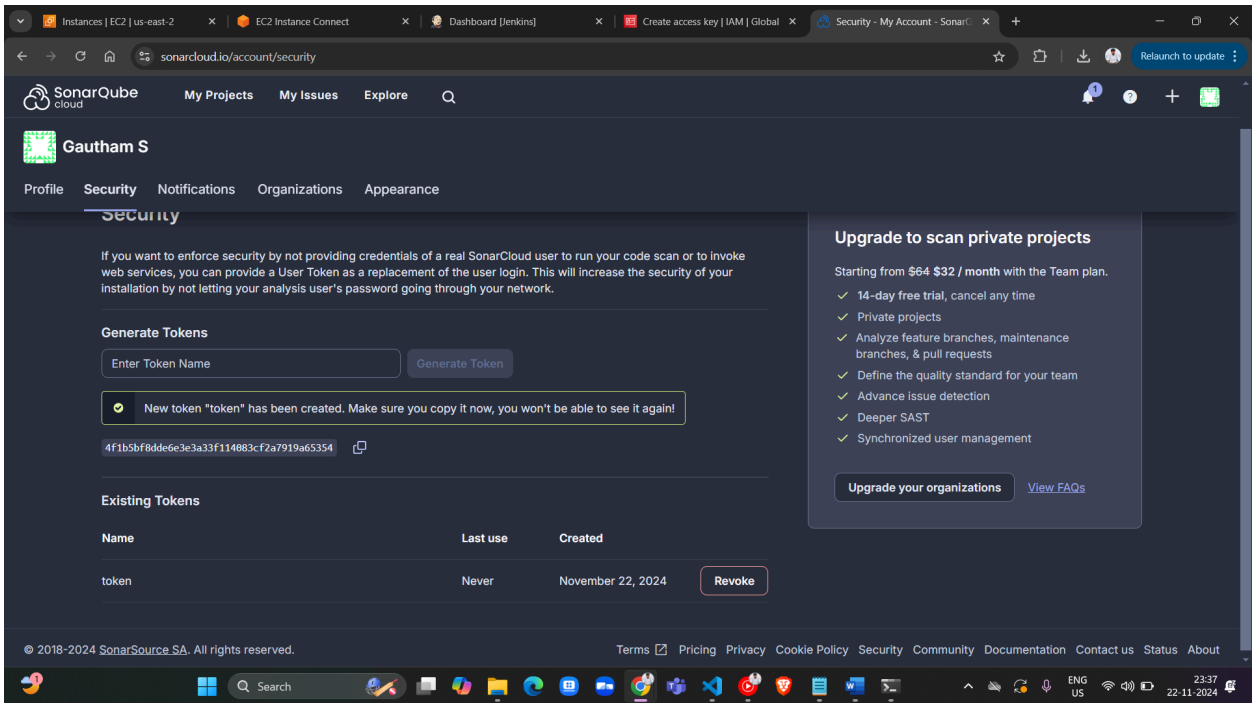
2. Create a new organization and project.


Instances | EC2 | us-east-2
EC2 Instance Connect
Dashboard [Jenkins]
Create access key | IAM | Global
Create an organization - Sonar
sonarcloud.io/create-organization?installation_id=57489930&setup_action=install
Relaunch to update
SonarQube cloud
My Projects
My Issues
Explore
Create an organization
Organizations enable your team to collaborate across many projects.
1 Import organization details
Import Gautham S into a SonarQube Cloud organization
Name *
gauthamsrinivasan
Up to 255 characters
Key *
gauthamsrinivasan
Organization key must start with a lowercase letter or number, followed by lowercase letters, numbers or hyphens, and must end with a letter or number. Maximum length: 255 characters.
Add additional info
Search
ENG US
23:34
22-11-2024


sonarcloud.io/projects/create
Relaunch to update
SonarQube cloud
My Projects
My Issues
Explore
Analyze projects
Manual setup is not recommended, and leads to missing features like appropriate setup of your project or analysis feedback in the Pull Request. We recommend to import your projects
Organization
gauthamsrinivasan
Create another organization
Display Name *
project
Up to 255 characters
Project Key *
gauthamsrinivasan_project
Up to 400 characters. All letters, digits, dash, underscore, period or colon.
Project visibility *
Public
Anyone will be able to browse your source code and see the result of your analysis.
Private (Upgrade to unlock Private visibility)
Only members of the organization will be able to browse your source code and see the result of your analysis.
Upgrade to scan private projects
50% off
Starting from $64 $32 / month with the Team plan.
14-day free trial, cancel any time
Private projects
Analyze feature branches, maintenance branches, & pull requests
Define the quality standard for your team
Advance issue detection
Deeper SAST
Synchronized user management
Upgrade gauthamsrinivasan
View FAQs
Already have a coupon?
If you've already paid for your plan and were provided with a coupon, apply it directly here.
Search
ENG US
23:35
22-11-2024

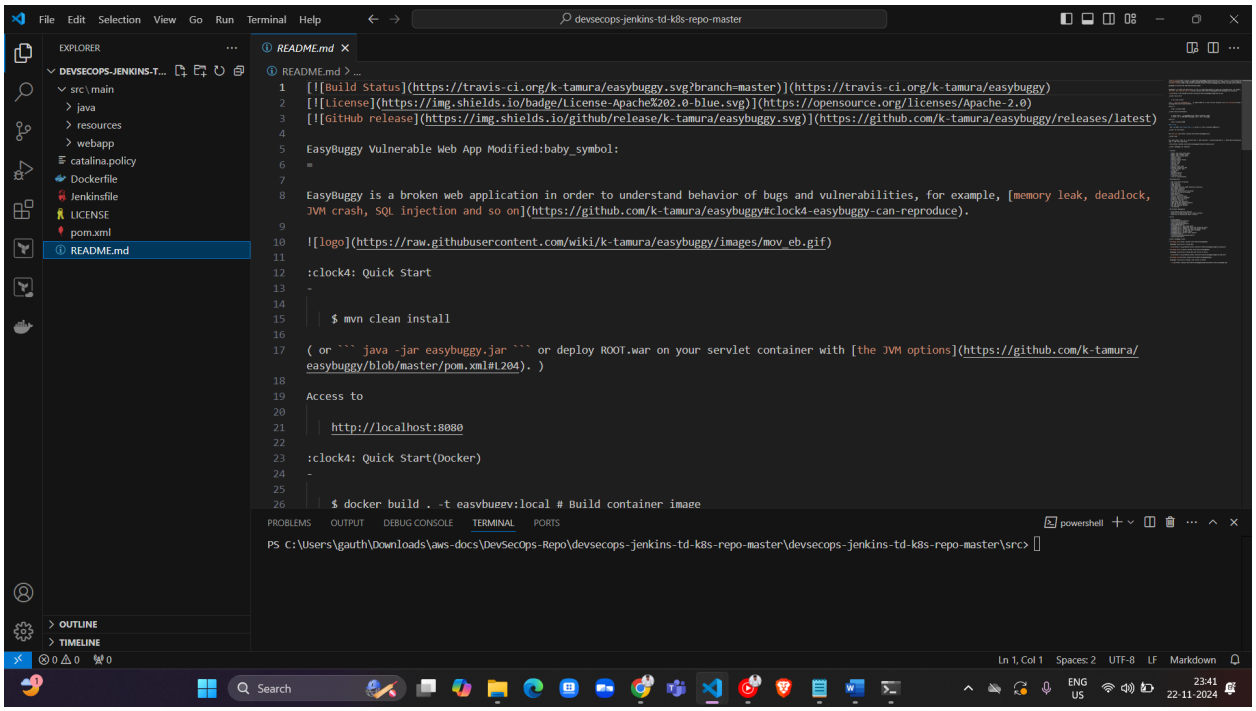


3. Generate a token from the "My Account > Security" tab.

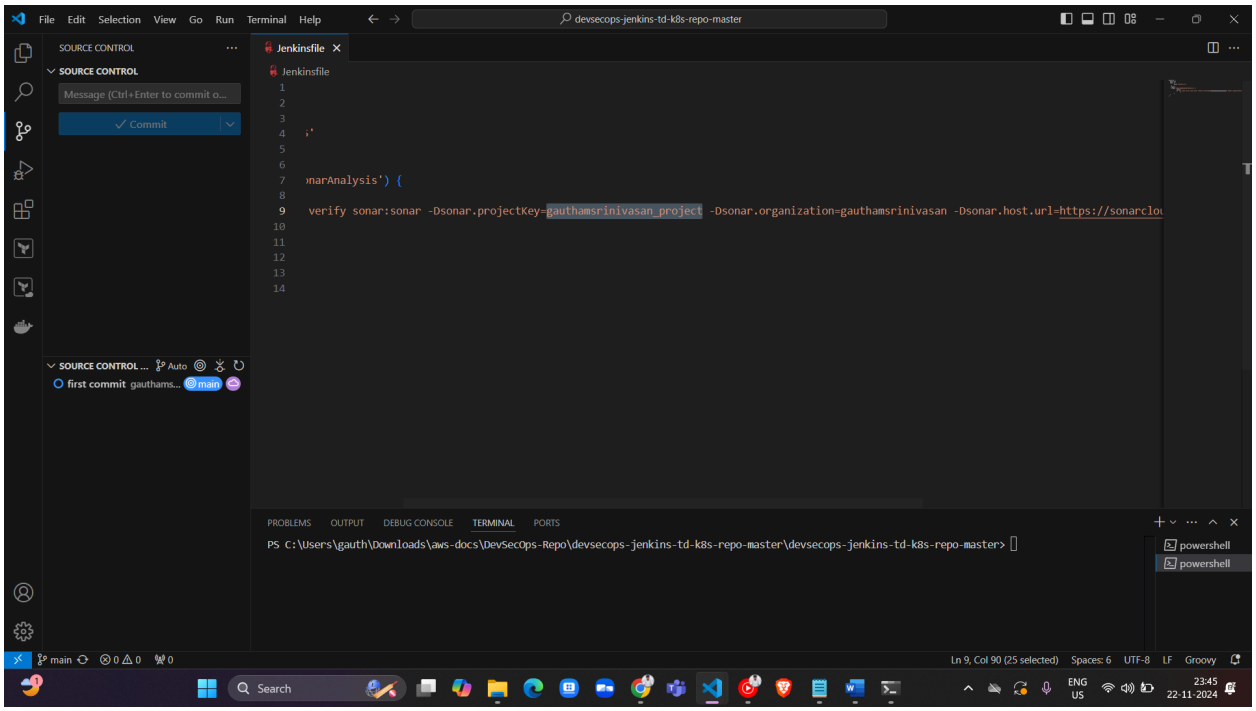


Update Jenkins Pipeline

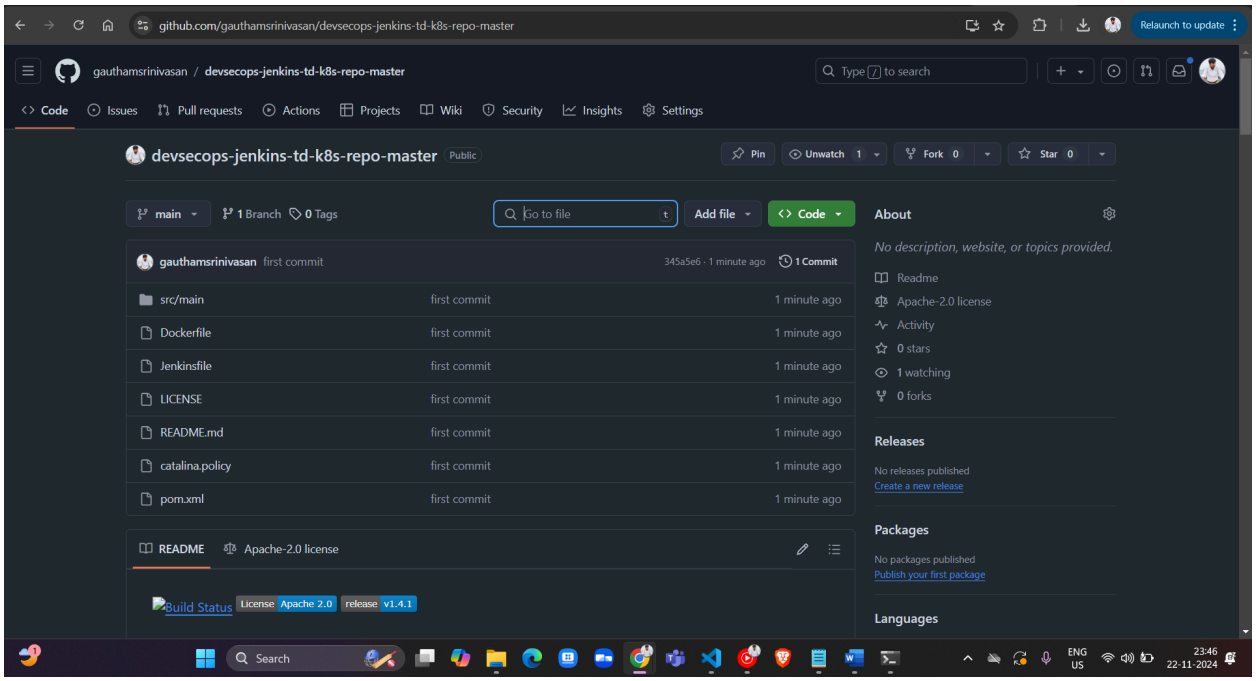
1. Clone the repository:
   `git clone https://github.com/Akshiv20/devsecops-jenkins-td-k8s-repo.git`



2. Update the following values in the `Jenkinsfile`:
   - Sonar project key
   - Organization key
   - Token

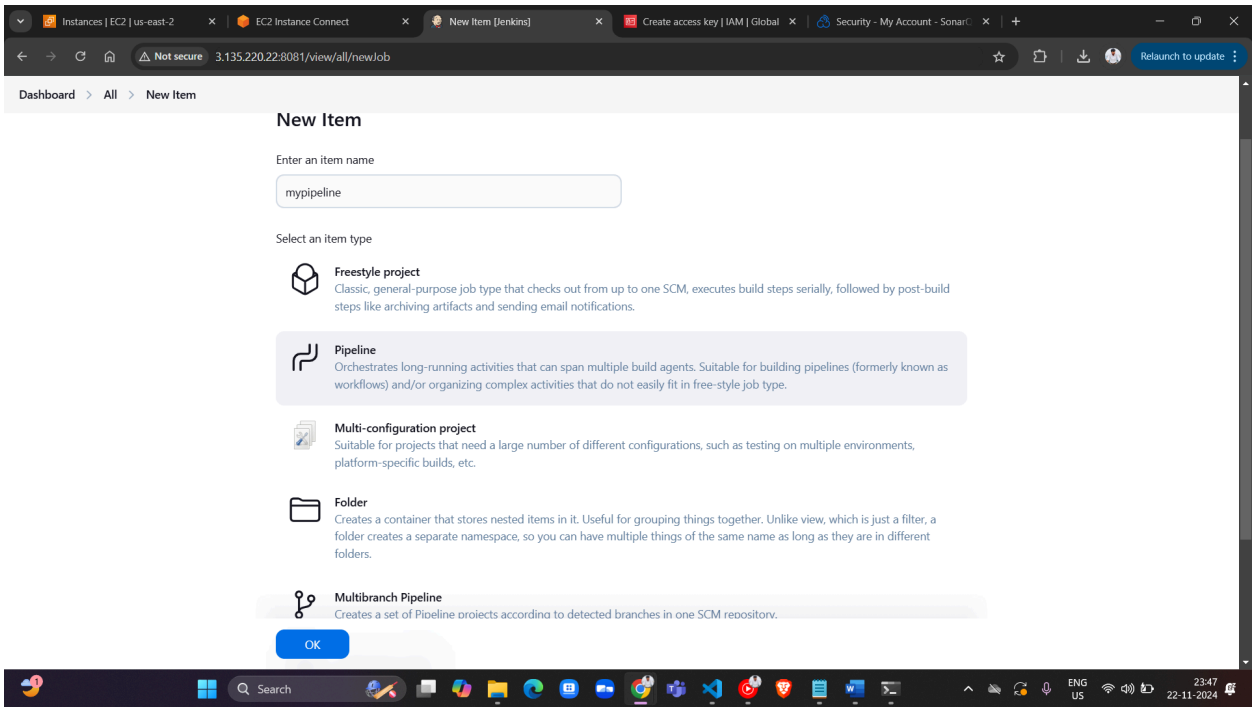


Save and publish the repo to your github account and verify

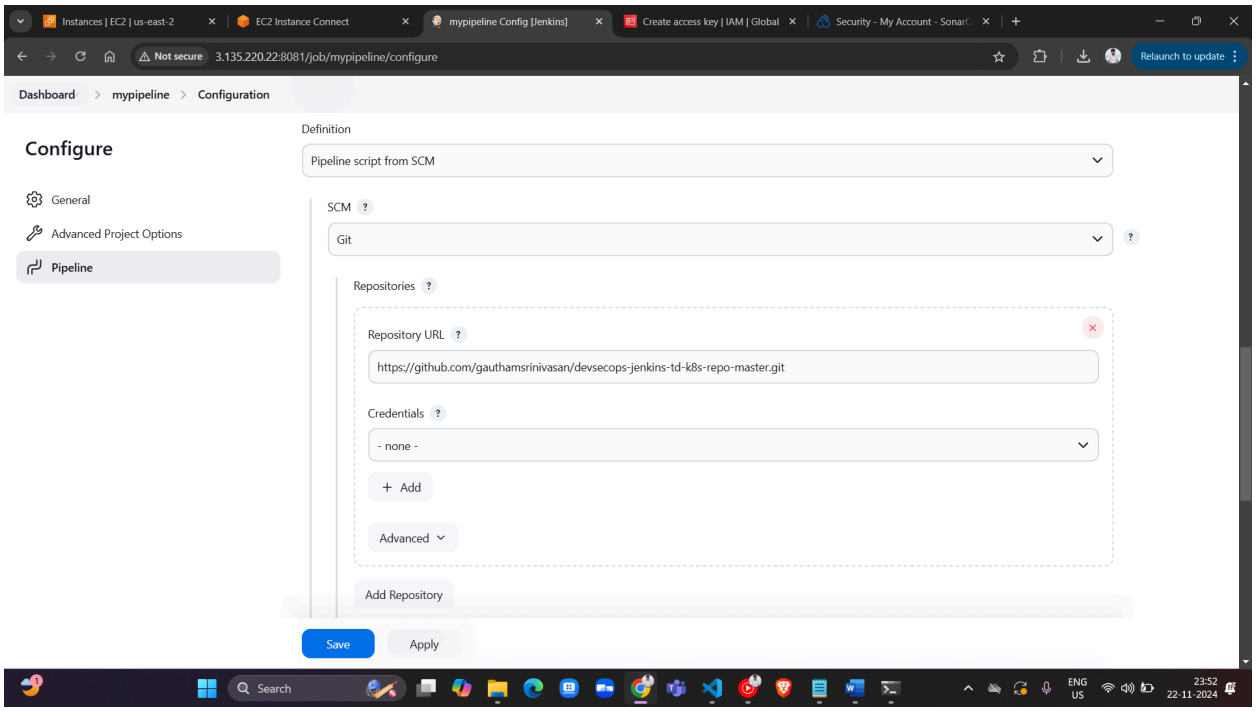


Run SAST Scan

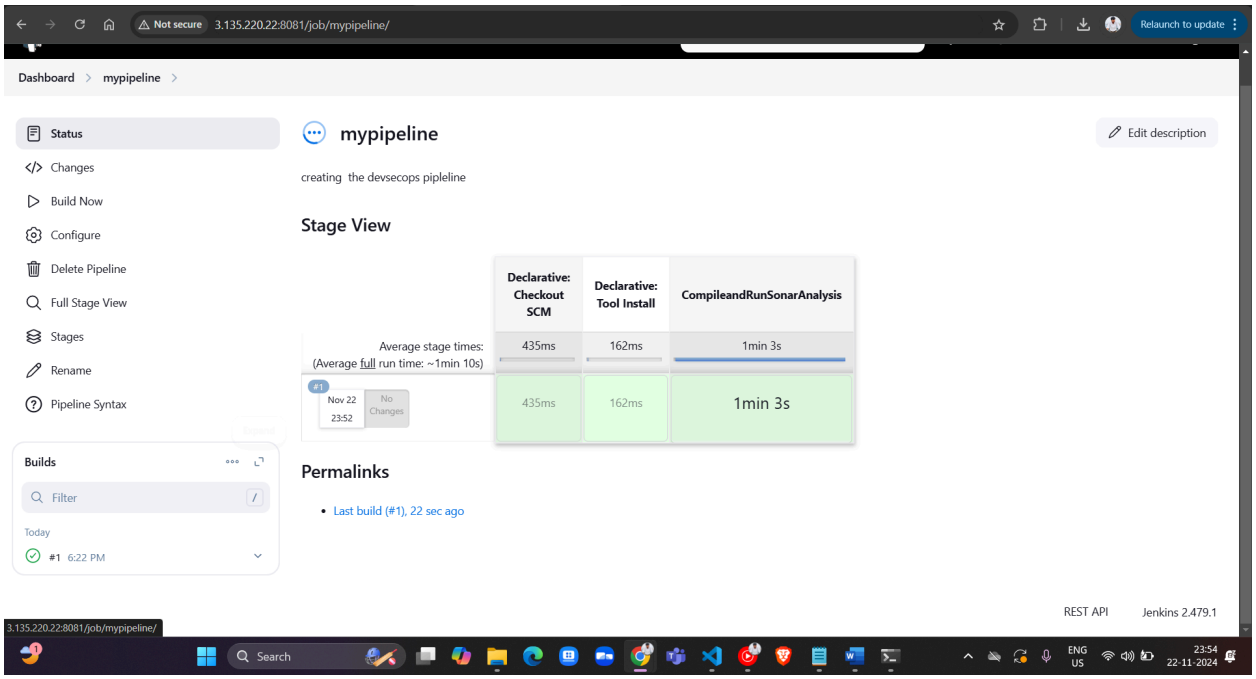
1. Create a new Jenkins pipeline.

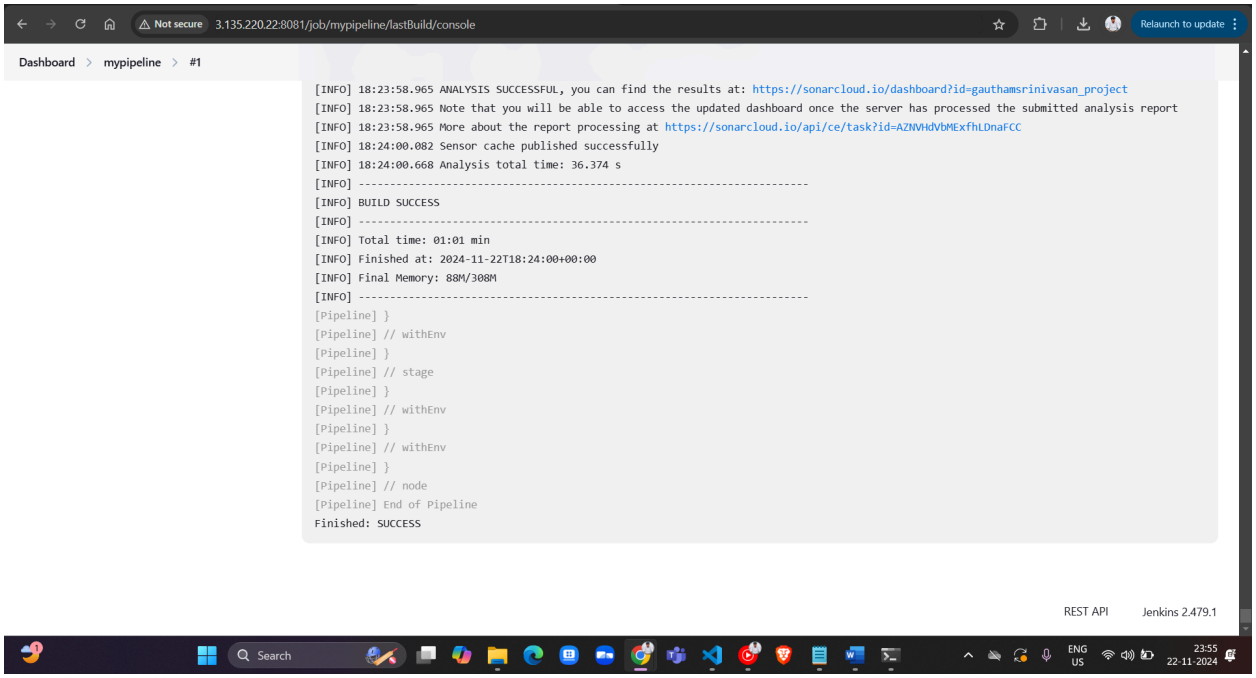


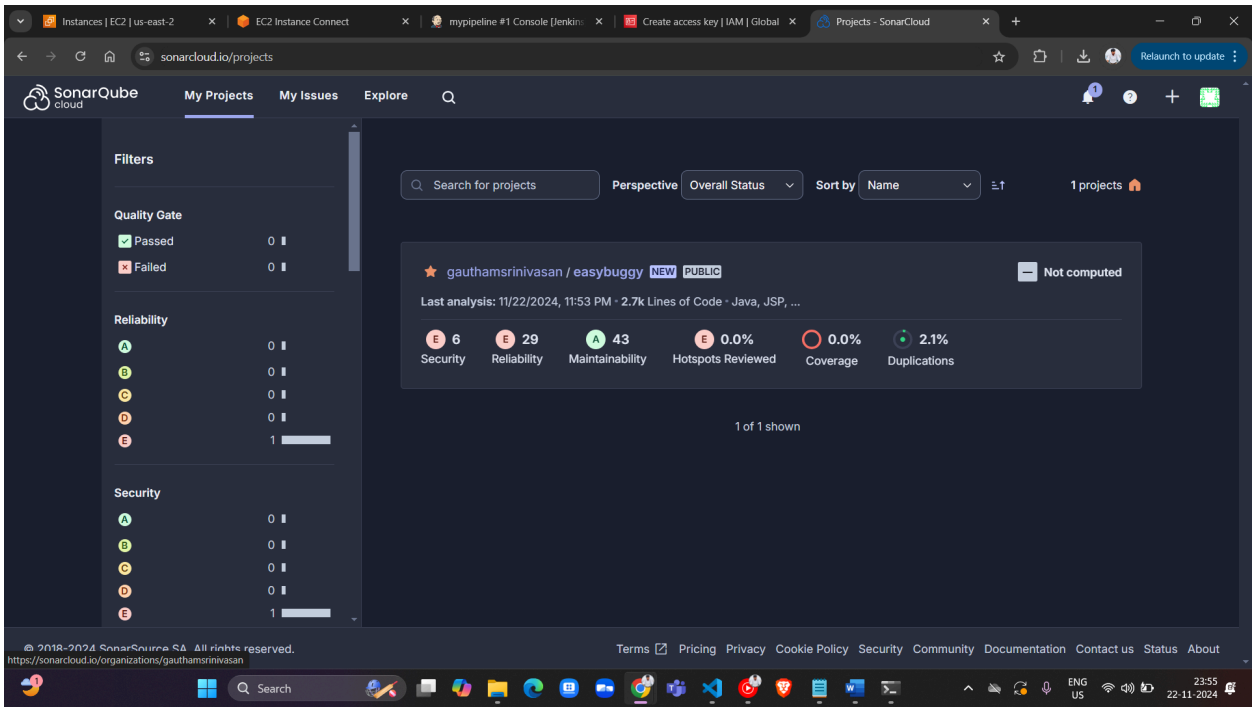
2. Configure the pipeline to use the `devsecops-jenkins-td-k8s-repo` repository.



3. Trigger a build and verify the results on SonarCloud.


Not secure 3.135.220.22:8081/job/mypipeline/
Relaunch to update
Dashboard > mypipeline >
Status
Changes
Build Now
Configure
Delete Pipeline
Full Stage View
Stages
Rename
Pipeline Syntax
Builds
Filter
Today
#1 6:22 PM
mypipeline
Edit description
creating the devsecops pipleline
Stage View
Declarative: Checkout SCM
Declarative: Tool Install
CompileandRunSonarAnalysis
Average stage times:
(Average full run time: ~1min 10s)
435ms
162ms
1min 3s
#1
Nov 22
23:52
No Changes
435ms
162ms
1min 3s
Permalinks
Last build (#1), 22 sec ago
REST API
Jenkins 2.479.1
3.135.220.22:8081/job/mypipeline/
Search
ENG
US
23:54
22-11-2024

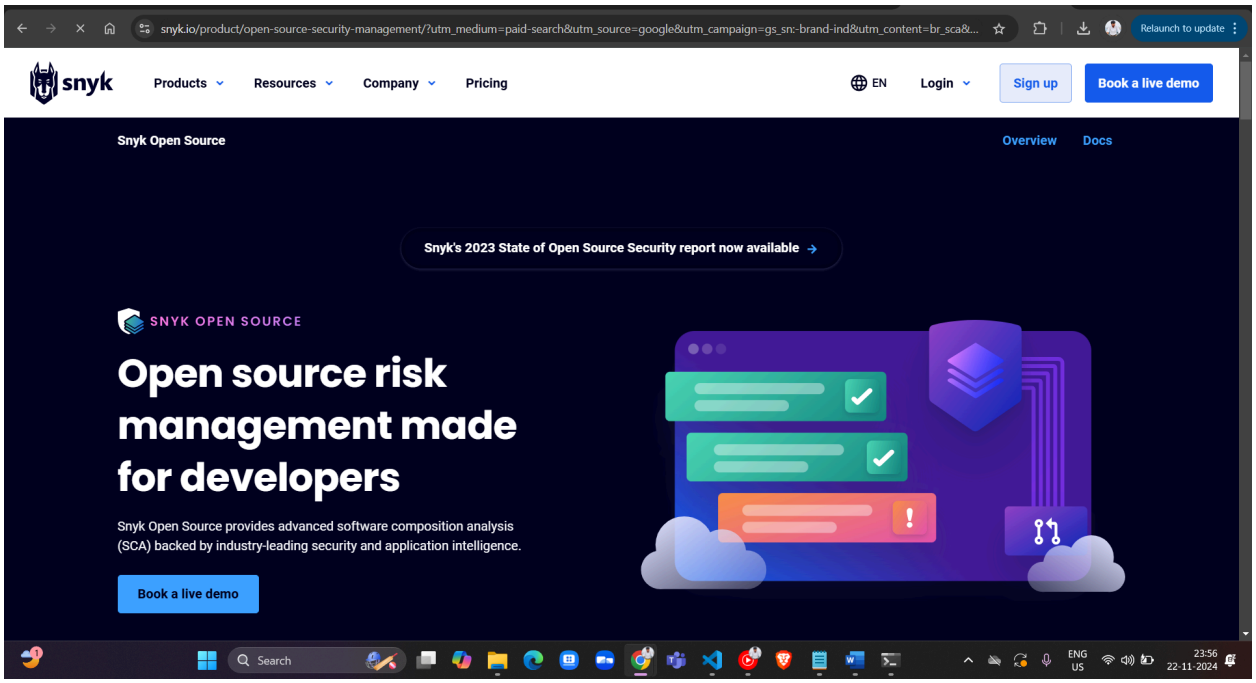

Not secure 3.135.220.22:8081/job/mypipeline/lastBuild/console
Relaunch to update
Dashboard > mypipeline > #1
[INFO] 18:23:58.965 ANALYSIS SUCCESSFUL, you can find the results at: https://sonarcloud.io/dashboard?id=gauthamsrinivasan_project
[INFO] 18:23:58.965 Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
[INFO] 18:23:58.965 More about the report processing at https://sonarcloud.io/api/ce/task?id=AZNVHdVbMExfhLDnaFCC
[INFO] 18:24:00.082 Sensor cache published successfully
[INFO] 18:24:00.668 Analysis total time: 36.374 s
[INFO] --
[INFO] BUILD SUCCESS
[INFO] --
[INFO] Total time: 01:01 min
[INFO] Finished at: 2024-11-22T18:24:00+00:00
[INFO] Final Memory: 88M/308M
[INFO] --
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
REST API
Jenkins 2.479.1
Search
ENG
US
23:55
22-11-2024

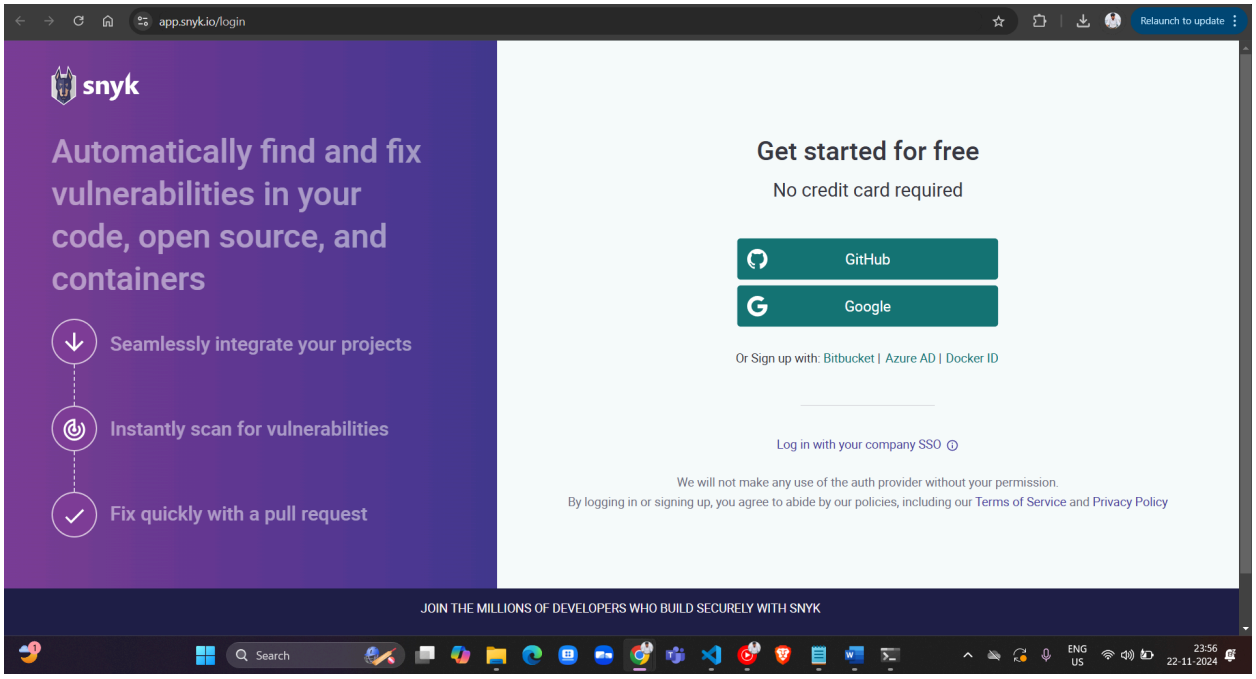


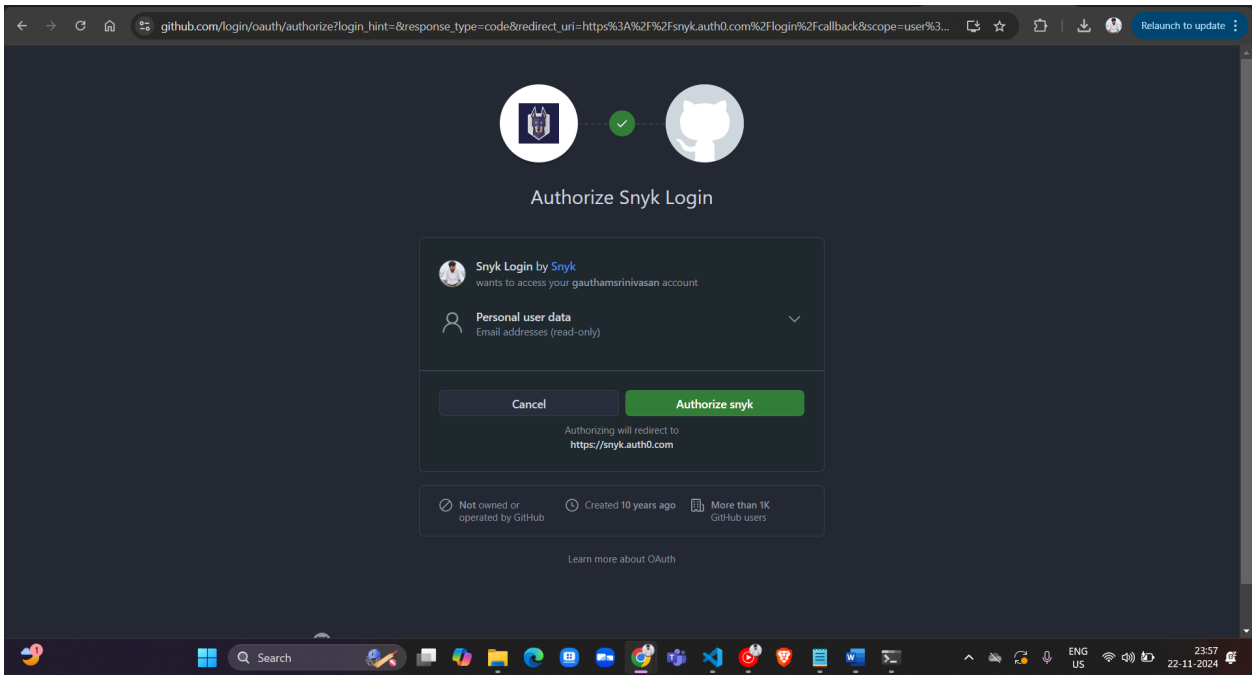
Step 6. SCA with Snyk

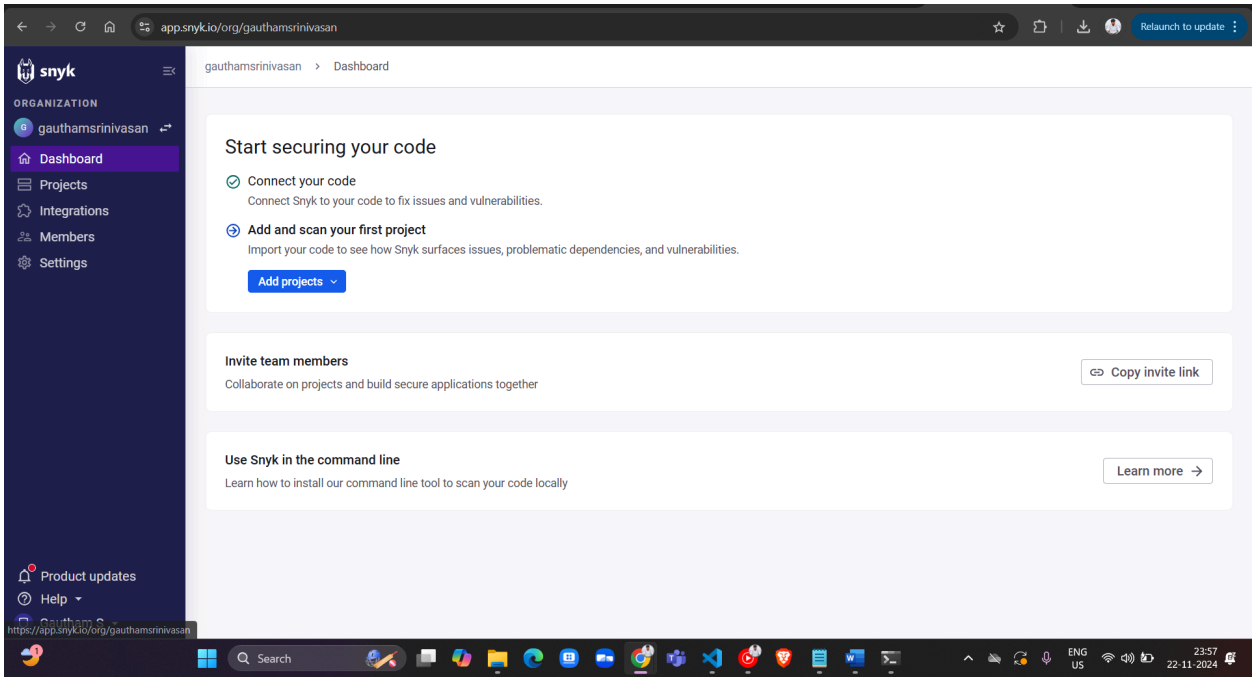
1. Configure Snyk

1. Sign up for Snyk using GitHub.

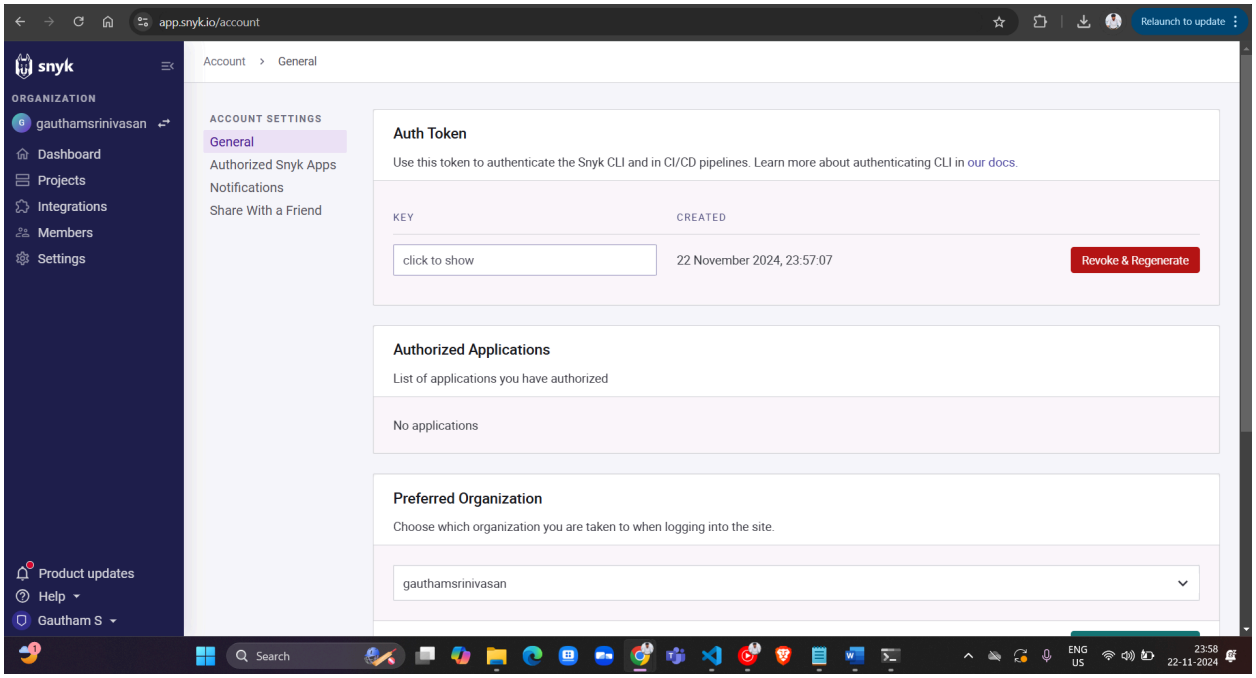

snyk.io/product/open-source-security-management/?utm_medium=paid-search&utm_source=google&utm_campaign=gs_sn:-brand-ind&utm_content=br_sca&...
Relaunch to update
snyk
Products
Resources
Company
Pricing
EN
Login
Sign up
Book a live demo
Snyk Open Source
Overview
Docs
Snyk's 2023 State of Open Source Security report now available
SNYK OPEN SOURCE
Open source risk management made for developers
Snyk Open Source provides advanced software composition analysis (SCA) backed by industry-leading security and application intelligence.
Book a live demo
Search
ENG US
23:56 22-11-2024

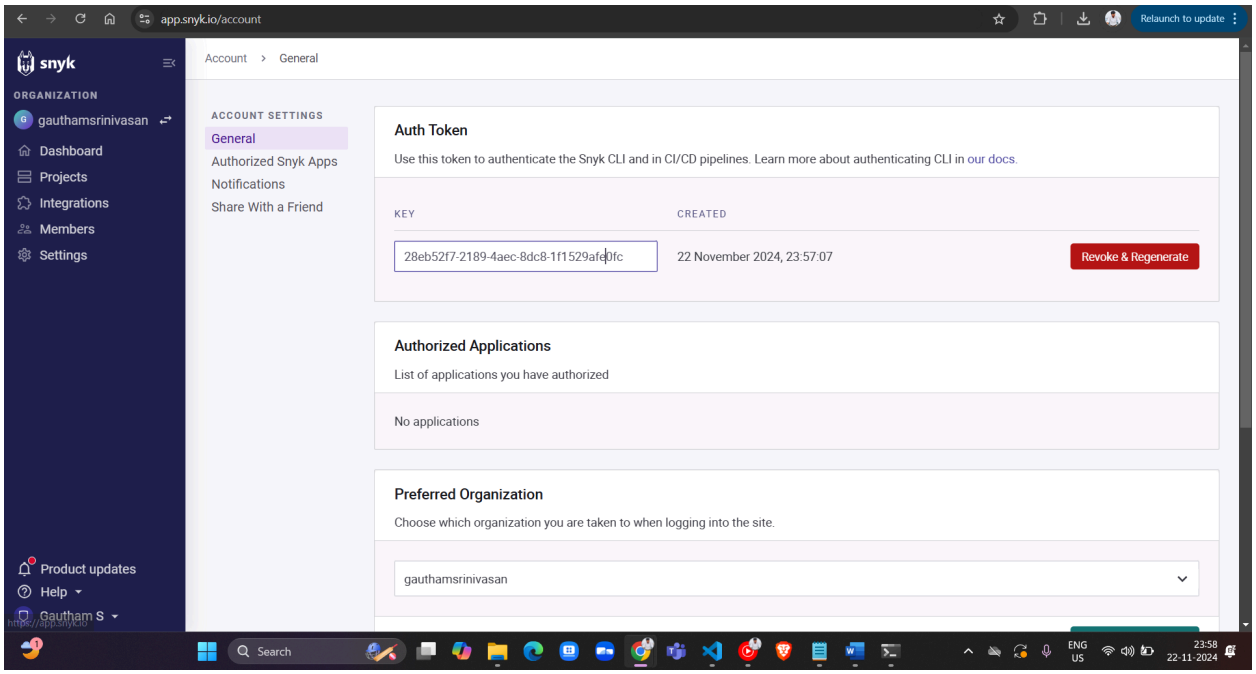

app.snyk.io/login
Relaunch to update
snyk
Automatically find and fix vulnerabilities in your code, open source, and containers
Seamlessly integrate your projects
Instantly scan for vulnerabilities
Fix quickly with a pull request
Get started for free
No credit card required
GitHub
Google
Or Sign up with: Bitbucket | Azure AD | Docker ID
Log in with your company SSO
We will not make any use of the auth provider without your permission.
By logging in or signing up, you agree to abide by our policies, including our Terms of Service and Privacy Policy
JOIN THE MILLIONS OF DEVELOPERS WHO BUILD SECURELY WITH SNYK
Search
ENG US
23:56 22-11-2024





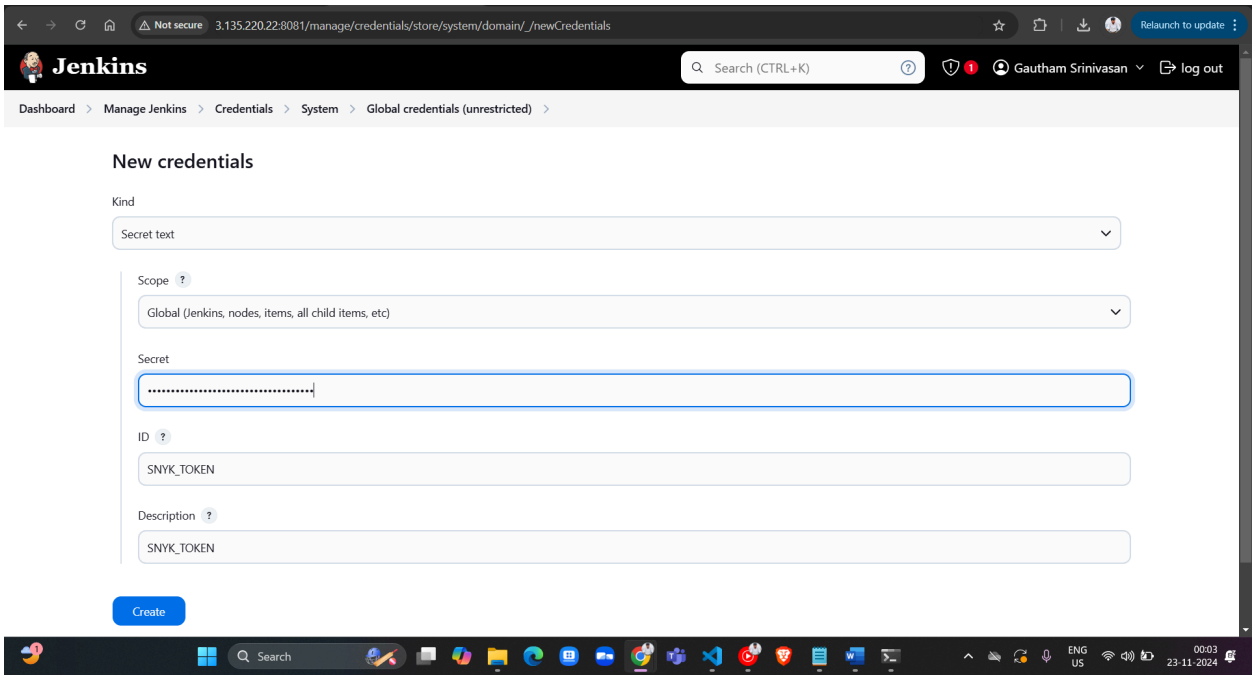
2. Generate an authentication token from "Account Settings > Auth Tokens".





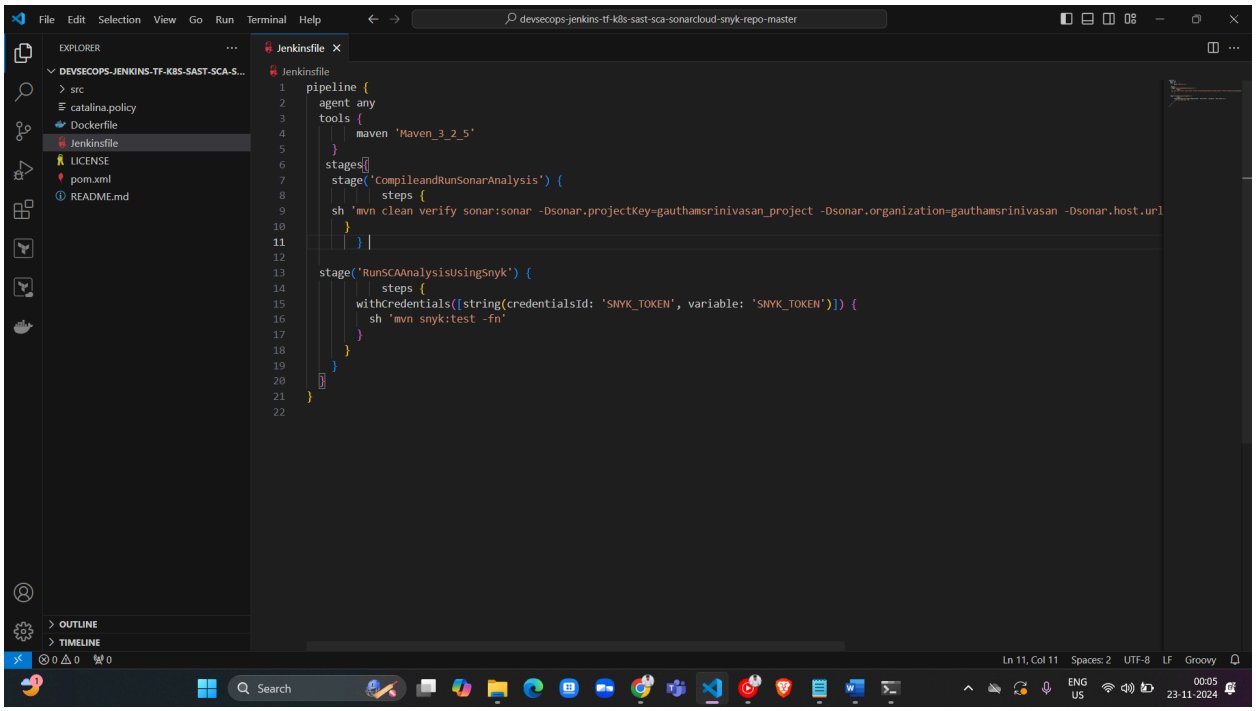
2. Add Credentials in Jenkins

1. Go to Jenkins > Manage Jenkins > Credentials.
2. Add a new secret text:
    - ID: `SNYK_TOKEN`
    - Secret: Paste the Snyk token.

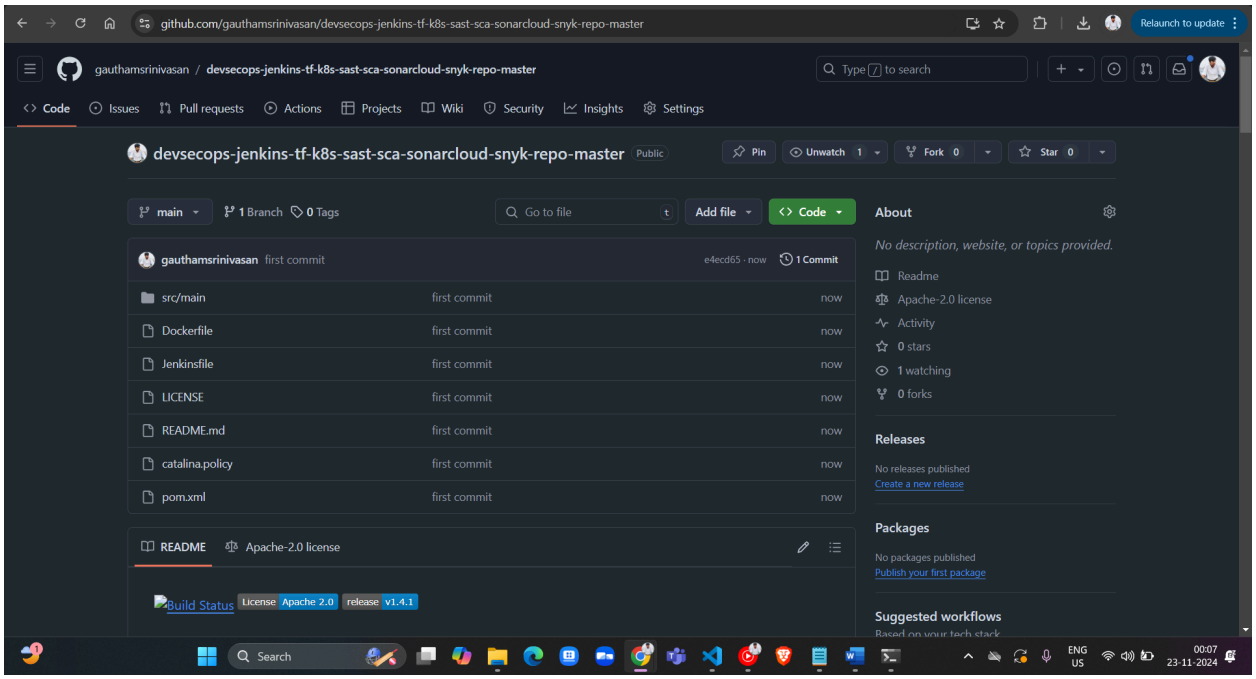


3. Update Jenkins Pipeline

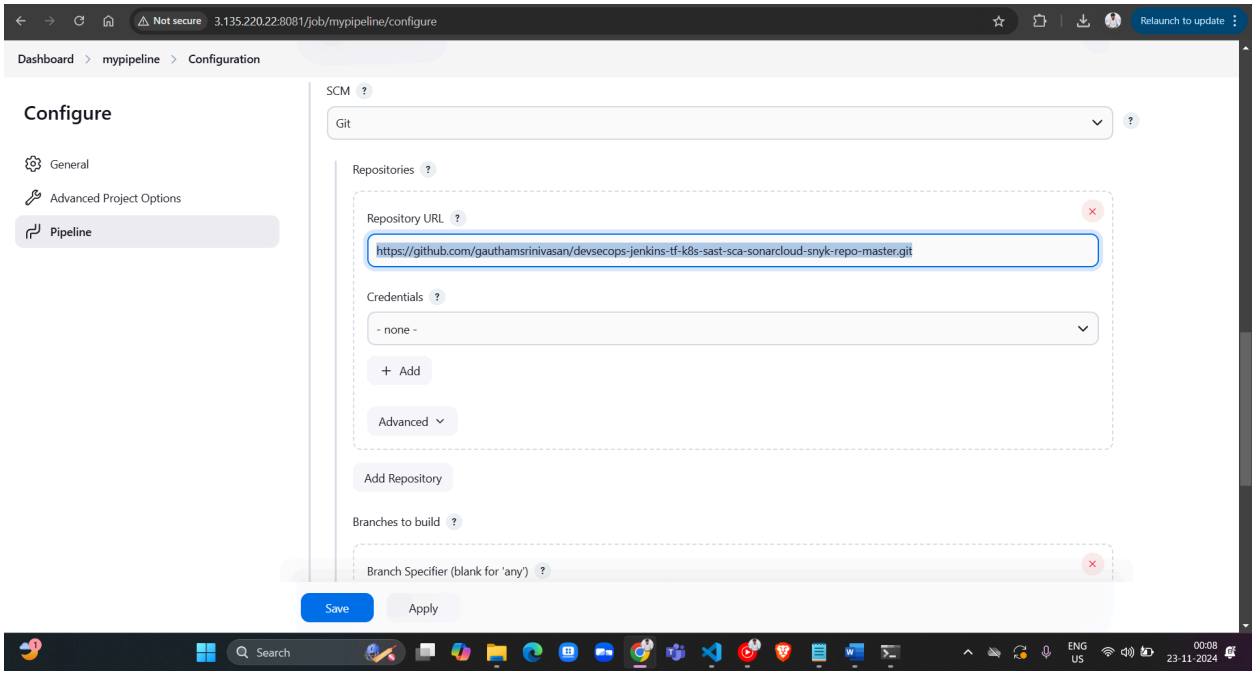
1. Clone the repository:
   `git clone https://github.com/Akshiv20/devsecops-jenkins-tf-k8s-sast-sca-sonarcloud-snyk-repo.git`
2. Update the `Jenkinsfile` with the Sonar token and Snyk token.

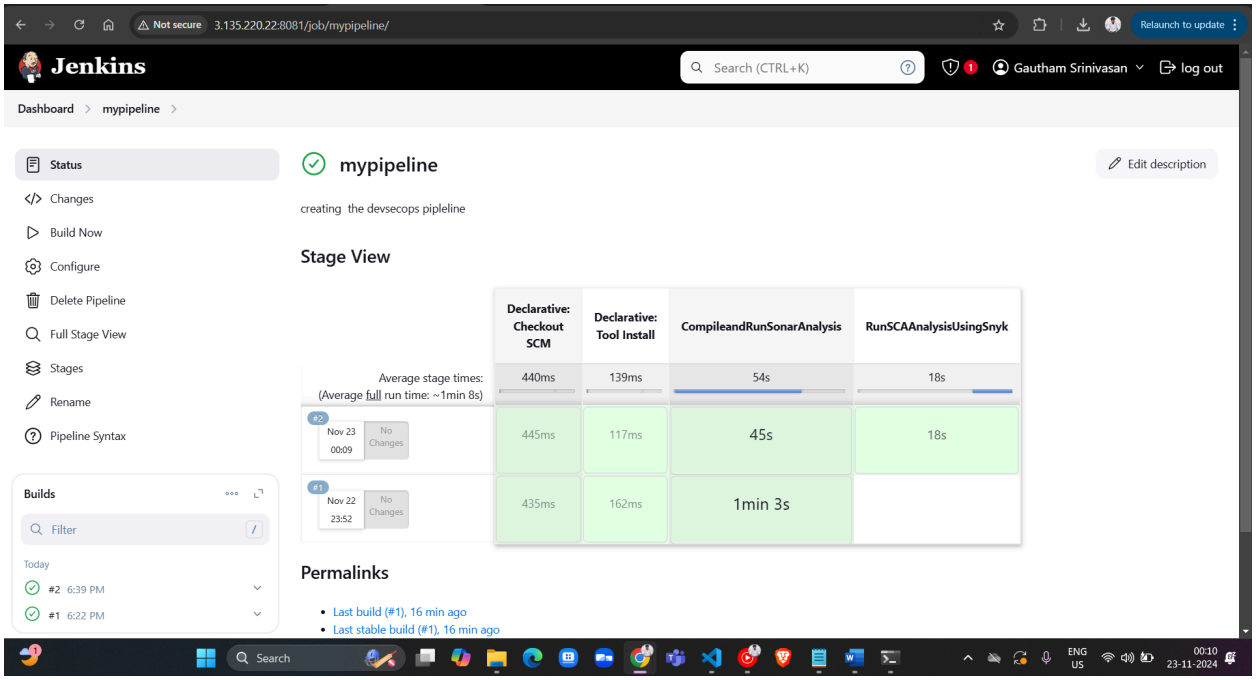


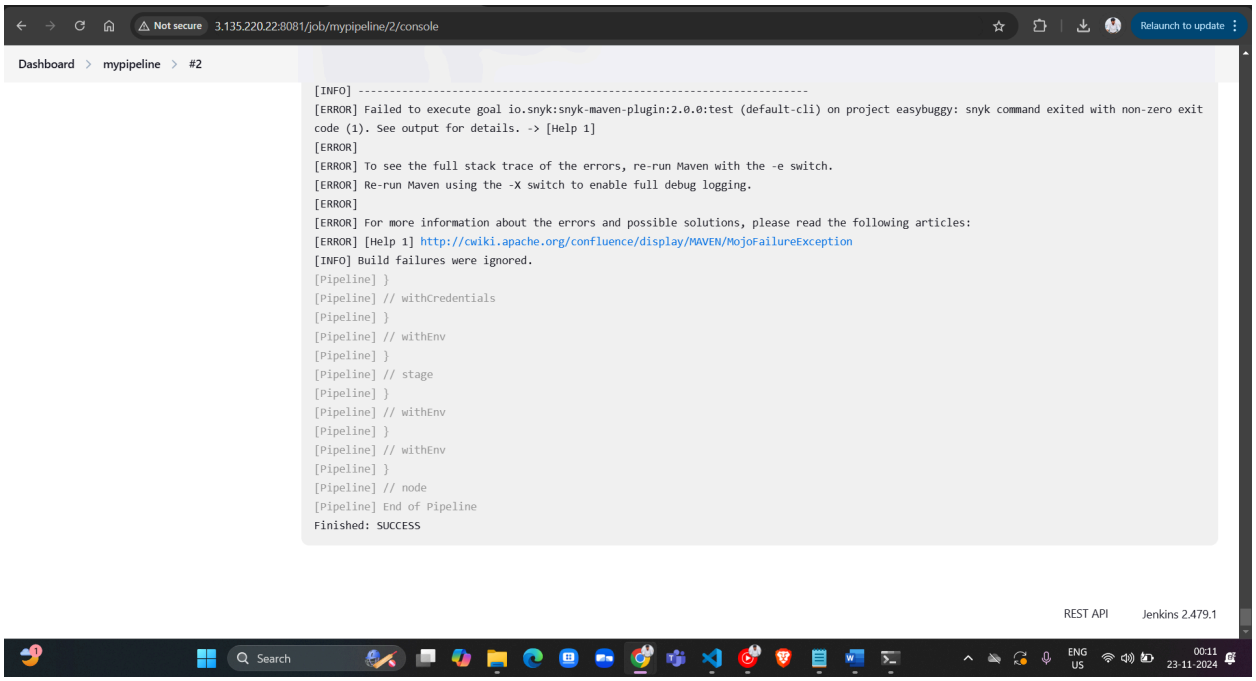
Save and publish the repo to your github account and verify



- Change the github url in the pipeline and build now and verify


Not secure 3.135.220.22:8081/job/mypipeline/configure
Relaunch to update
Dashboard > mypipeline > Configuration
Configure
General
Advanced Project Options
Pipeline
SCM
Git
Repositories
Repository URL
https://github.com/gauthamsrinivasan/devsecops-jenkins-tf-k8s-sast-sca-sonarcloud-snyk-repo-master.git
Credentials
- none -
+ Add
Advanced
Add Repository
Branches to build
Branch Specifier (blank for 'any')
Save
Apply
Search
ENG
US
00:08
23-11-2024

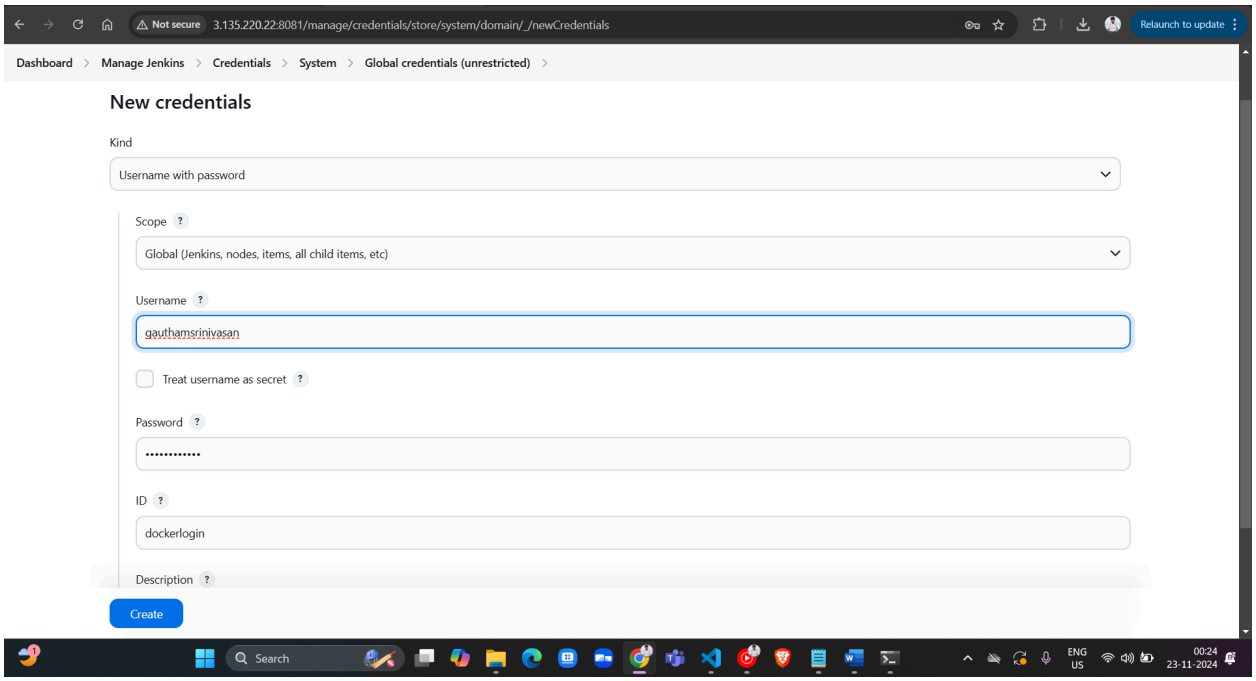

Not secure 3.135.220.22:8081/job/mypipeline/
Relaunch to update
Jenkins
Search (CTRL+K)
Gautham Srinivasan
log out
Dashboard > mypipeline >
Status
Changes
Build Now
Configure
Delete Pipeline
Full Stage View
Stages
Rename
Pipeline Syntax
Builds
Filter
Today
#2 6:39 PM
#1 6:22 PM
mypipeline
Edit description
creating the devsecops pipleline
Stage View
Declarative: Checkout SCM
Declarative: Tool Install
CompileandRunSonarAnalysis
RunSCAAnalysisUsingSnyk
Average stage times:
(Average full run time: ~1min 8s)
440ms
139ms
54s
18s
#2
Nov 23
00:09
No Changes
445ms
117ms
45s
18s
#1
Nov 22
23:52
No Changes
435ms
162ms
1min 3s
Permalinks
Last build (#1), 16 min ago
Last stable build (#1), 16 min ago
Search
ENG
US
00:10
23-11-2024



Step 7. Build and Push to ECR

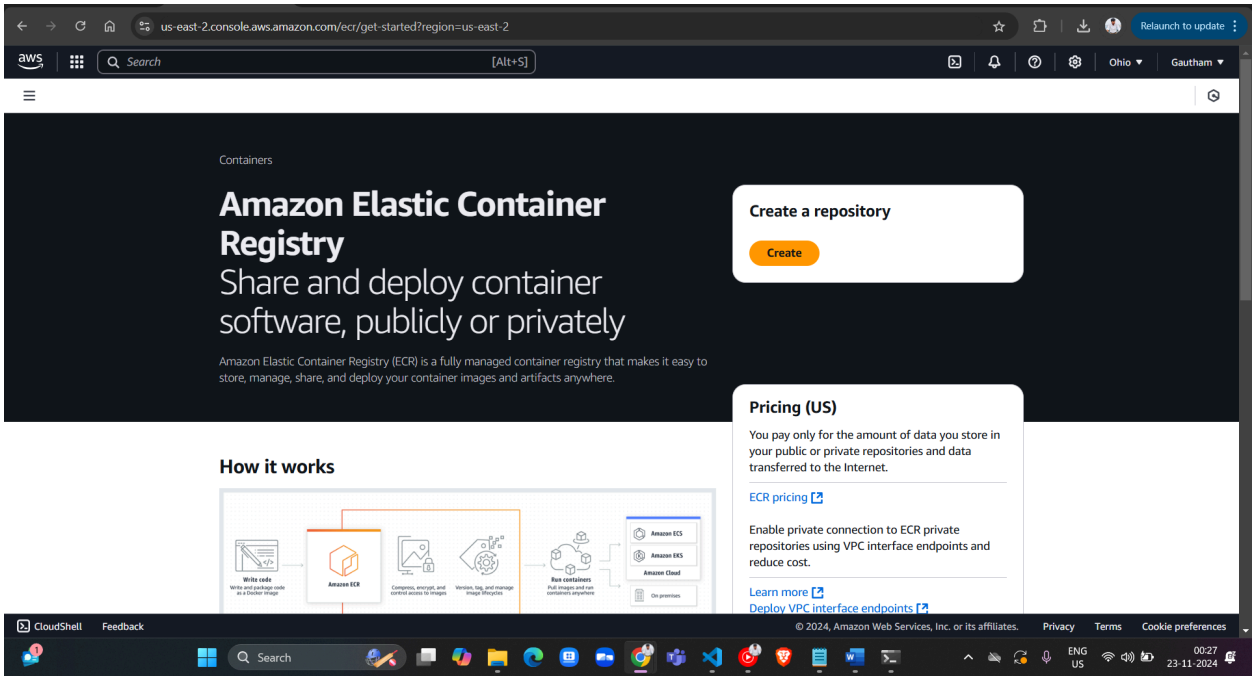
1. Configure Docker Credentials

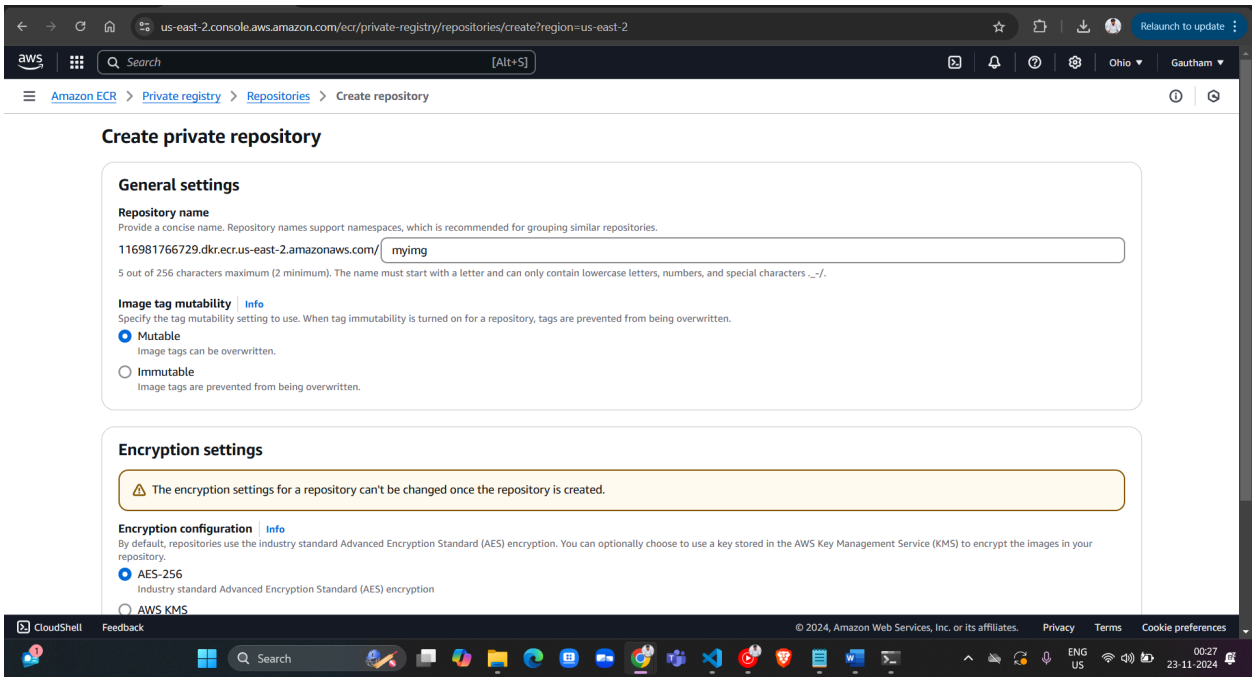
1. Add Docker Hub credentials in Jenkins:
    - ID: `dockerlogin`
    - Username and Password: Your Docker Hub credentials.



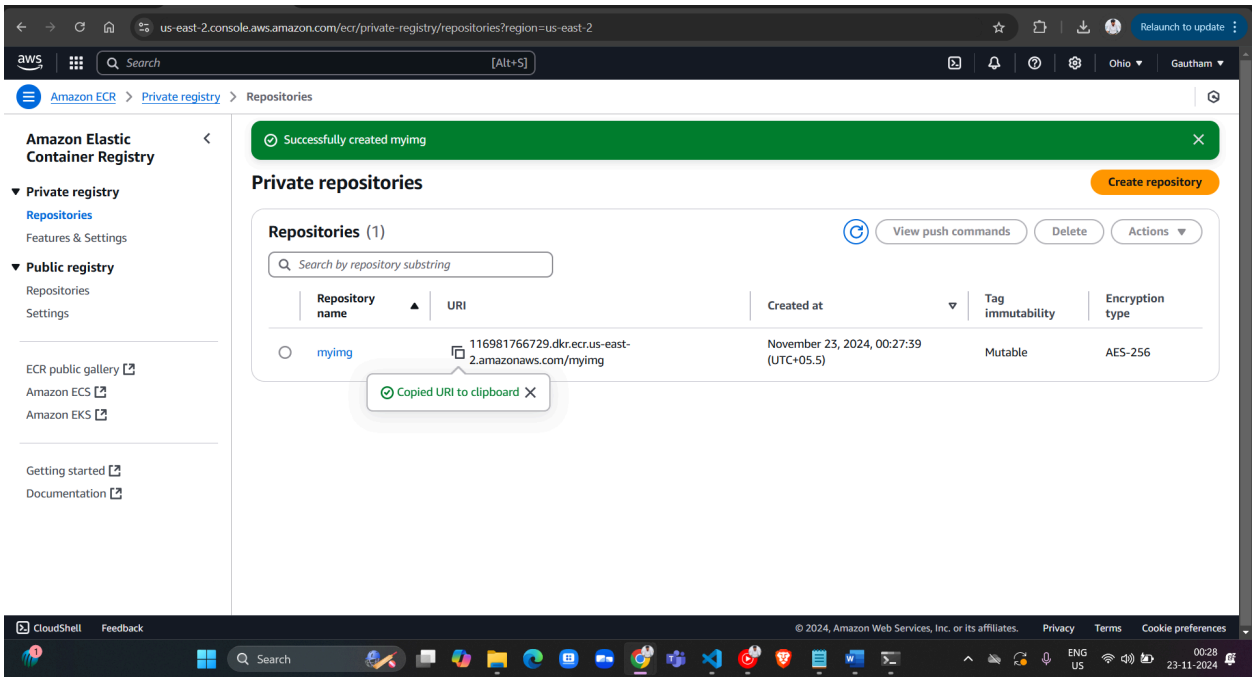
2.Create ECR

1. Go to amazon ecr service and create a new repository.



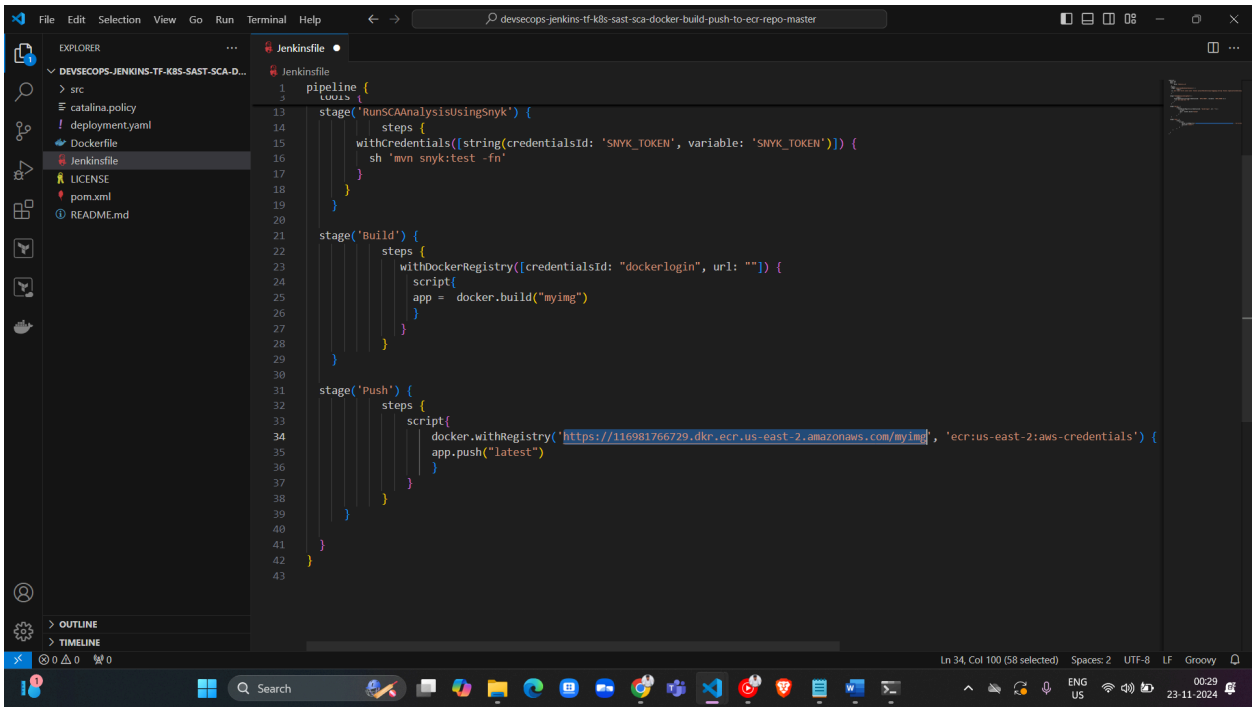


2. Copy the uri (“116981766729.dkr.ecr.us-east-2.amazonaws.com/myimg”)



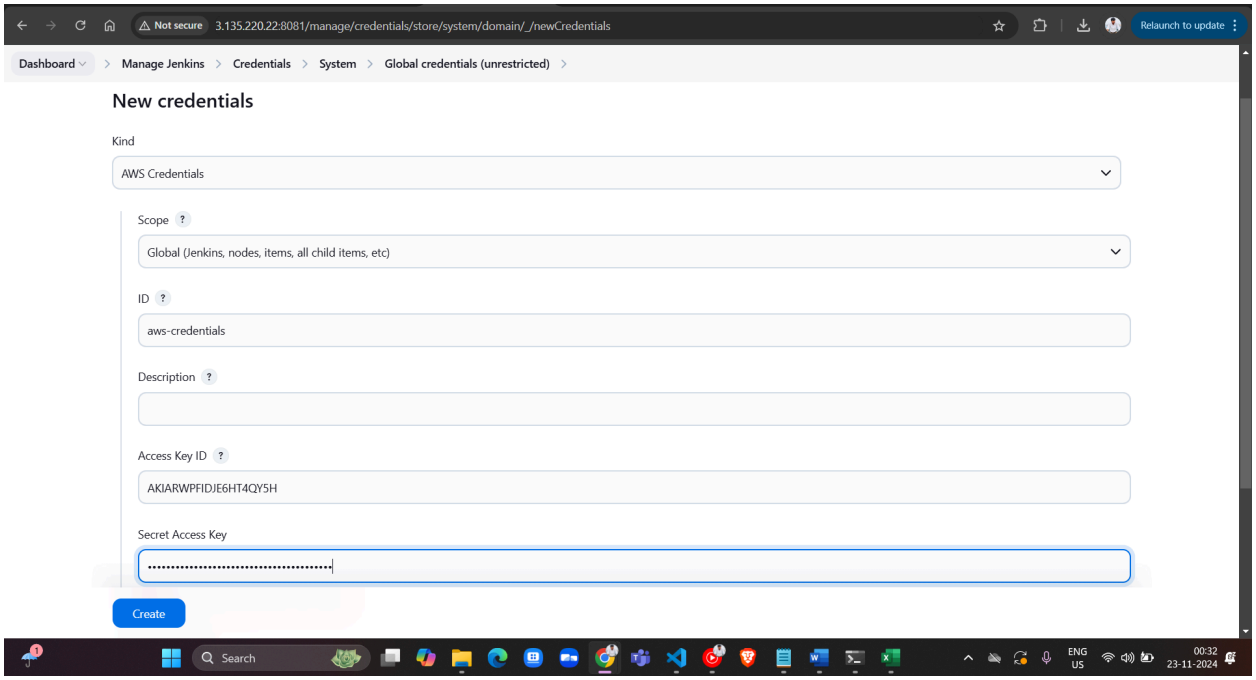
3. Update Jenkins Pipeline

1. Clone the repository:
   `git clone https://github.com/Akshiv20/devsecops-jenkins-tf-k8s-sast-sca-docker-build-push-to-ecr-repo.git`
2. Update the `Jenkinsfile` with the ECR repository URI.

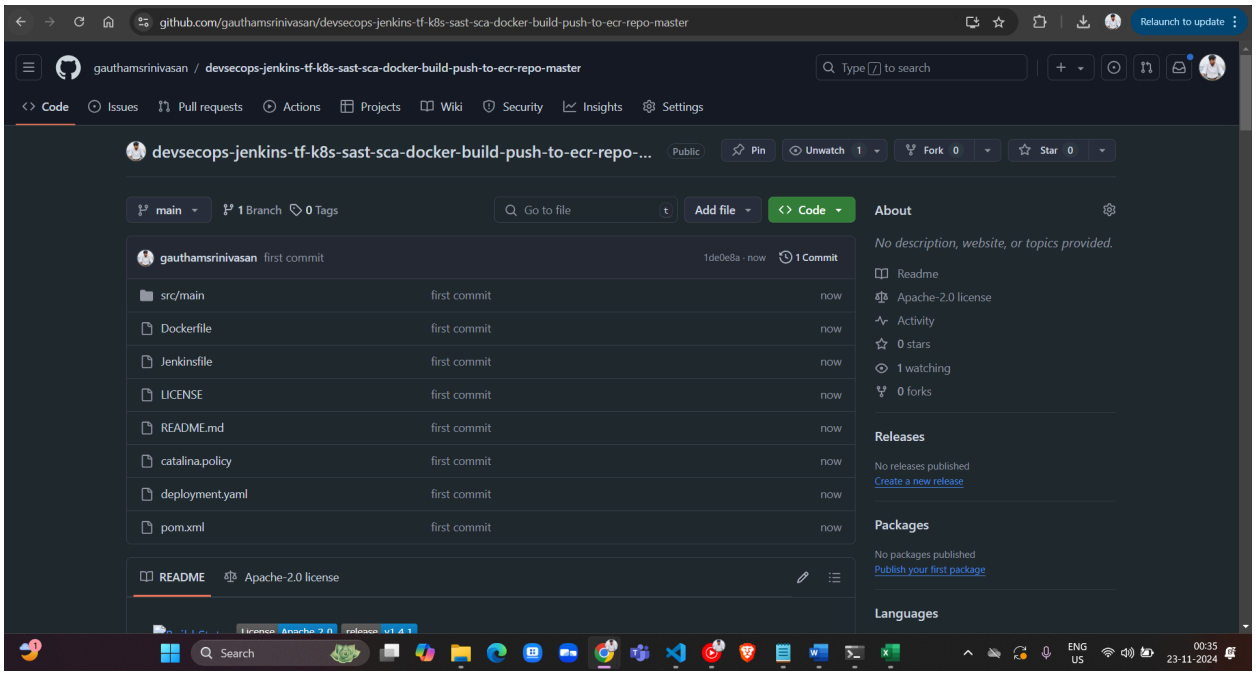


4. Add AWS Credentials to Jenkins

1. Go to Jenkins > Credentials > Add AWS Credentials.

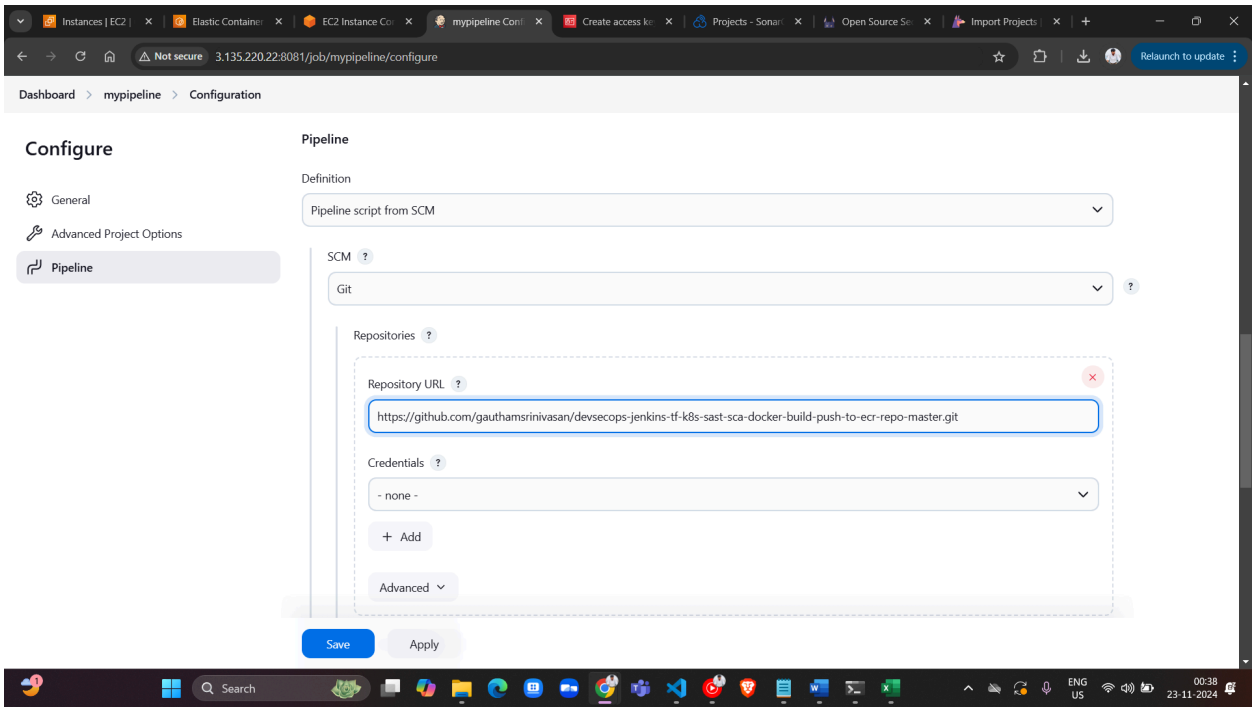


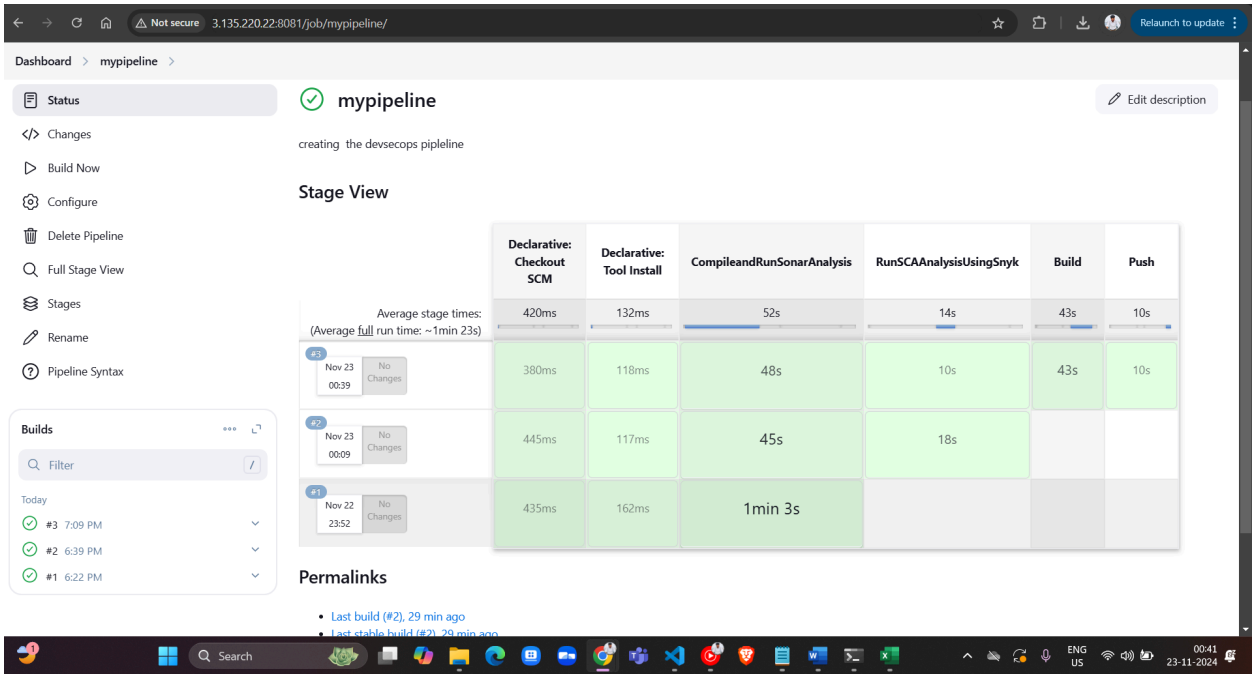
Save and publish the folder to github and verify the repository.



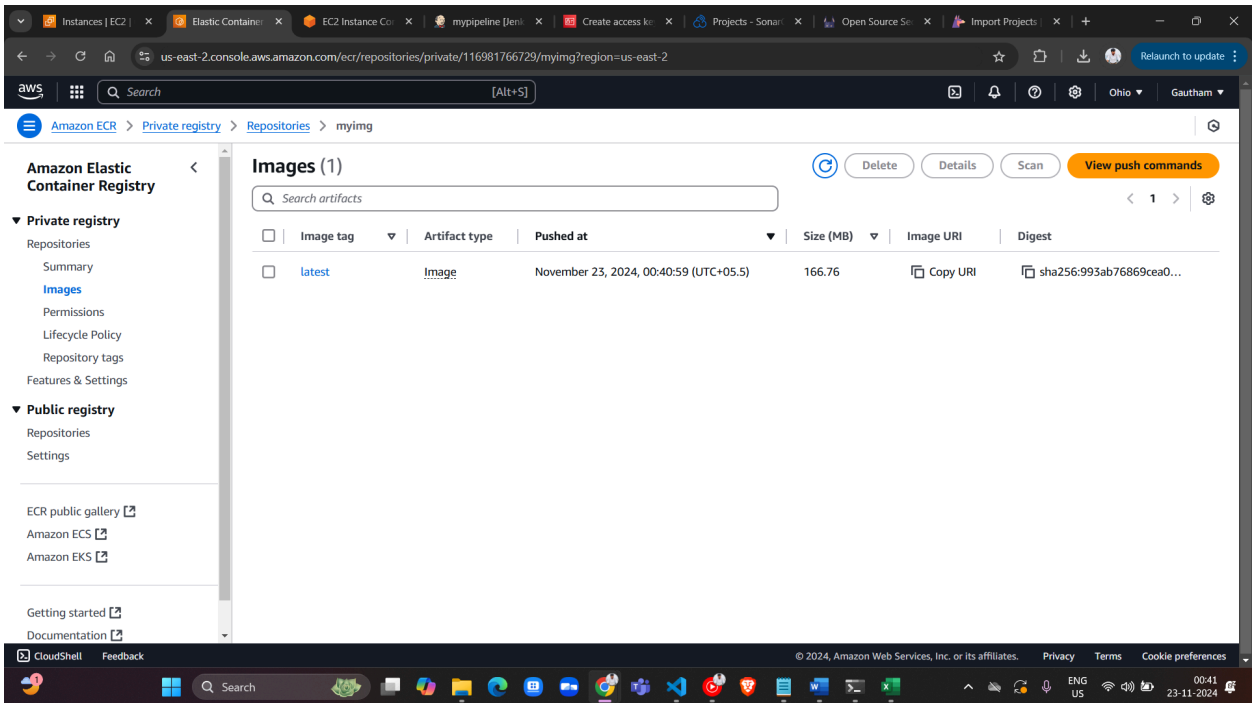
5. Push Image to ECR

1. Trigger the pipeline in Jenkins.





2. Verify the image in the AWS ECR Console.



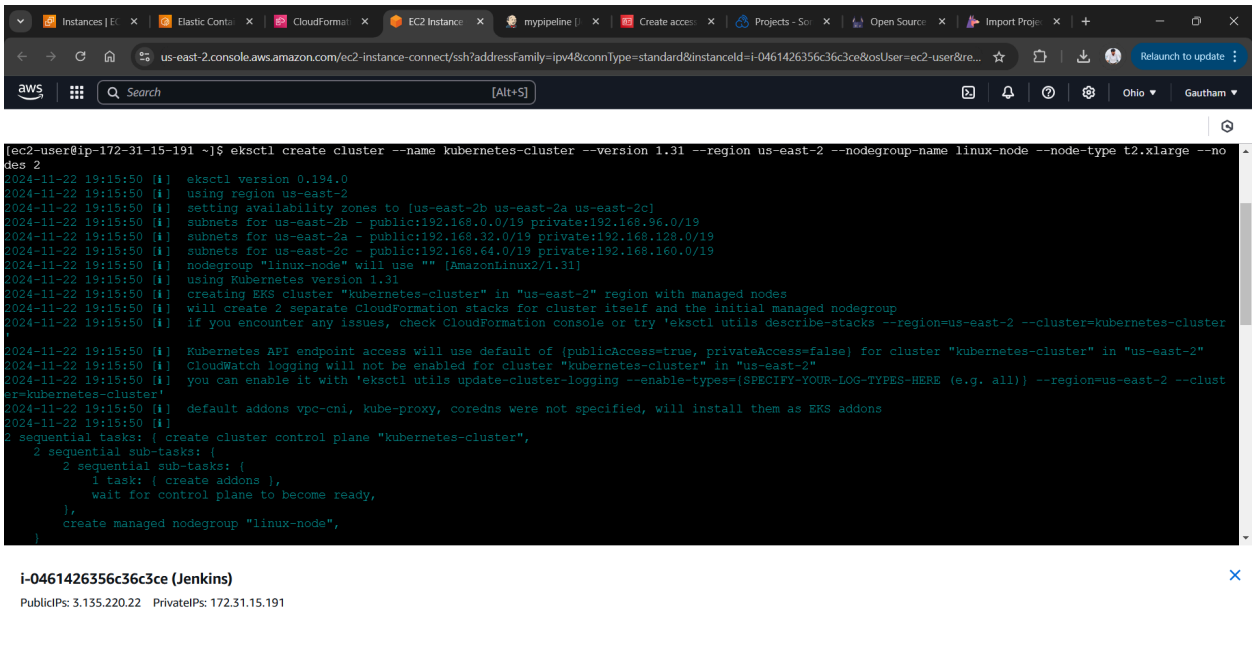
Step 8. Kubernetes Deployment

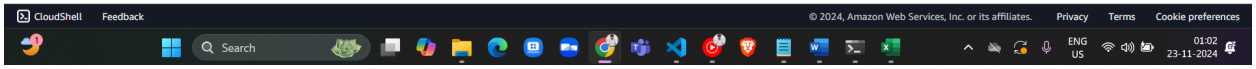
Create EKS Cluster

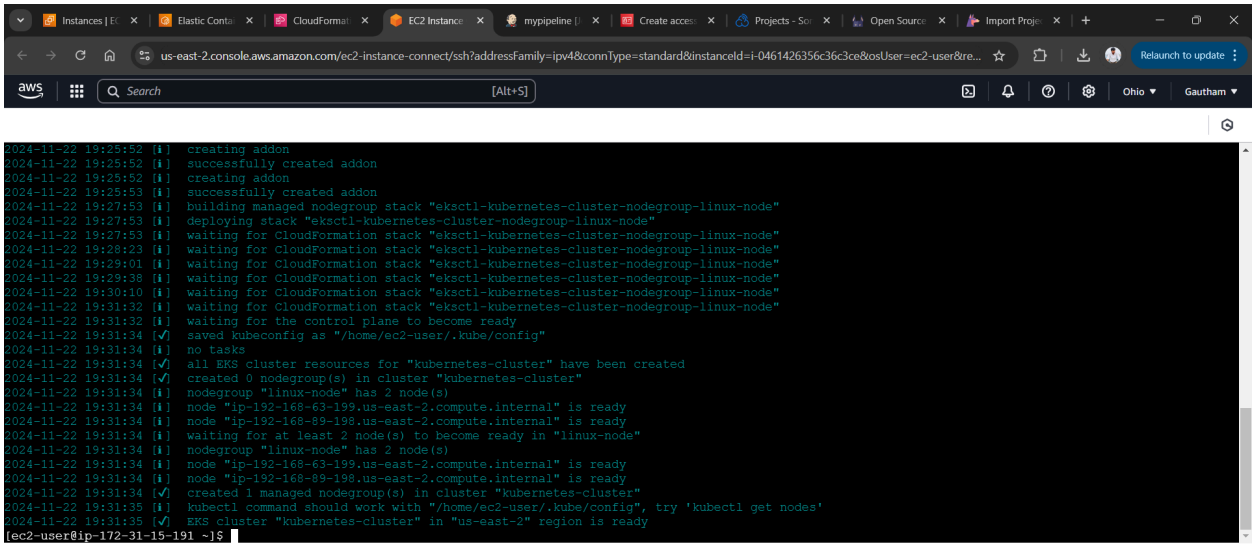
1. Create the cluster:

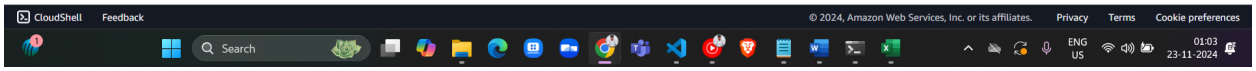
```
eksctl create cluster --name kubernetes-cluster --version 1.31
--region us-east-2 --nodegroup-name linux-node --node-type
t2.xlarge --nodes 2
```

```
(Wait approximately 10 mins for the process to complete..)
```

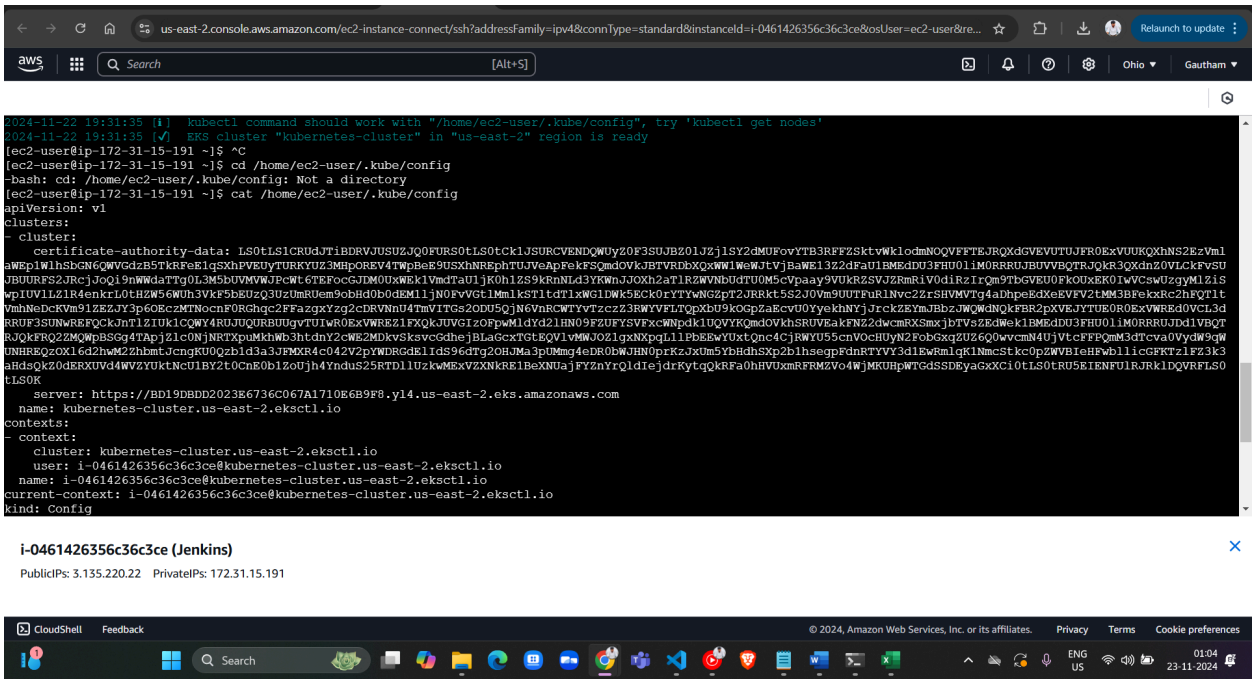






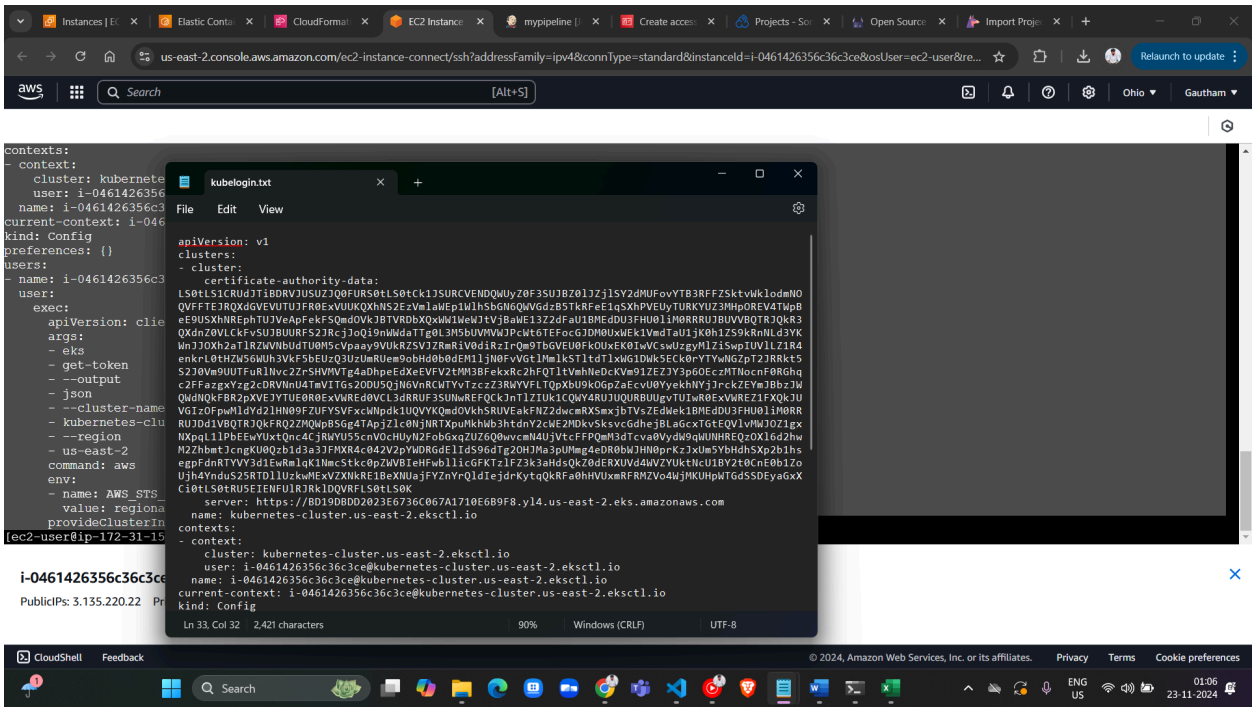


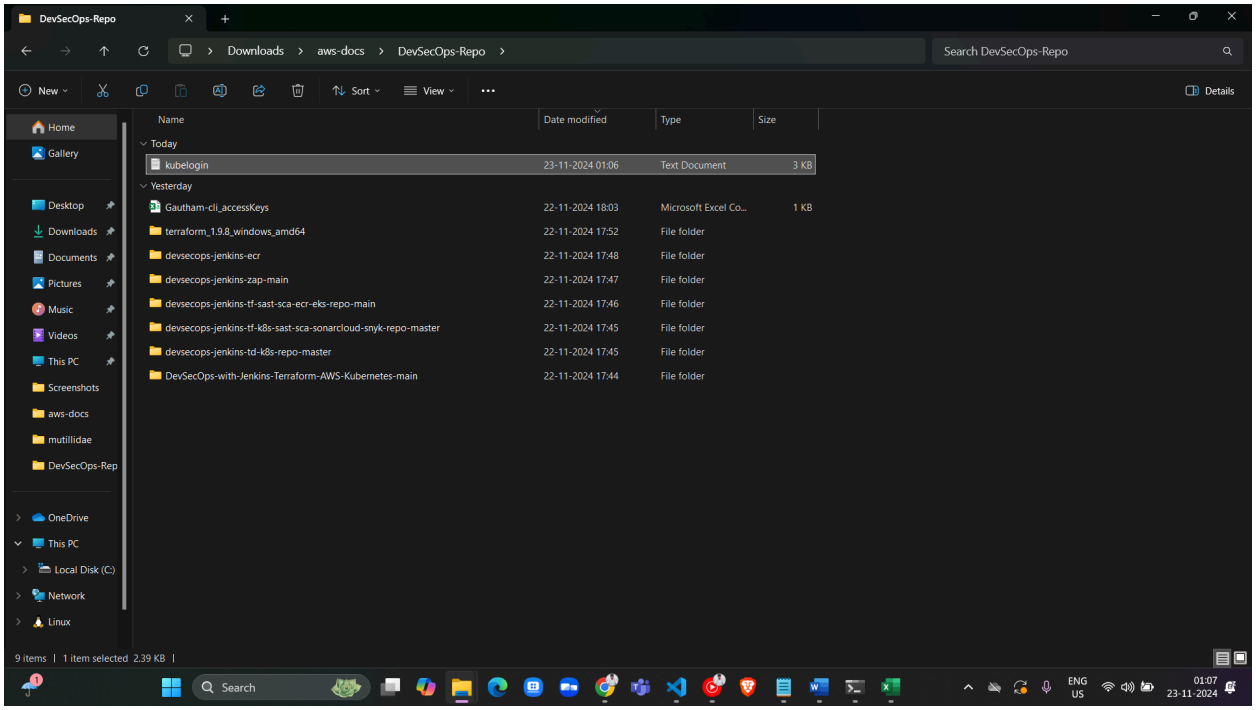
2. Go the “/home/ec2-user/.kube/config” and view the file



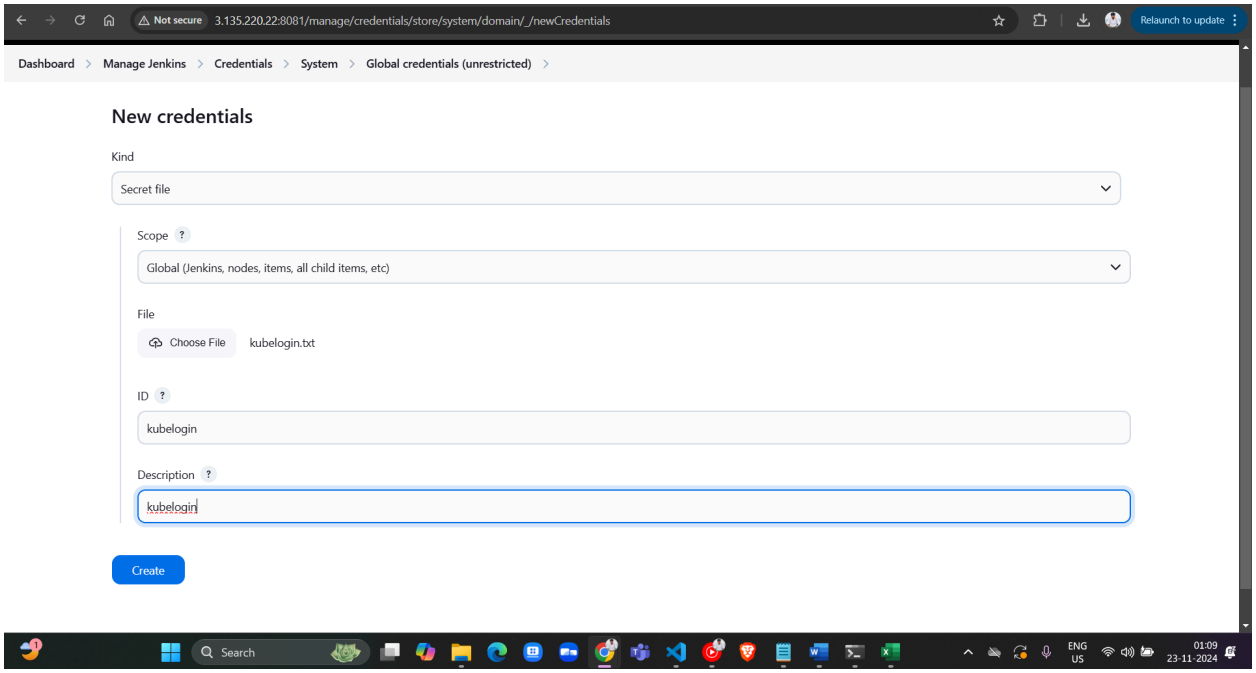
2. Configure Jenkins with Kubeconfig

1. Copy the kubeconfig instance content to the “kubelogin” file in the local machine.



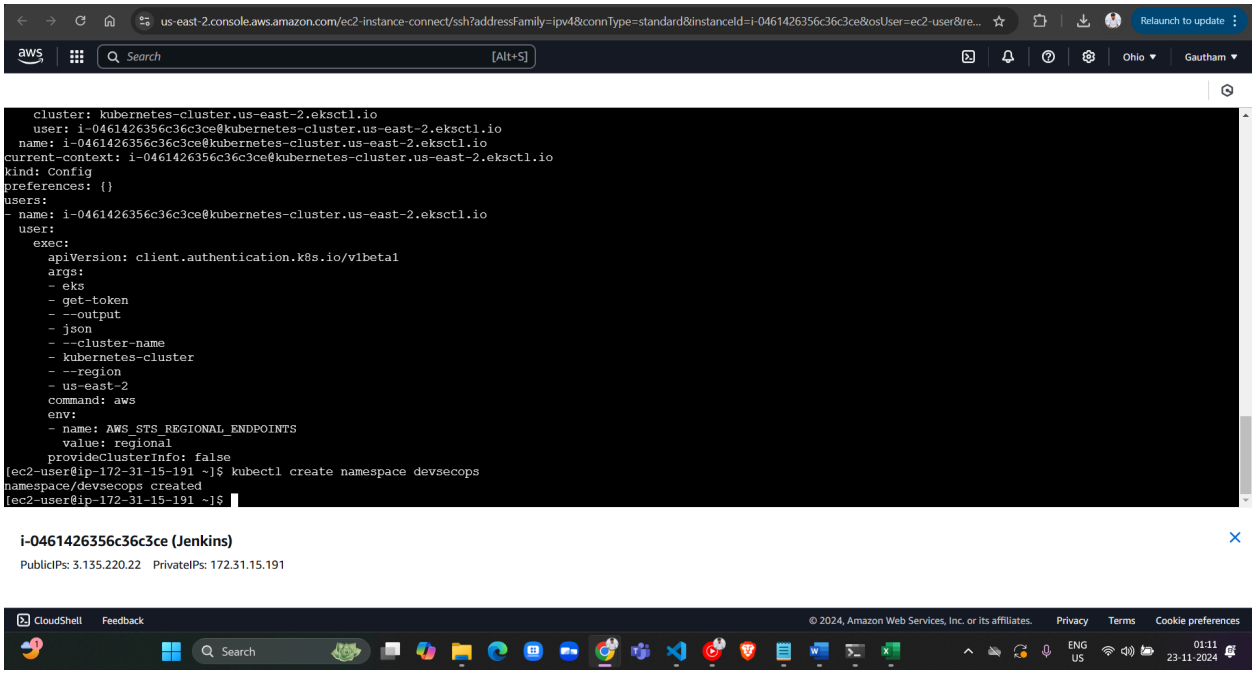


2. Add it as a secret file in Jenkins credentials.

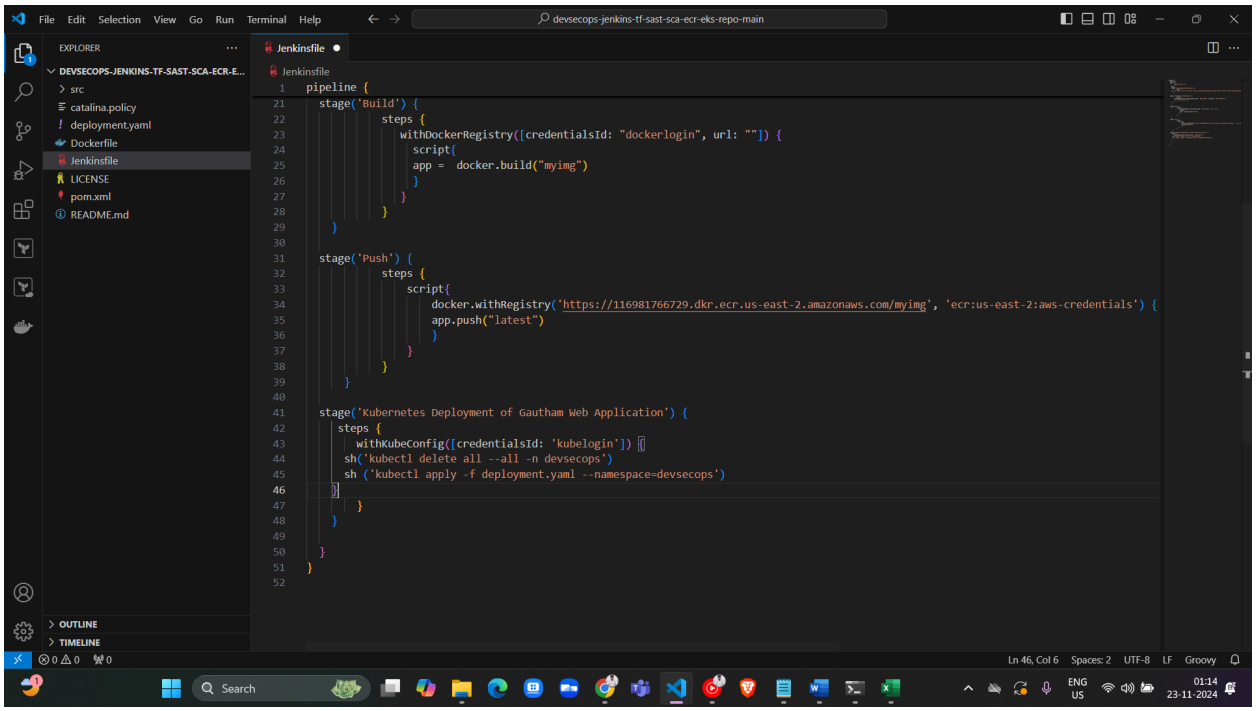


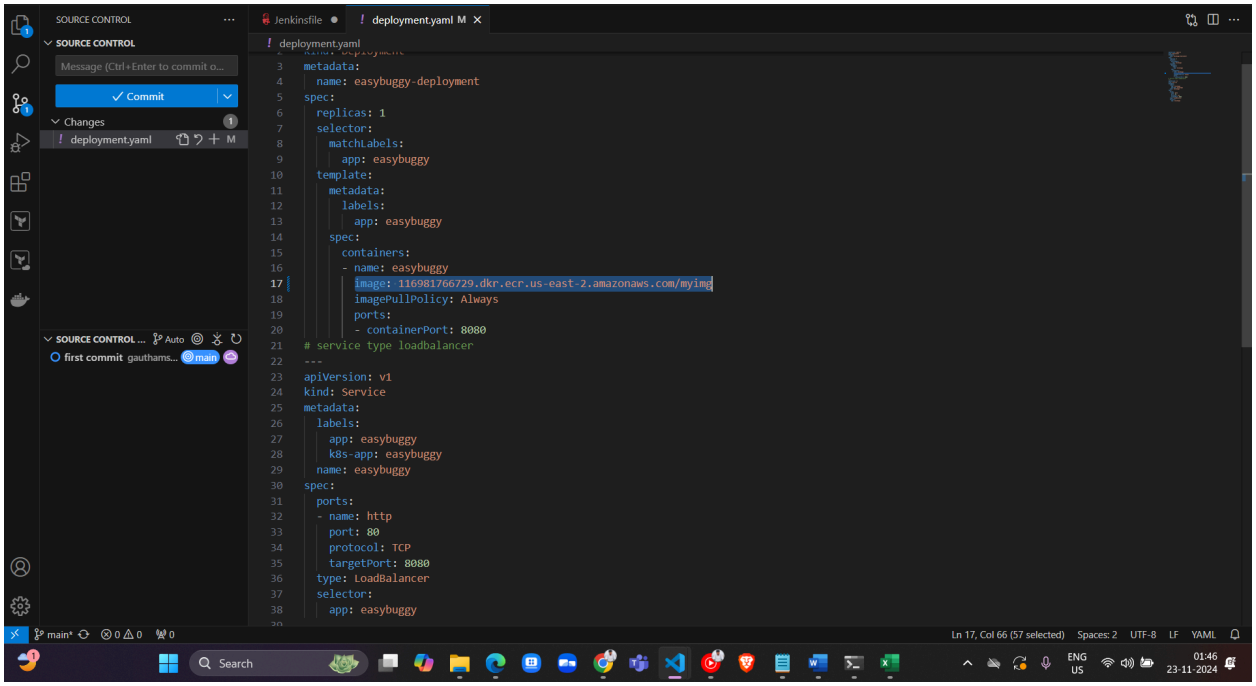
3. Deploy to Kubernetes

1. Create a namespace using ““kubectl create namespace devsecops”

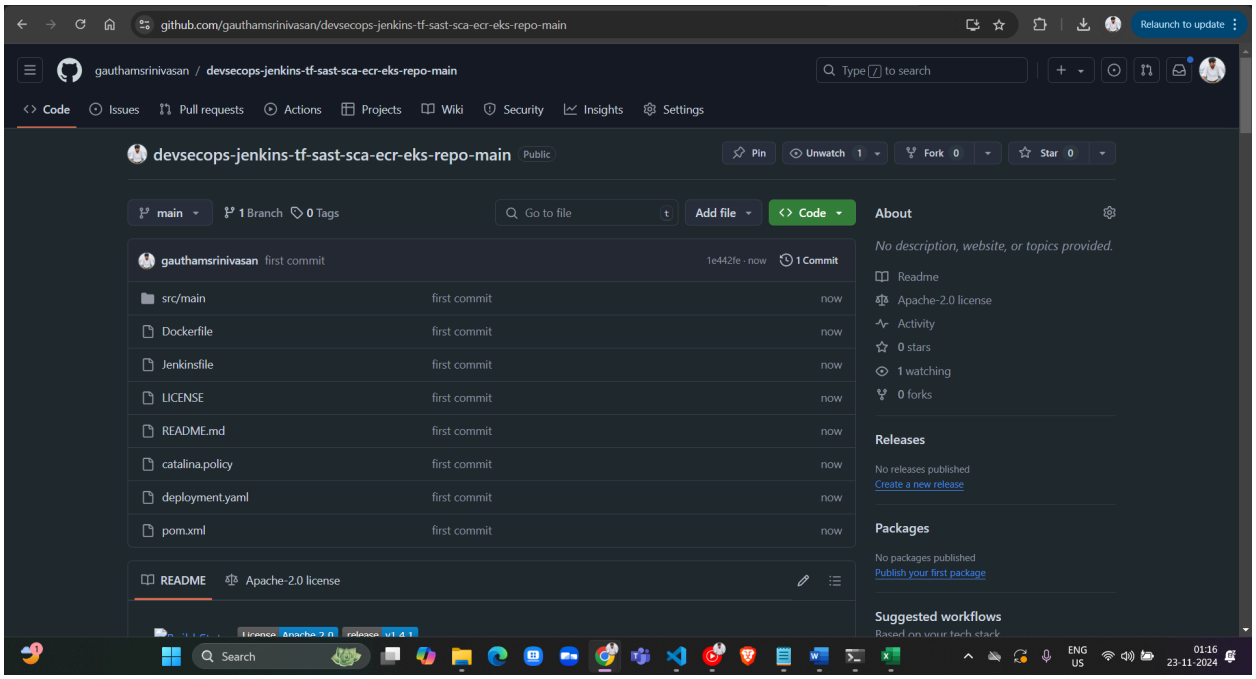


2. Update the `deployment.yaml` file with the ECR image URI.

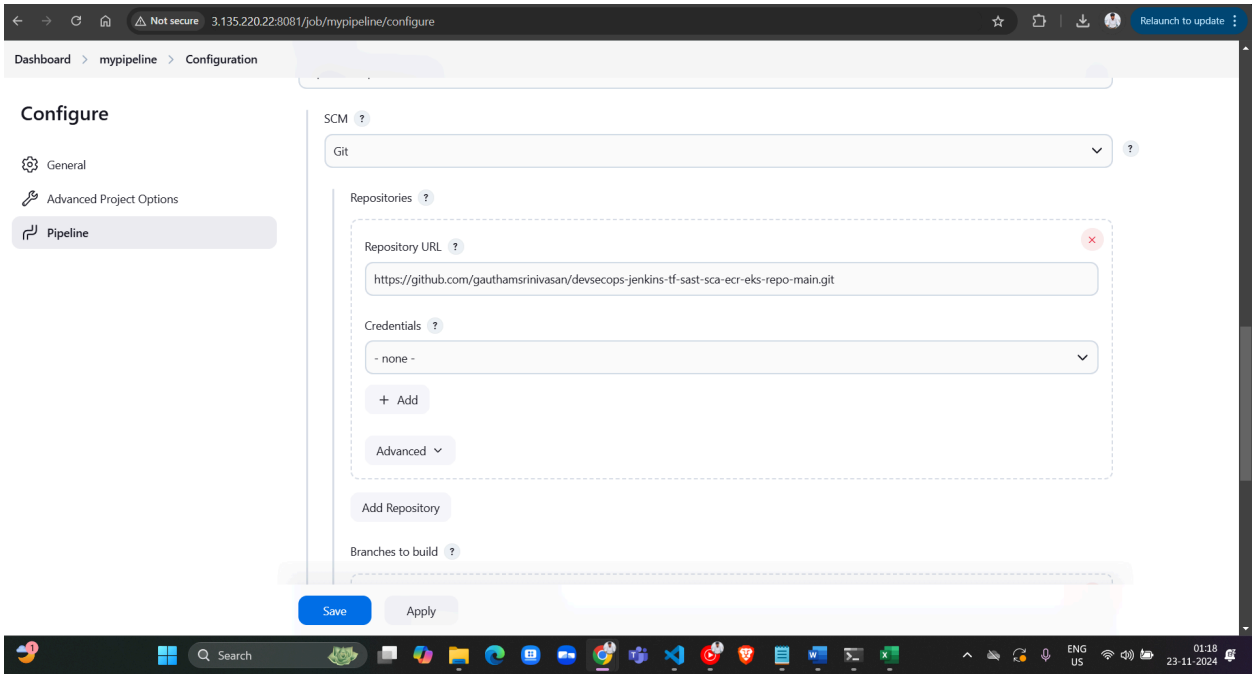




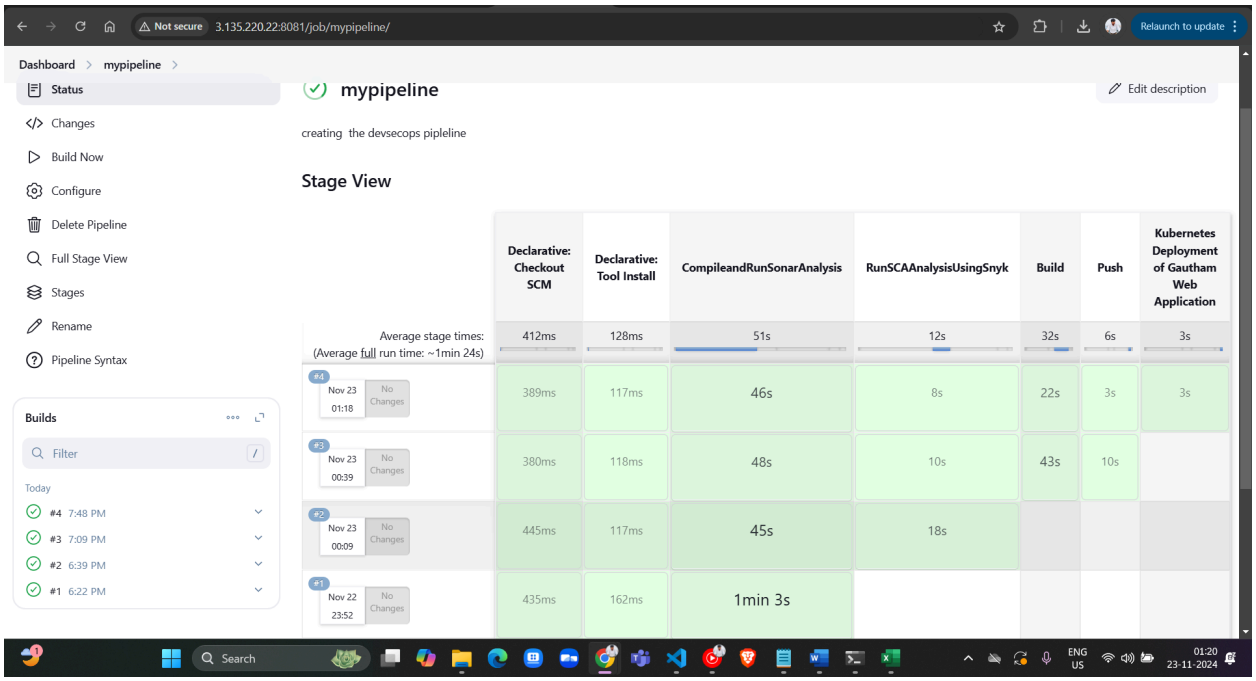
3. Push the folder to the github and verify



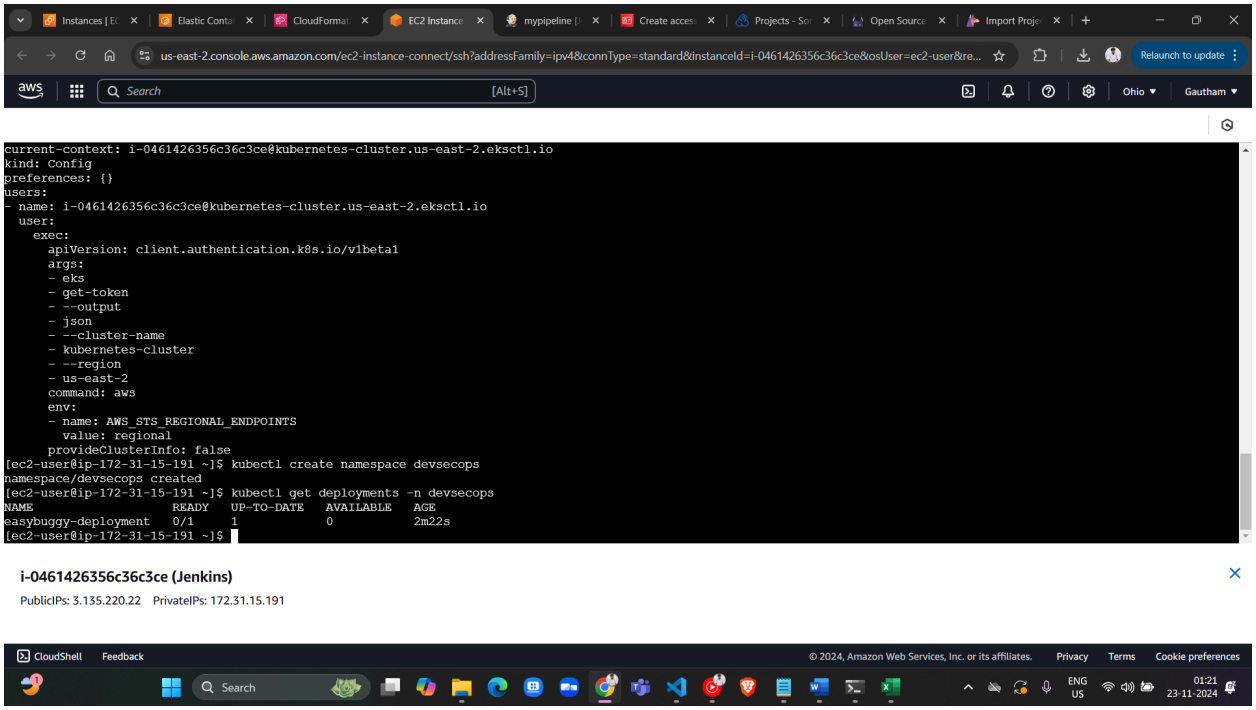
4. Change the jenkins github url



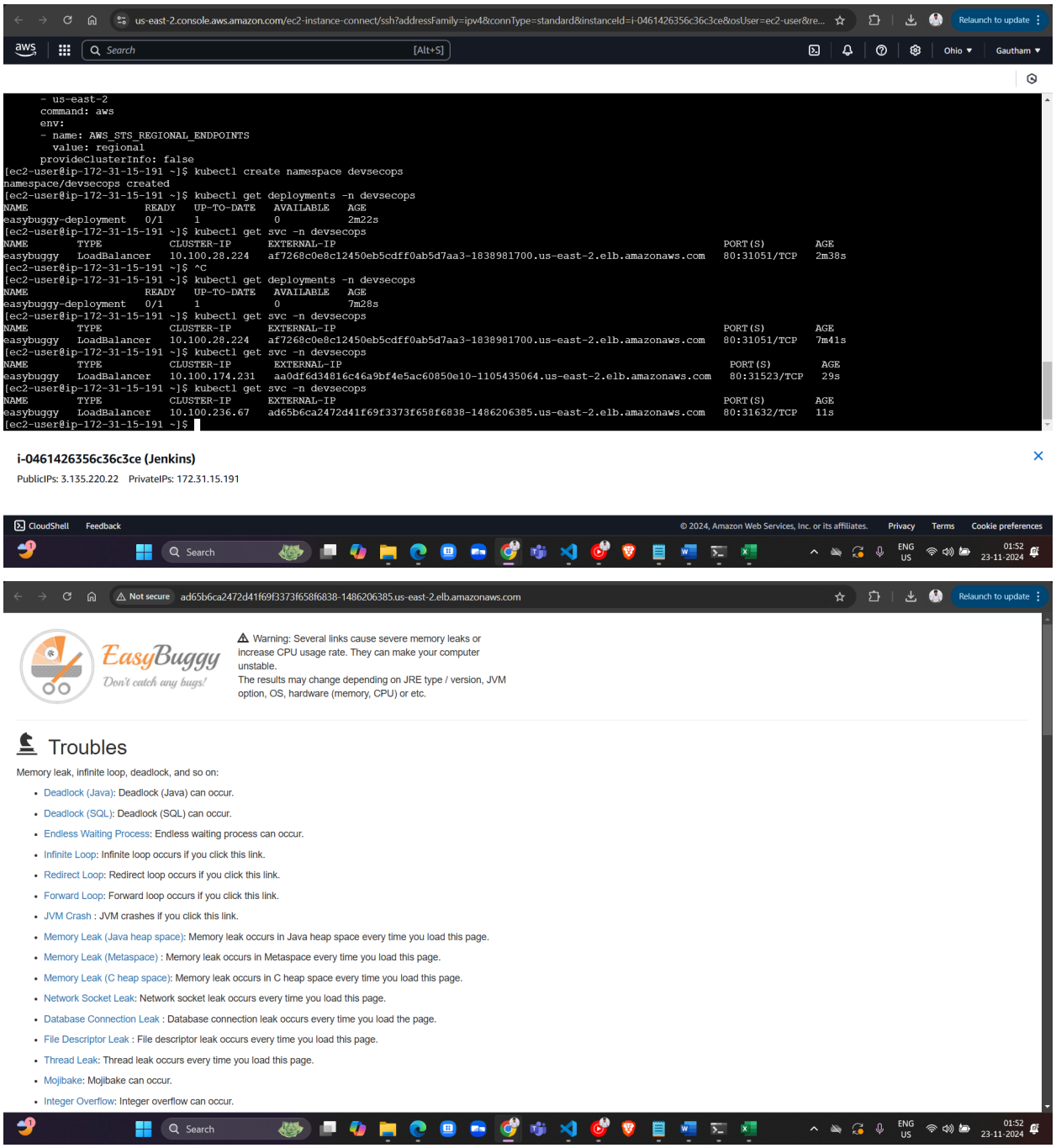
5. Build now and verify the output



6. Go to the Instance verify with the command “kubectl get deployments -n devsecops”

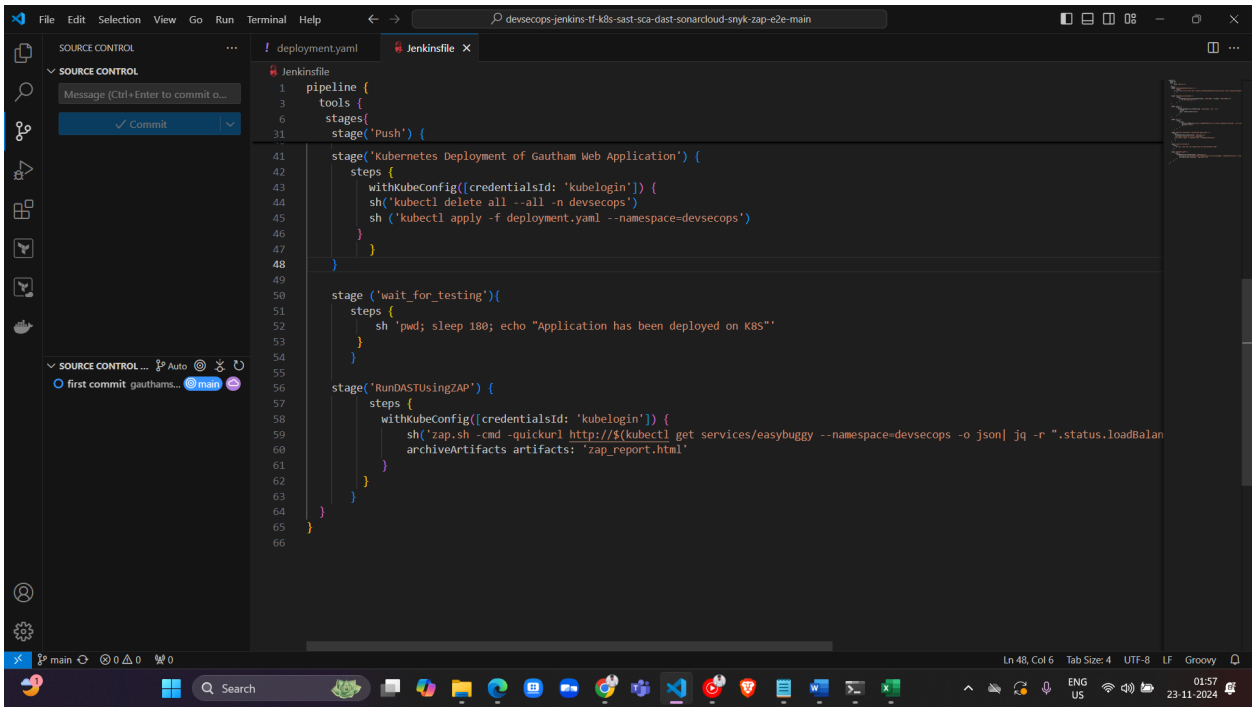


7. Check the service of the deployment using the service “kubectl get svc -n devsecops”

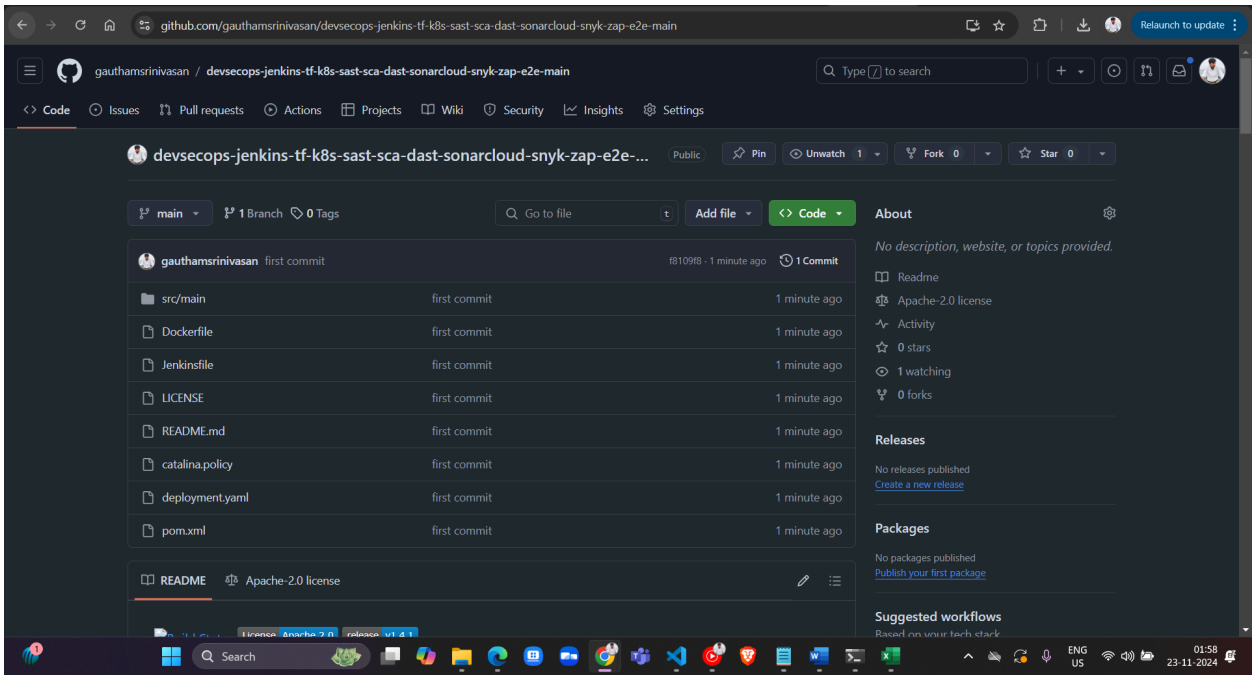


Step 9. DAST with OWASP ZAP

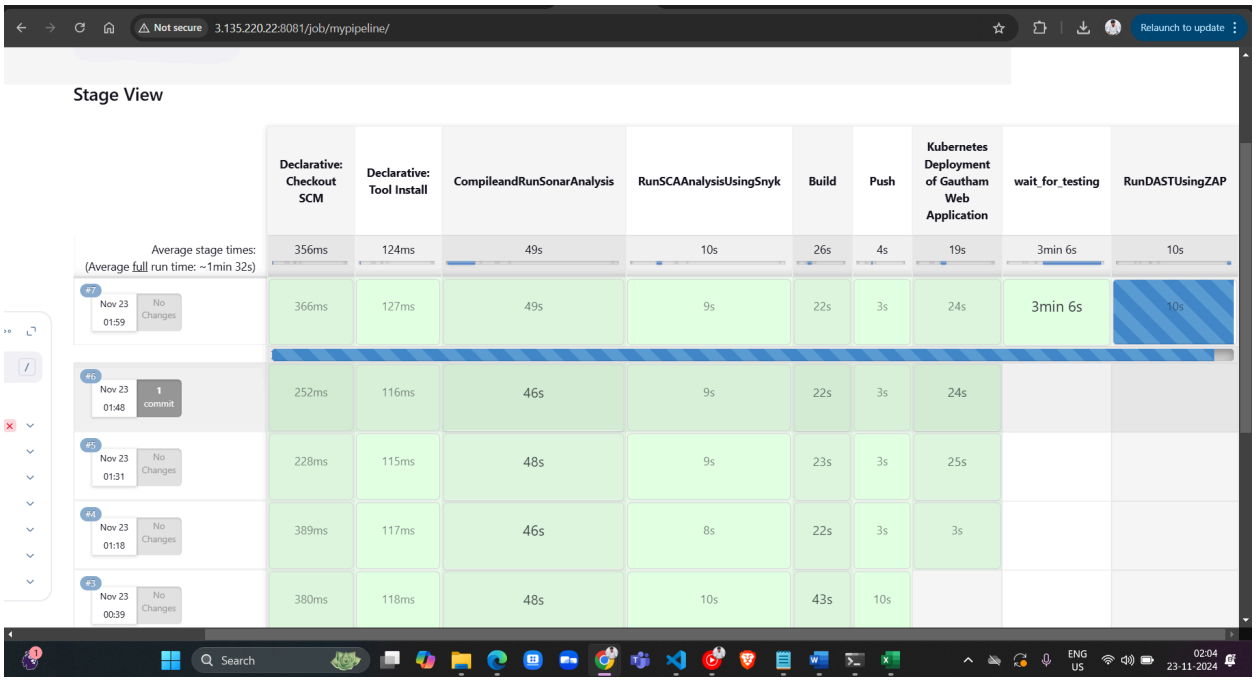
1. Add an OWASP ZAP stage to the pipeline.

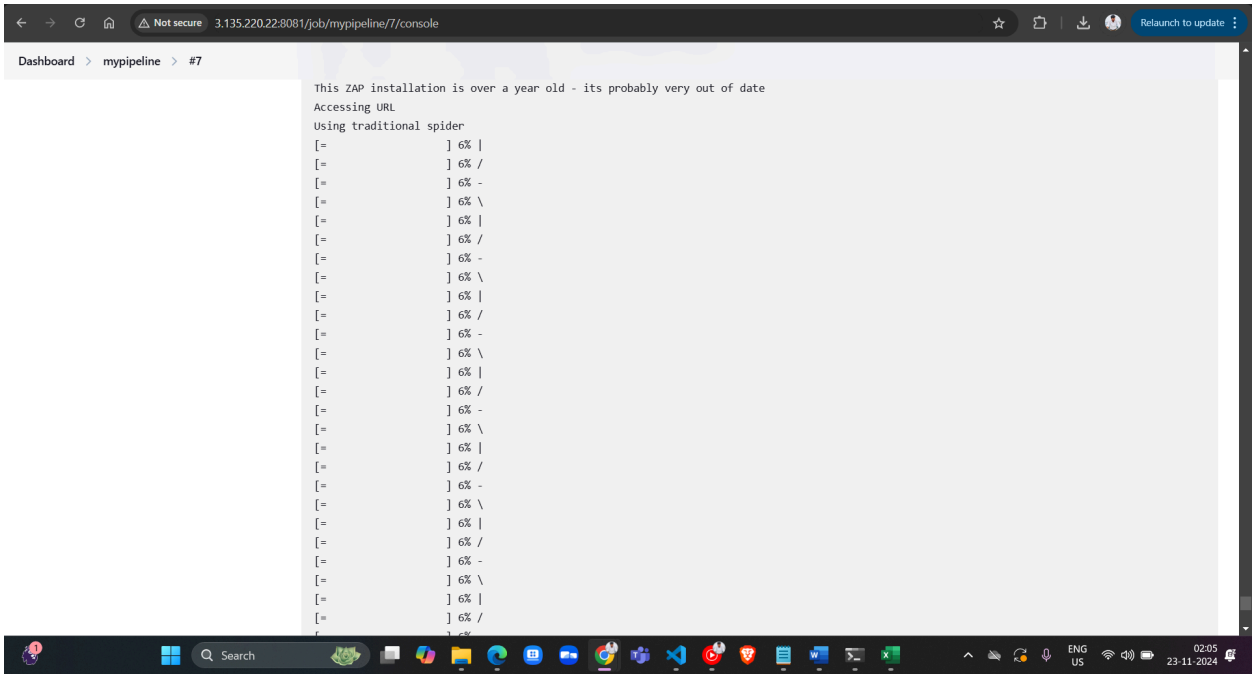


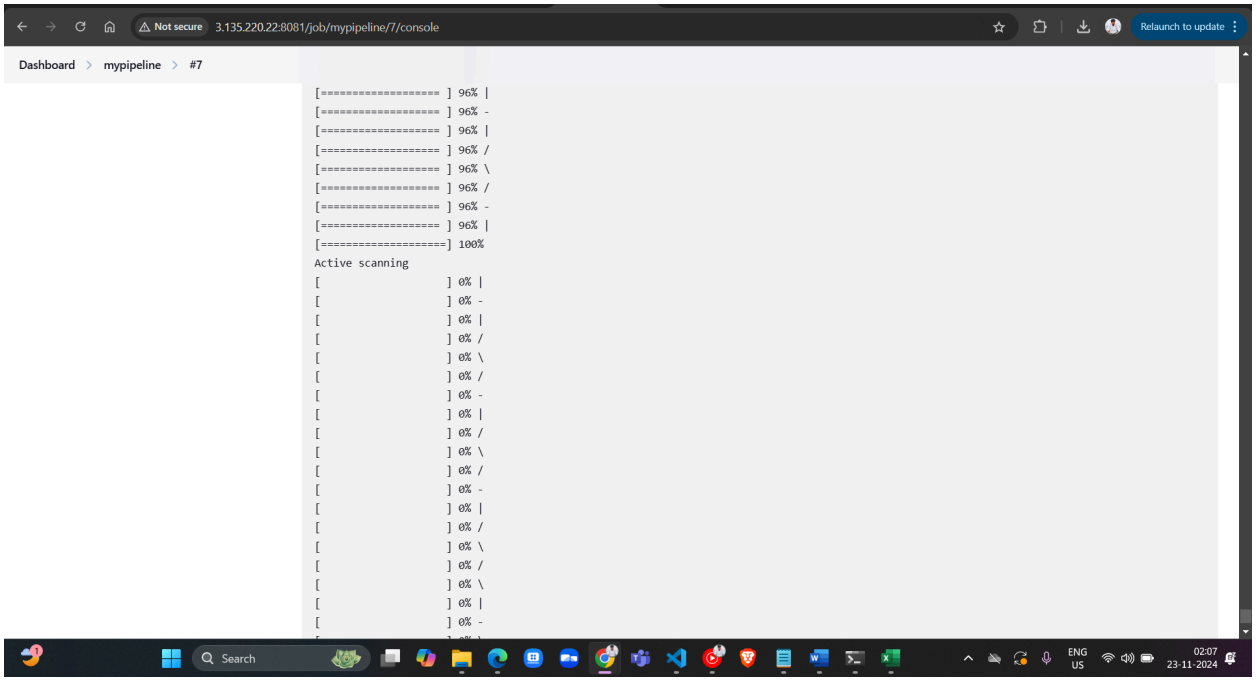
2. Publish your folder to the repo and verify

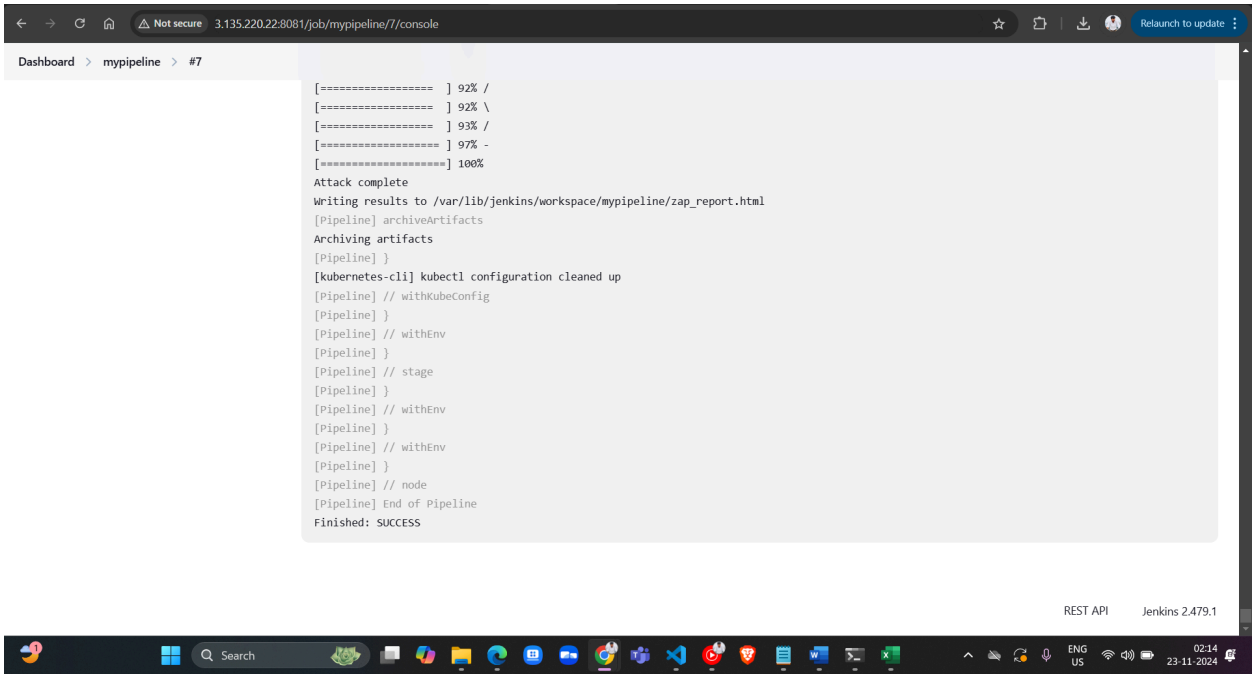


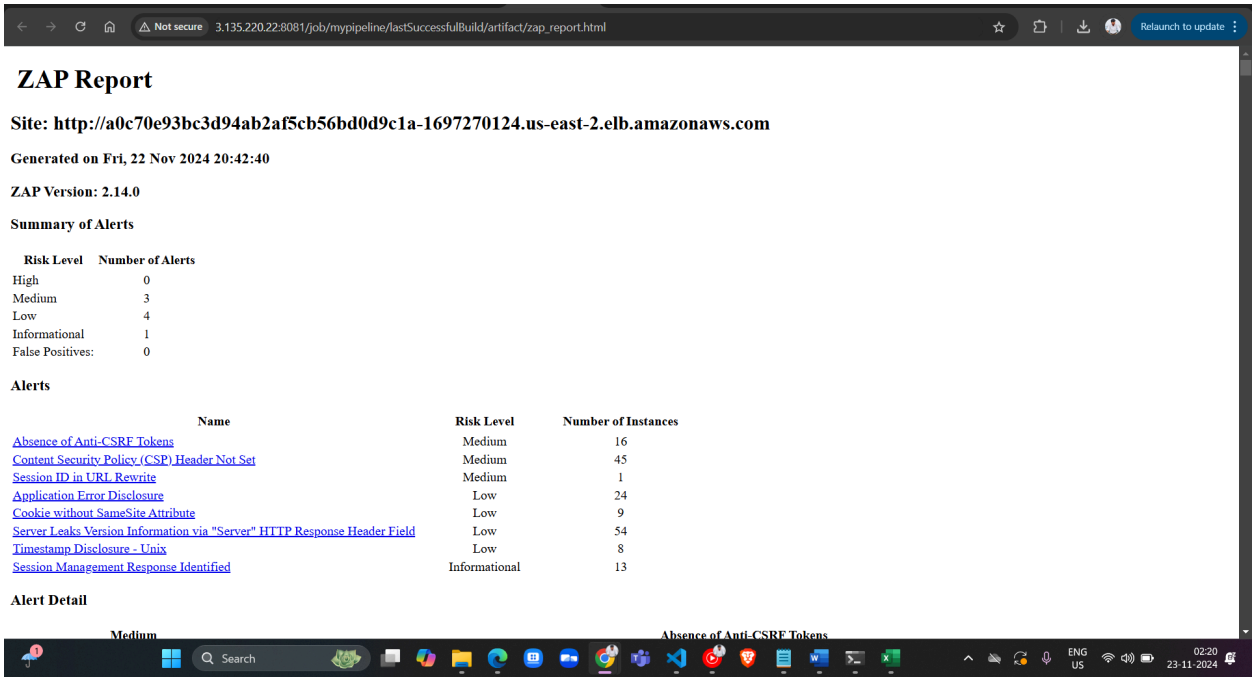
3. Trigger the pipeline to generate a report


Not secure 3.135.220.22:8081/job/mypipeline/
Relaunch to update
Stage View
Declarative: Checkout SCM
Declarative: Tool Install
CompileandRunSonarAnalysis
RunSCAAnalysisUsingSnyk
Build
Push
Kubernetes Deployment of Gautham Web Application
wait_for_testing
RunDASTUsingZAP
Average stage times:
(Average full run time: ~1min 32s)
356ms 124ms 49s 10s 26s 4s 19s 3min 6s 10s
#7 Nov 23 01:59 No Changes
366ms 127ms 49s 9s 22s 3s 24s 3min 6s 10s
#6 Nov 23 01:48 1 commit
252ms 116ms 46s 9s 22s 3s 24s
#5 Nov 23 01:31 No Changes
228ms 115ms 48s 9s 23s 3s 25s
#4 Nov 23 01:18 No Changes
389ms 117ms 46s 8s 22s 3s 3s
#3 Nov 23 00:39 No Changes
380ms 118ms 48s 10s 43s 10s
Search
ENG US
02:04 23-11-2024


Not secure 3.135.220.22:8081/job/mypipeline/7/console
Relaunch to update
Dashboard > mypipeline > #7
This ZAP installation is over a year old - its probably very out of date
Accessing URL
Using traditional spider
[=] 6% |
[=] 6% /
[=] 6% -
[=] 6% \
[=] 6% |
[=] 6% /
[=] 6% -
[=] 6% \
[=] 6% |
[=] 6% /
[=] 6% -
[=] 6% \
[=] 6% |
[=] 6% /
[=] 6% -
[=] 6% \
[=] 6% |
[=] 6% /
[=] 6% -
[=] 6% \
[=] 6% |
[=] 6% /
[=] 6% -
[=] 6% \
[=] 6% |
[=] 6% /
Search
ENG US
02:05 23-11-2024


Not secure 3.135.220.22:8081/job/mypipeline/7/console
Relaunch to update
Dashboard > mypipeline > #7
[===================] 96% |
[===================] 96% -
[===================] 96% |
[===================] 96% /
[===================] 96% \
[===================] 96% /
[===================] 96% -
[===================] 96% |
[====================] 100%
Active scanning
[ ] 0% |
[ ] 0% -
[ ] 0% |
[ ] 0% /
[ ] 0% \
[ ] 0% /
[ ] 0% -
[ ] 0% |
[ ] 0% /
[ ] 0% \
[ ] 0% /
[ ] 0% -
[ ] 0% |
[ ] 0% /
[ ] 0% \
[ ] 0% /
[ ] 0% \
[ ] 0% |
[ ] 0% -
Search
ENG US
02:07 23-11-2024


Not secure 3.135.220.22:8081/job/mypipeline/7/console
Relaunch to update
Dashboard > mypipeline > #7
[==================] 92% /
[==================] 92% \
[==================] 93% /
[===================] 97% -
[====================] 100%
Attack complete
Writing results to /var/lib/jenkins/workspace/mypipeline/zap_report.html
[Pipeline] archiveArtifacts
Archiving artifacts
[Pipeline] }
[kubernetes-cli] kubectl configuration cleaned up
[Pipeline] // withKubeConfig
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
REST API
Jenkins 2.479.1
Search
ENG US
02:14 23-11-2024



Not secure 3.135.220.22:8081/job/mypipeline/lastSuccessfulBuild/artifact/zap_report.html

Relaunch to update

ZAP Report

Site: http://a0c70e93bc3d94ab2af5cb56bd0d9c1a-1697270124.us-east-2.elb.amazonaws.com

Generated on Fri, 22 Nov 2024 20:42:40

ZAP Version: 2.14.0

Summary of Alerts

Risk Level	Number of Alerts
High	0
Medium	3
Low	4
Informational	1
False Positives:	0

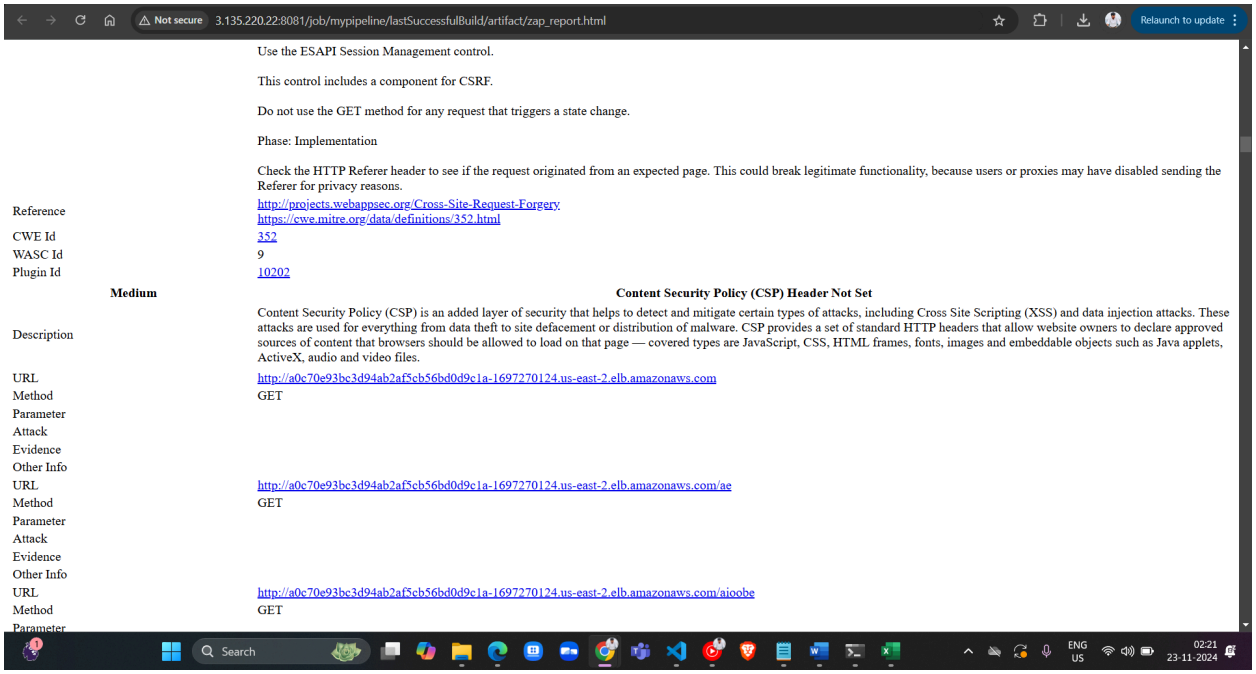
Alerts

Name	Risk Level	Number of Instances
Absence of Anti-CSRF Tokens	Medium	16
Content Security Policy (CSP) Header Not Set	Medium	45
Session ID in URL Rewrite	Medium	1
Application Error Disclosure	Low	24
Cookie without SameSite Attribute	Low	9
Server Leaks Version Information via "Server" HTTP Response Header Field	Low	54
Timestamp Disclosure - Unix	Low	8
Session Management Response Identified	Informational	13

Alert Detail

Medium **Absence of Anti-CSRF Tokens**

Search ENG US 02:20 23-11-2024



Not secure 3.135.220.22:8081/job/mypipeline/lastSuccessfulBuild/artifact/zap_report.html

Relaunch to update

	Use the ESAPI Session Management control. This control includes a component for CSRF. Do not use the GET method for any request that triggers a state change. Phase: Implementation Check the HTTP Referer header to see if the request originated from an expected page. This could break legitimate functionality, because users or proxies may have disabled sending the Referer for privacy reasons.
Reference	http://projects.webappsec.org/Cross-Site-Request-Forgery https://cwe.mitre.org/data/definitions/352.html
CWE Id	352
WASC Id	9
Plugin Id	10202

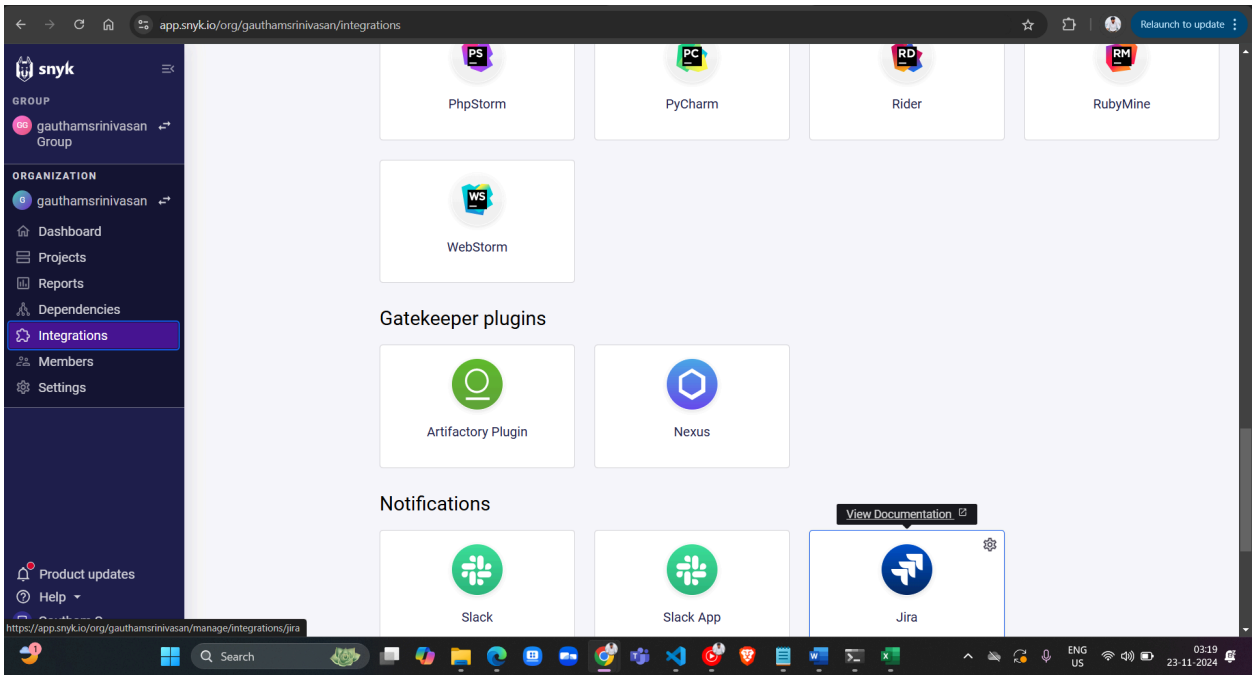
Medium **Content Security Policy (CSP) Header Not Set**

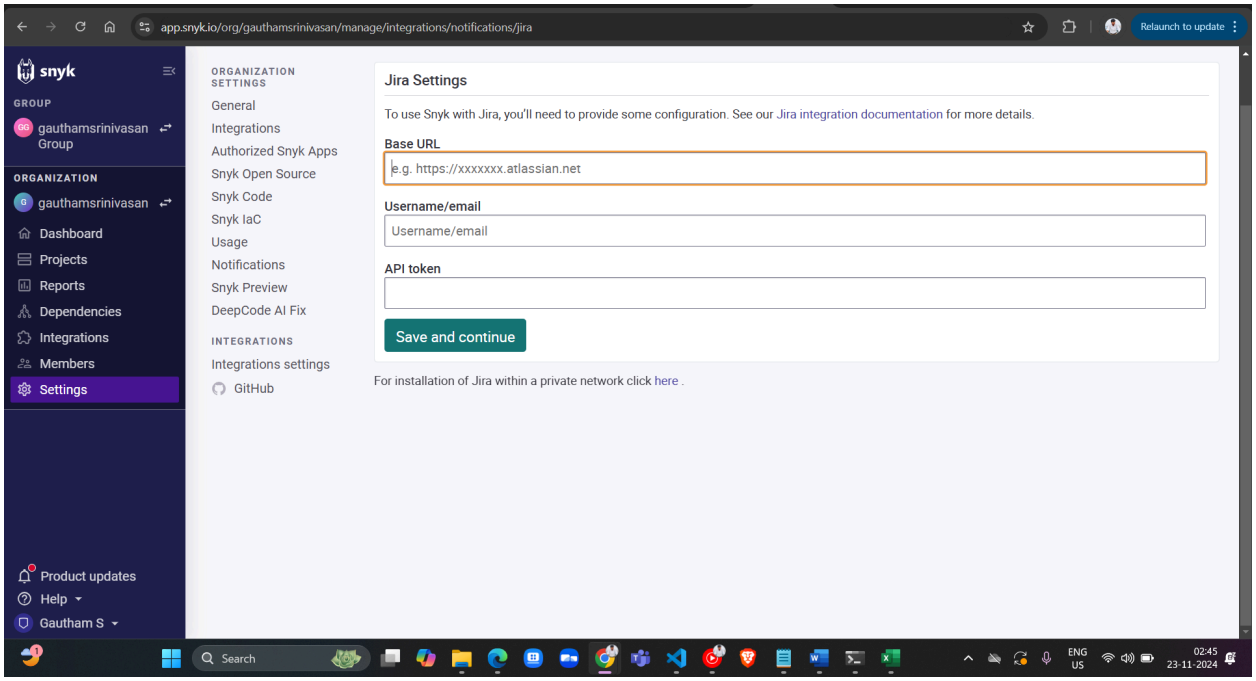
Description	Content Security Policy (CSP) is an added layer of security that helps to detect and mitigate certain types of attacks, including Cross Site Scripting (XSS) and data injection attacks. These attacks are used for everything from data theft to site defacement or distribution of malware. CSP provides a set of standard HTTP headers that allow website owners to declare approved sources of content that browsers should be allowed to load on that page — covered types are JavaScript, CSS, HTML frames, fonts, images and embeddable objects such as Java applets, ActiveX, audio and video files.
URL	http://a0c70e93bc3d94ab2af5cb56bd0d9c1a-1697270124.us-east-2.elb.amazonaws.com
Method	GET
Parameter	
Attack	
Evidence	
Other Info	
URL	http://a0c70e93bc3d94ab2af5cb56bd0d9c1a-1697270124.us-east-2.elb.amazonaws.com/ae
Method	GET
Parameter	
Attack	
Evidence	
Other Info	
URL	http://a0c70e93bc3d94ab2af5cb56bd0d9c1a-1697270124.us-east-2.elb.amazonaws.com/ajoobe
Method	GET
Parameter	

Search ENG US 02:21 23-11-2024

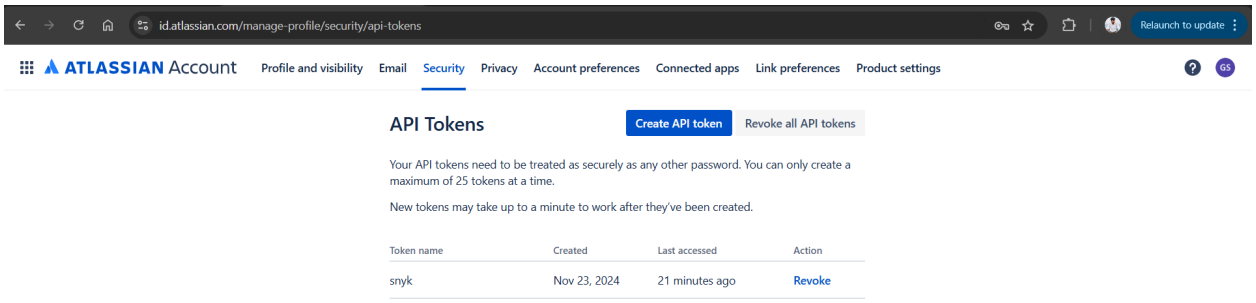
Step 10. Integrate Snyk with Jira to Assign Tickets for Vulnerabilities

1. Go to Snyk Console > Settings > Plans and pricing > Enable the Free trial for 14 days
2. Navigate to Integrations > Jira and enter the base url of your project board.

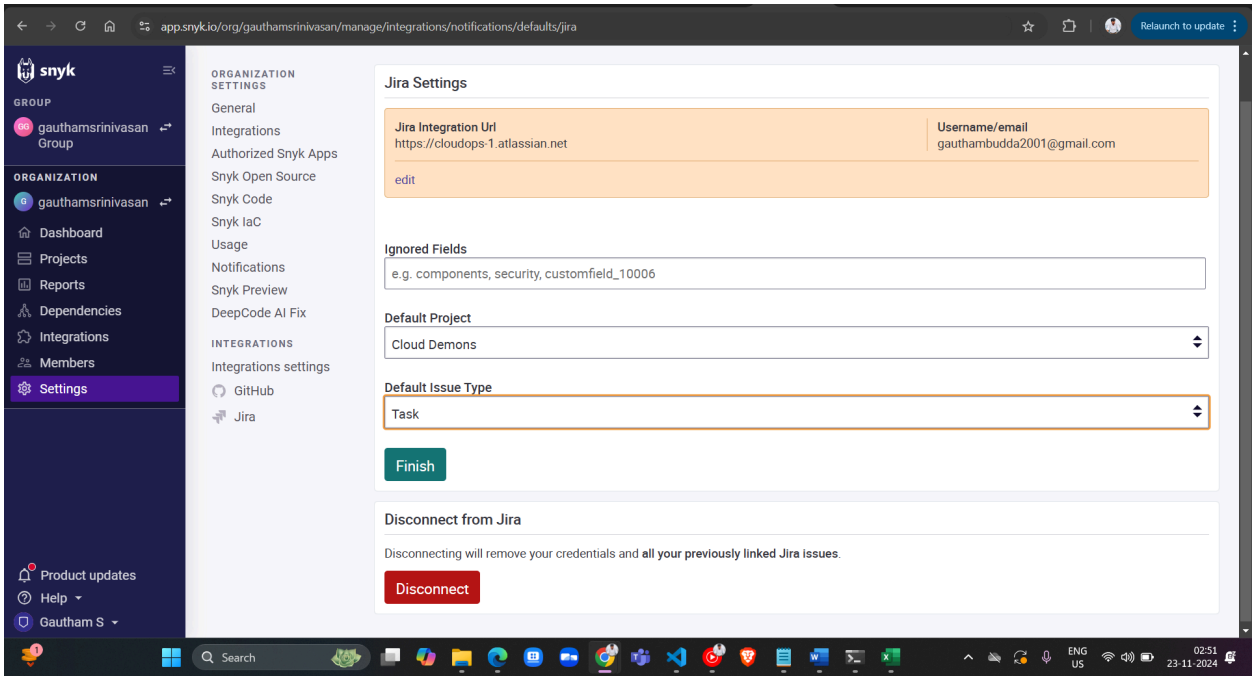




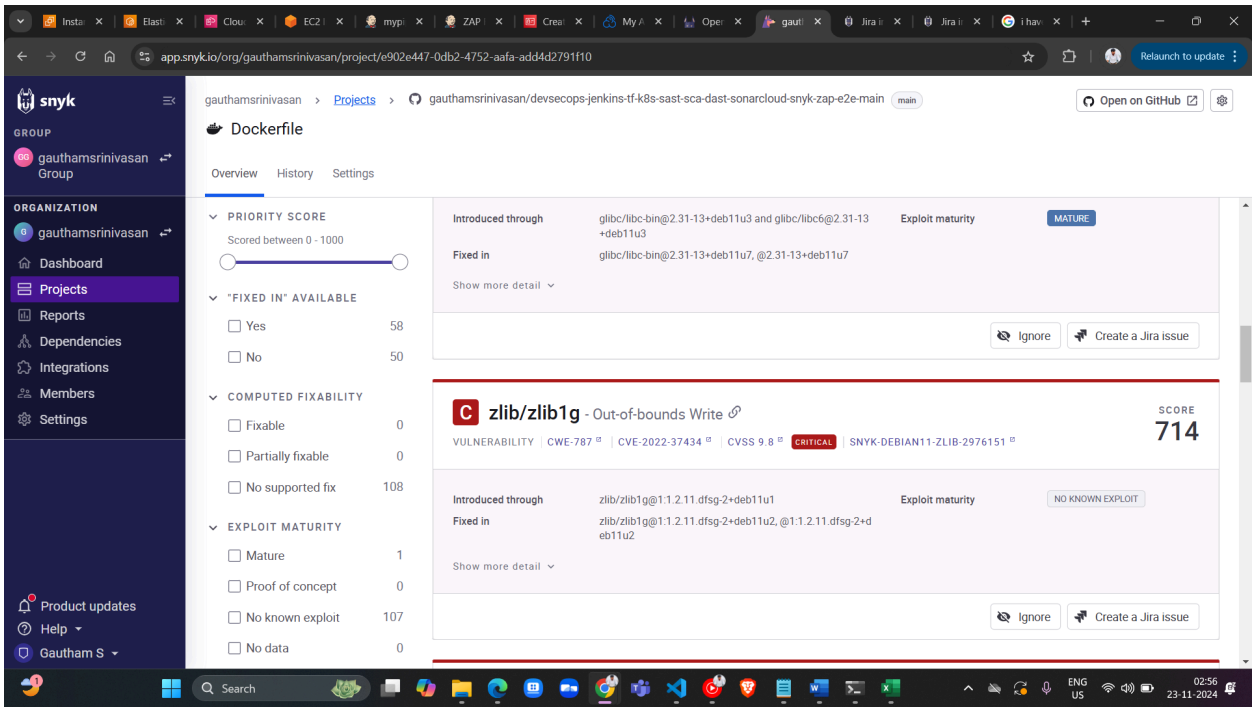
3. Go to Jira > Profile > Security > Create Tokens and copy it.



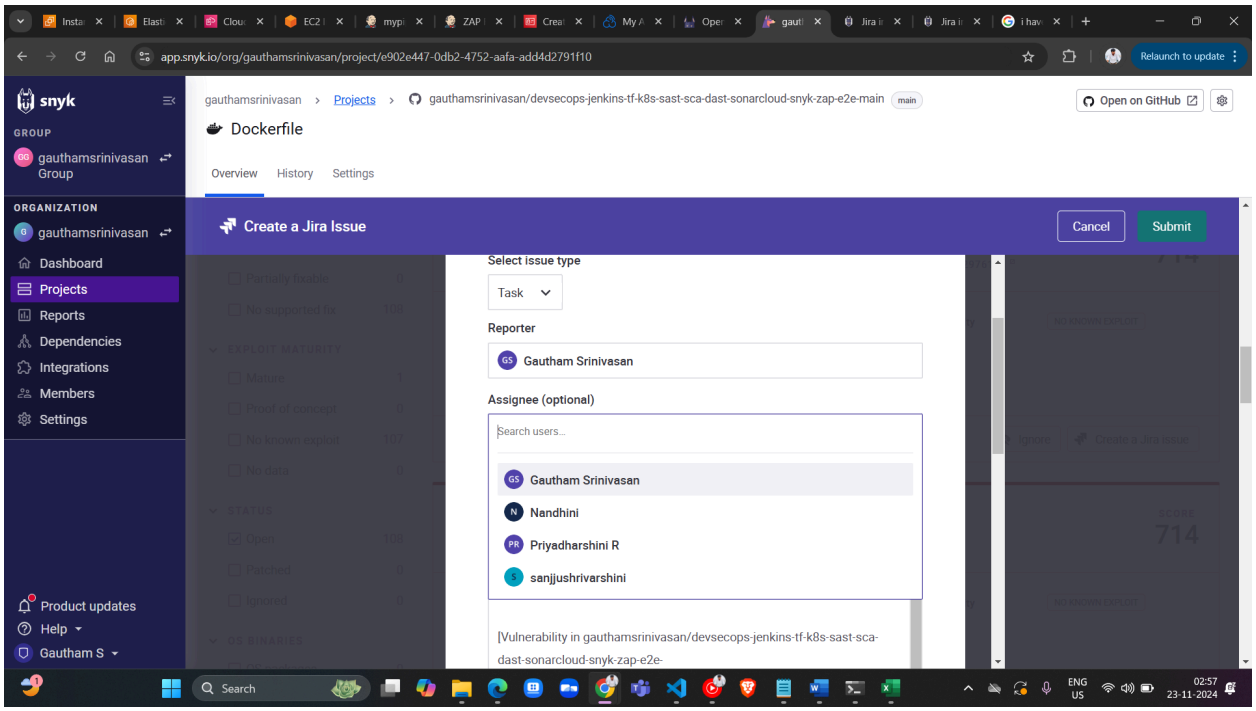
4. Verify your connection and jira project and select the type and click finish

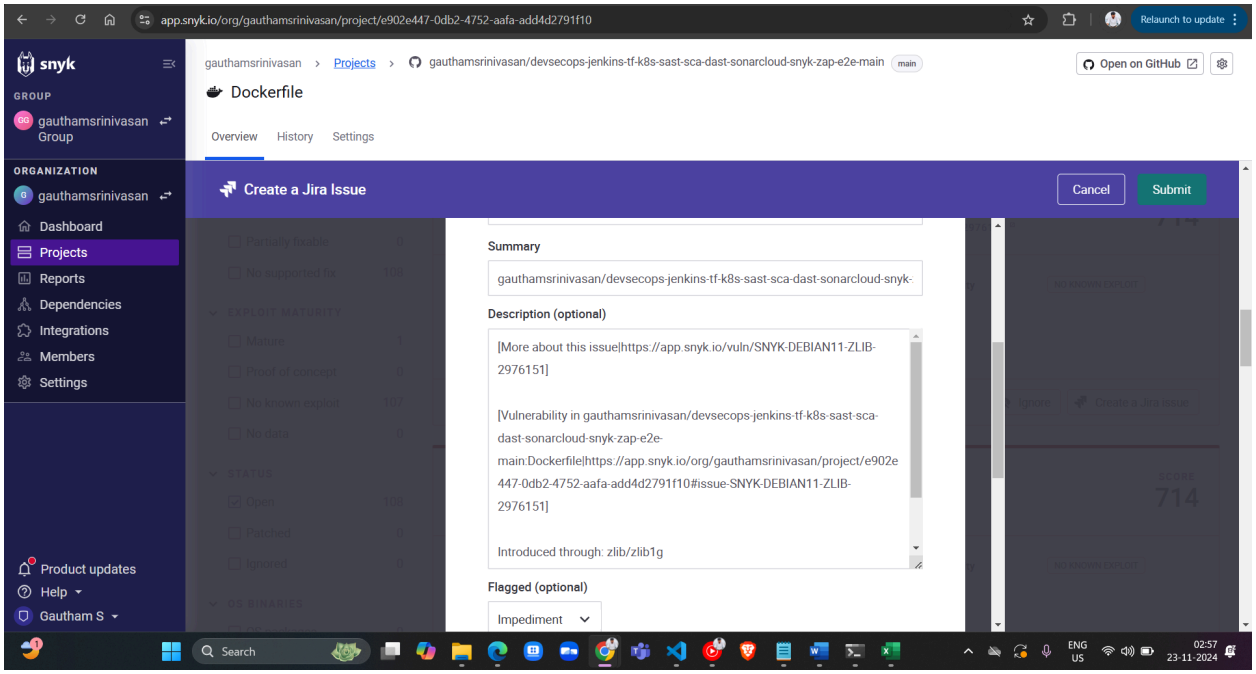


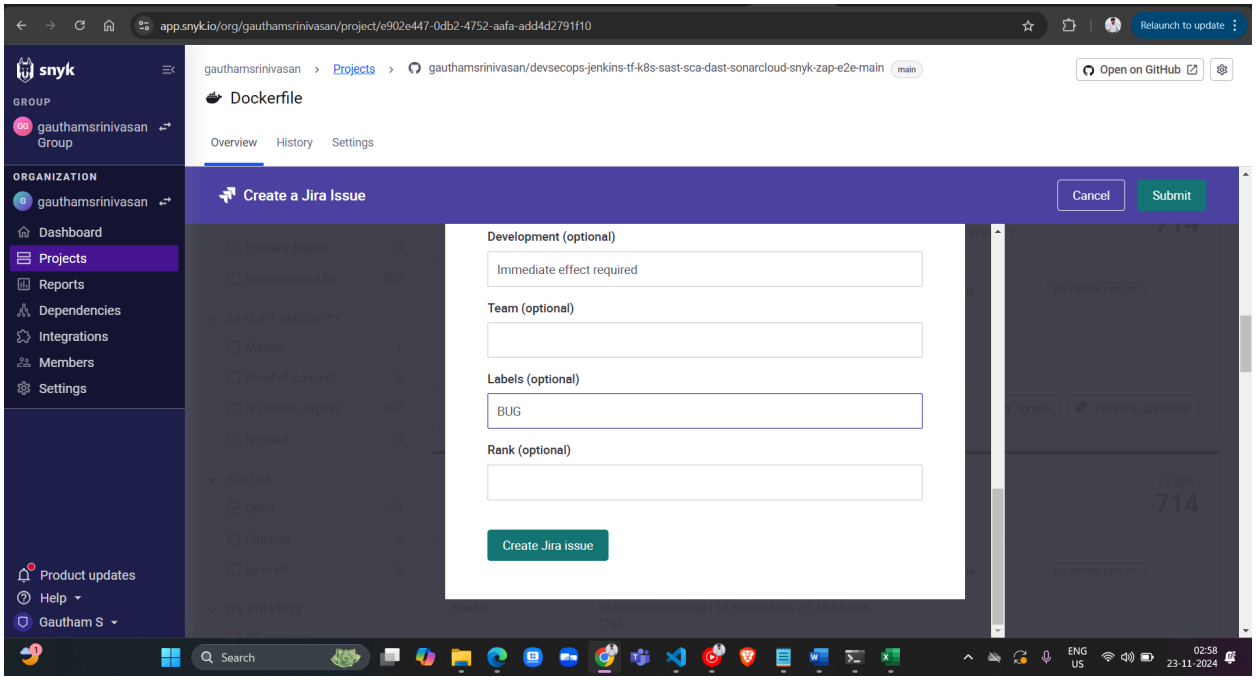
5. Go to your project in snyk and select a vulnerability and a new button **"Create a Jira Issue"** will be enabled



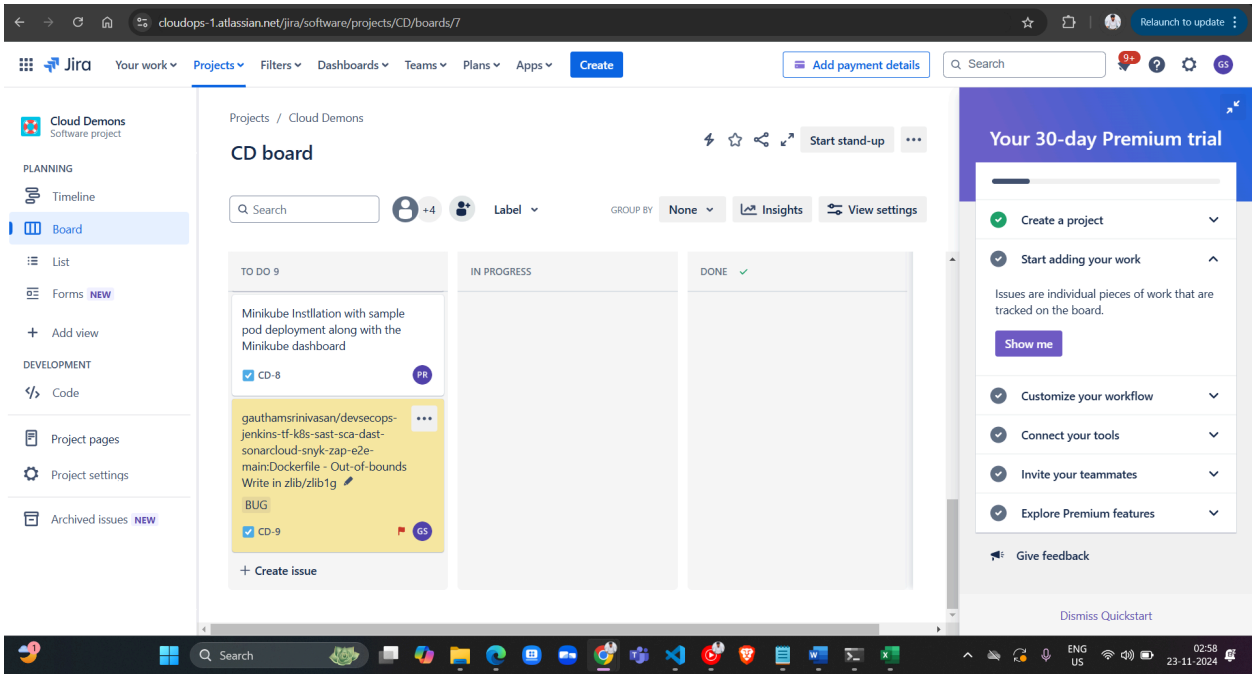
6. Assign the jira issue to your team member and flag the issue type and click create

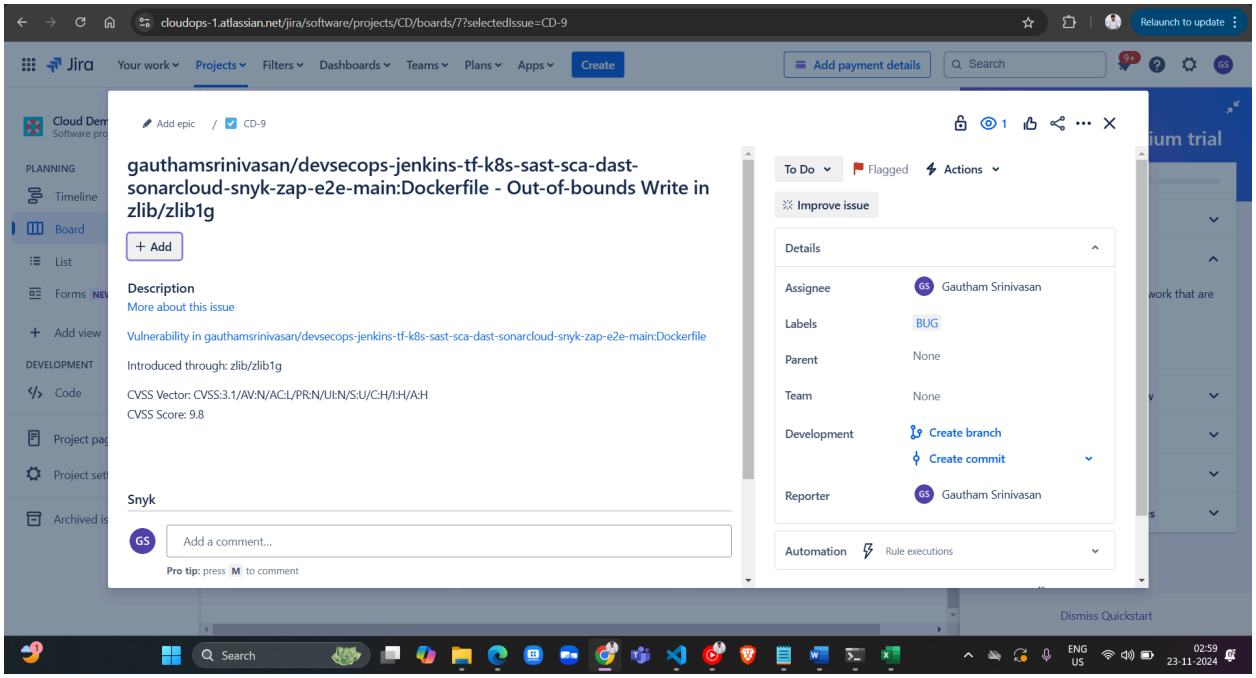

app.snyk.io/org/gauthamsrinivasan/project/e902e447-0db2-4752-aafa-add4d2791f10
Relaunch to update
snyk
GROUP
gauthamsrinivasan Group
ORGANIZATION
gauthamsrinivasan
Dashboard
Projects
Reports
Dependencies
Integrations
Members
Settings
Product updates
Help
Gautham S
gauthamsrinivasan › Projects › gauthamsrinivasan/devsecops-jenkins-tf-k8s-sast-sca-dast-sonarcloud-snyk-zap-e2e-main main
Open on GitHub
Dockerfile
Overview
History
Settings
Create a Jira Issue
Cancel
Submit
Select issue type
Task
Reporter
Gautham Srinivasan
Assignee (optional)
Search users...
Gautham Srinivasan
Nandhini
Priyadharshini R
sanjjushrivarshini
[Vulnerability in gauthamsrinivasan/devsecops-jenkins-tf-k8s-sast-sca-dast-sonarcloud-snyk-zap-e2e-
Search
ENG US
02:57
23-11-2024


app.snyk.io/org/gauthamsrinivasan/project/e902e447-0db2-4752-aafa-add4d2791f10
Relaunch to update
snyk
GROUP
gauthamsrinivasan Group
ORGANIZATION
gauthamsrinivasan
Dashboard
Projects
Reports
Dependencies
Integrations
Members
Settings
Product updates
Help
Gautham S
gauthamsrinivasan › Projects › gauthamsrinivasan/devsecops-jenkins-tf-k8s-sast-sca-dast-sonarcloud-snyk-zap-e2e-main main
Open on GitHub
Dockerfile
Overview
History
Settings
Create a Jira Issue
Cancel
Submit
Summary
gauthamsrinivasan/devsecops-jenkins-tf-k8s-sast-sca-dast-sonarcloud-snyk-
Description (optional)
[More about this issue|https://app.snyk.io/vuln/SNYK-DEBIAN11-ZLIB-2976151]
[Vulnerability in gauthamsrinivasan/devsecops-jenkins-tf-k8s-sast-sca-dast-sonarcloud-snyk-zap-e2e-main:Dockerfile|https://app.snyk.io/org/gauthamsrinivasan/project/e902e447-0db2-4752-aafa-add4d2791f10#issue-SNYK-DEBIAN11-ZLIB-2976151]
Introduced through: zlib/zlib1g
Flagged (optional)
Impediment
Search
ENG US
02:57
23-11-2024

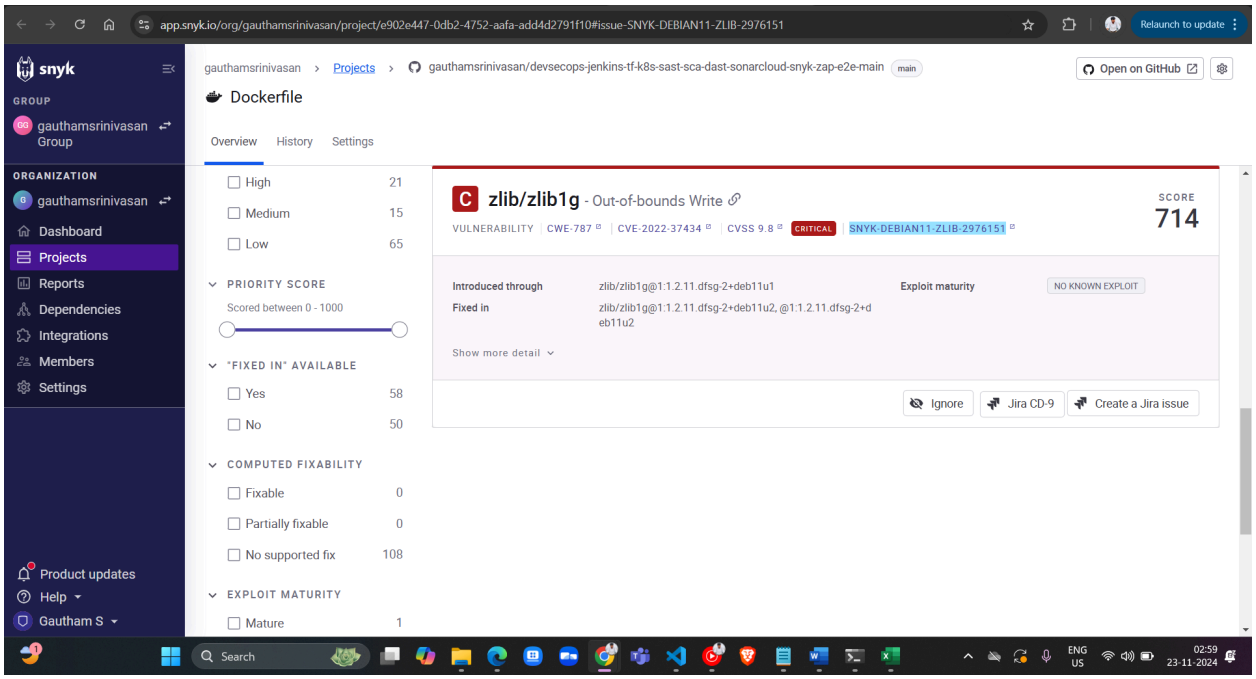


7. Verify the jira issue is assigned in your jira board console



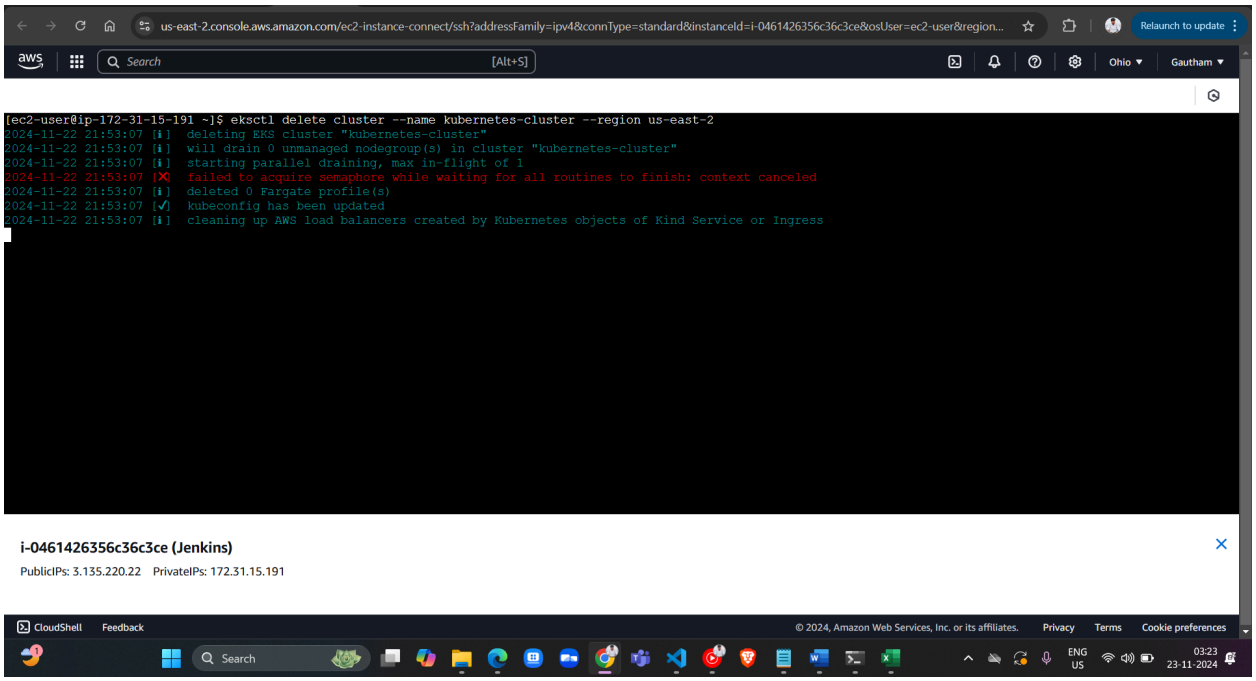


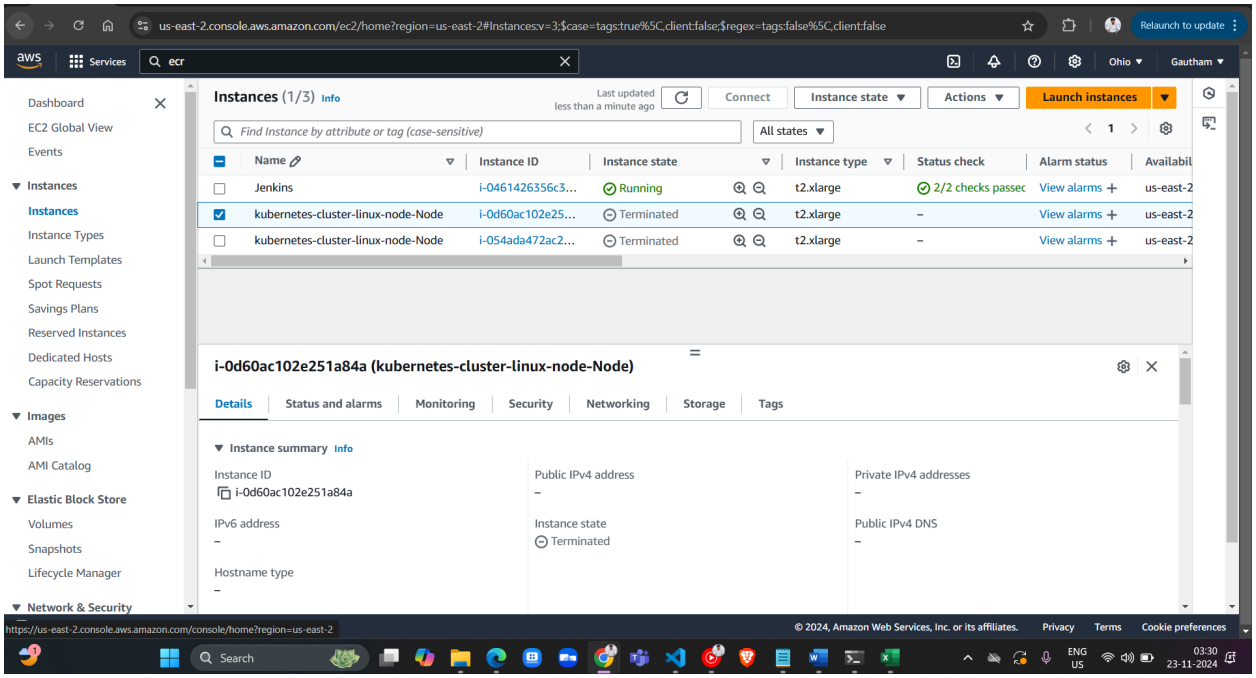
8. Navigate the snyk issue from the jira issued ticket by clicking the embedded link

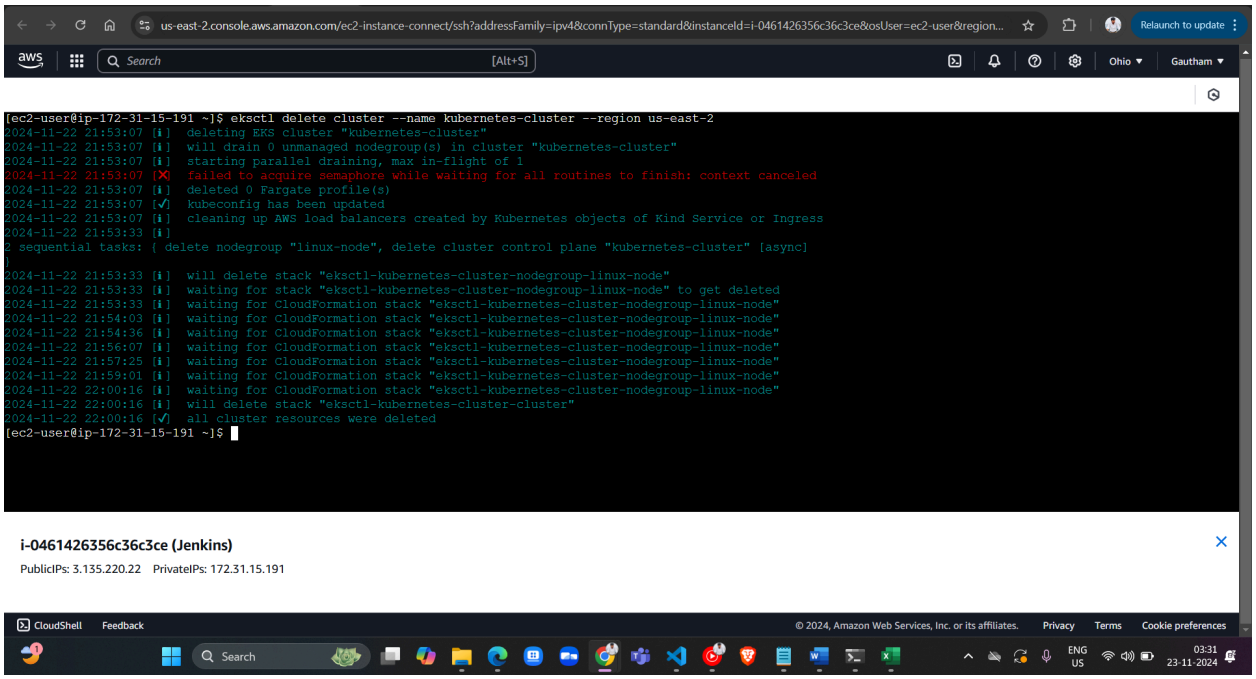


Step 11. Cleanup

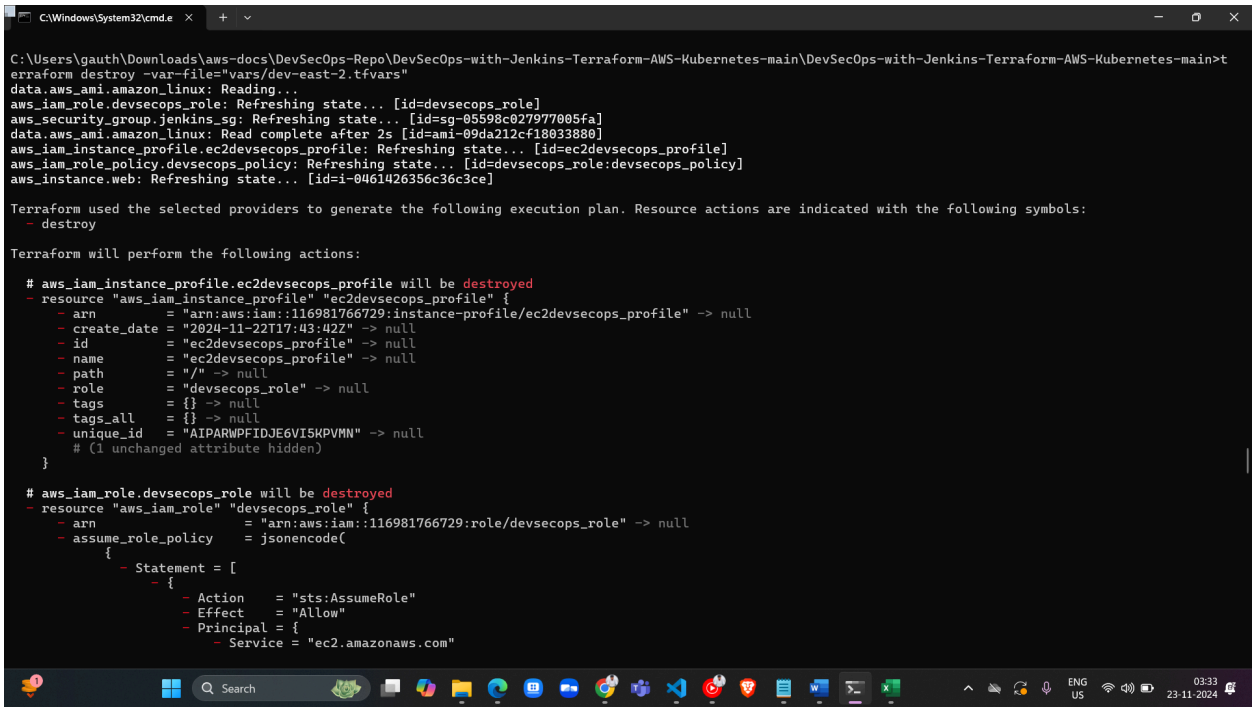
1. Delete the EKS cluster:
   `eksctl delete cluster --name kubernetes-cluster --region us-east-2`

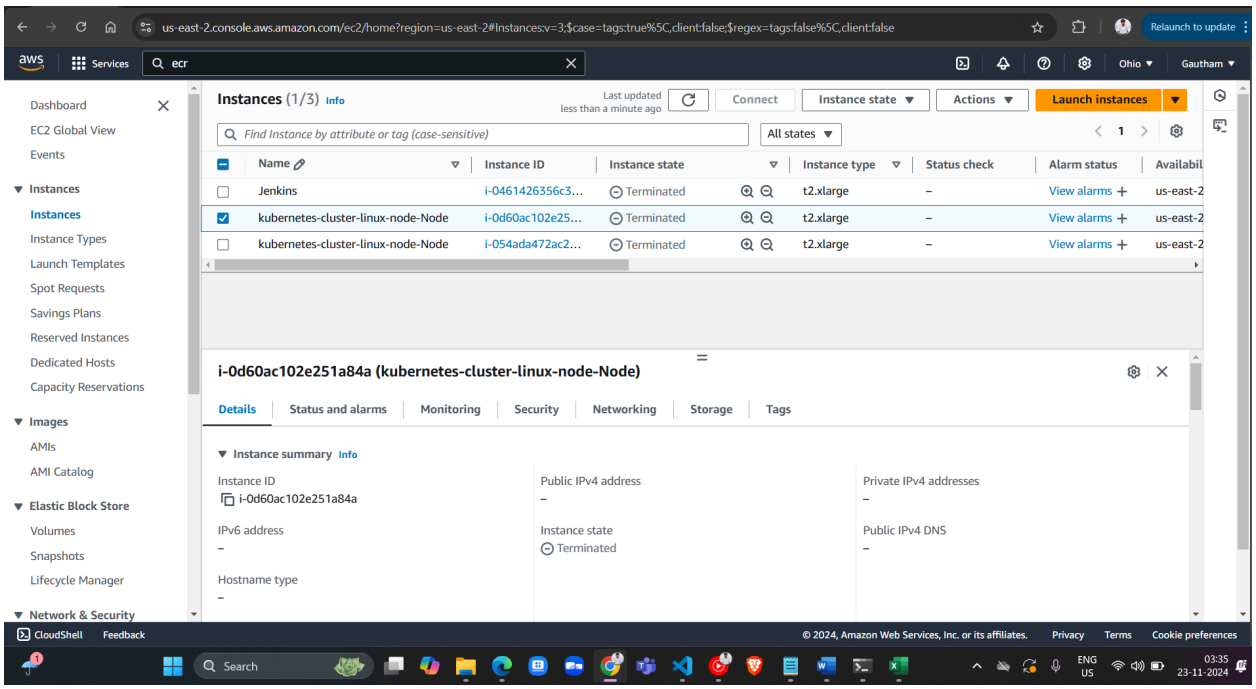

us-east-2.console.aws.amazon.com/ec2-instance-connect/ssh?addressFamily=ipv4&connType=standard&instanceId=i-0461426356c36c3ce&osUser=ec2-user&region...
Relaunch to update
Search [Alt+S]
Ohio
Gautham
[ec2-user@ip-172-31-15-191 ~]$ eksctl delete cluster --name kubernetes-cluster --region us-east-2
2024-11-22 21:53:07 [ℹ] deleting EKS cluster "kubernetes-cluster"
2024-11-22 21:53:07 [ℹ] will drain 0 unmanaged nodegroup(s) in cluster "kubernetes-cluster"
2024-11-22 21:53:07 [ℹ] starting parallel draining, max in-flight of 1
2024-11-22 21:53:07 [✖] failed to acquire semaphore while waiting for all routines to finish: context canceled
2024-11-22 21:53:07 [ℹ] deleted 0 Fargate profile(s)
2024-11-22 21:53:07 [✓] kubeconfig has been updated
2024-11-22 21:53:07 [ℹ] cleaning up AWS load balancers created by Kubernetes objects of Kind Service or Ingress
i-0461426356c36c3ce (Jenkins)
PublicIPs: 3.135.220.22 PrivateIPs: 172.31.15.191
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Search
ENG US
03:23 23-11-2024


us-east-2.console.aws.amazon.com/ec2/home?region=us-east-2#Instances:v=3;$case=tags:true%5C,client:false;$regex=tags:false%5C,client:false
Relaunch to update
Services
ecr
Ohio
Gautham
Dashboard
EC2 Global View
Events
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Instances (1/3) Info
Last updated less than a minute ago
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availabil
Jenkins
i-0461426356c3...
Running
t2.xlarge
2/2 checks passec
View alarms
us-east-2
kubernetes-cluster-linux-node-Node
i-0d60ac102e25...
Terminated
t2.xlarge
View alarms
us-east-2
kubernetes-cluster-linux-node-Node
i-054ada472ac2...
Terminated
t2.xlarge
View alarms
us-east-2
i-0d60ac102e251a84a (kubernetes-cluster-linux-node-Node)
Details
Status and alarms
Monitoring
Security
Networking
Storage
Tags
Instance summary Info
Instance ID
i-0d60ac102e251a84a
Public IPv4 address
Private IPv4 addresses
IPv6 address
Instance state
Terminated
Public IPv4 DNS
Hostname type
https://us-east-2.console.aws.amazon.com/console/home?region=us-east-2
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Search
ENG US
03:30 23-11-2024

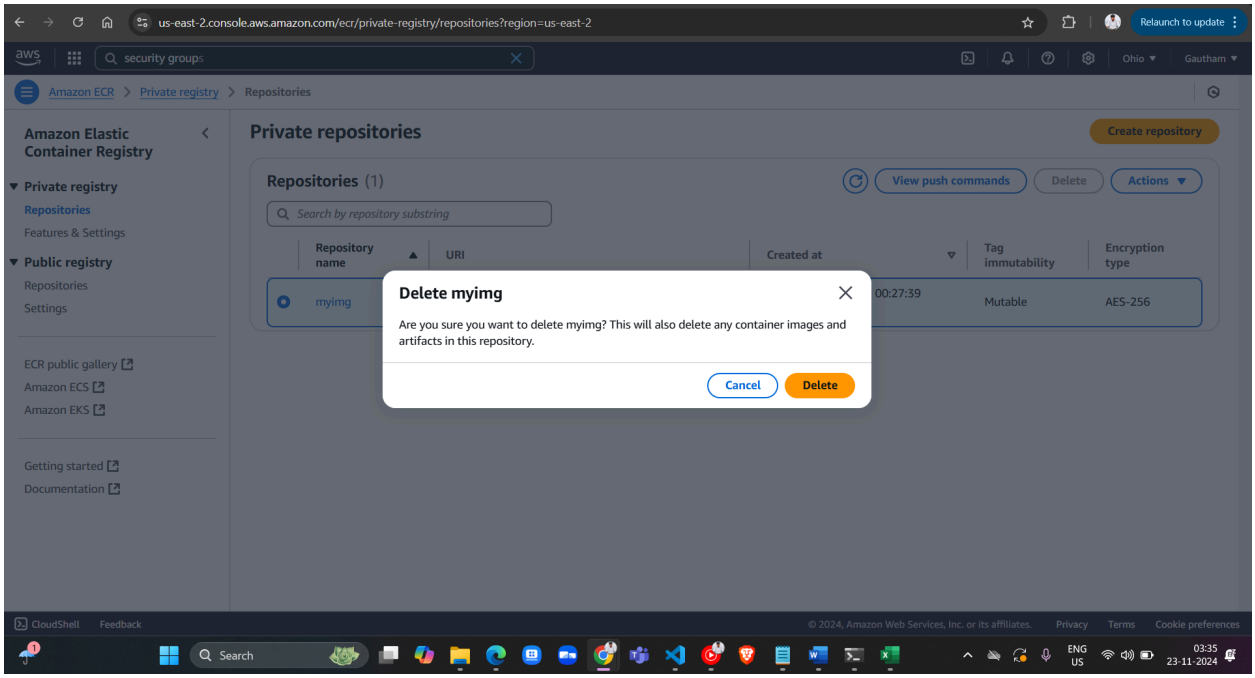


2. Remove other resources as required by using `-var-file="vars/dev-east-2.tfvars"`

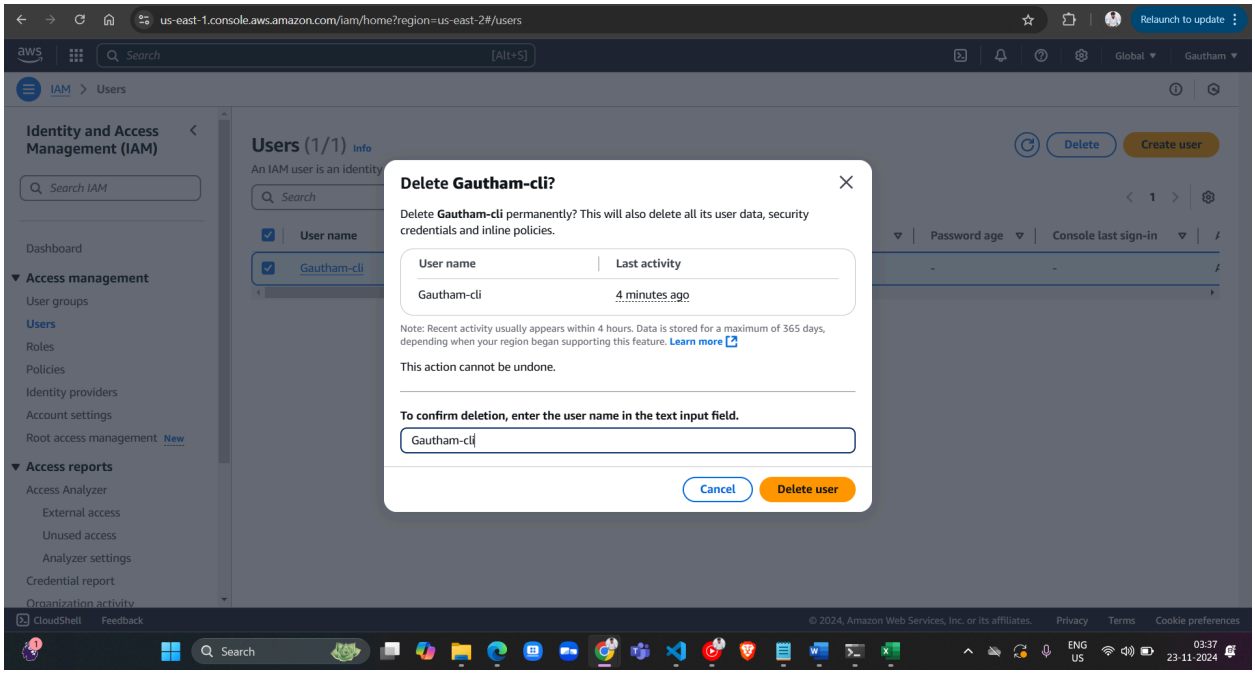




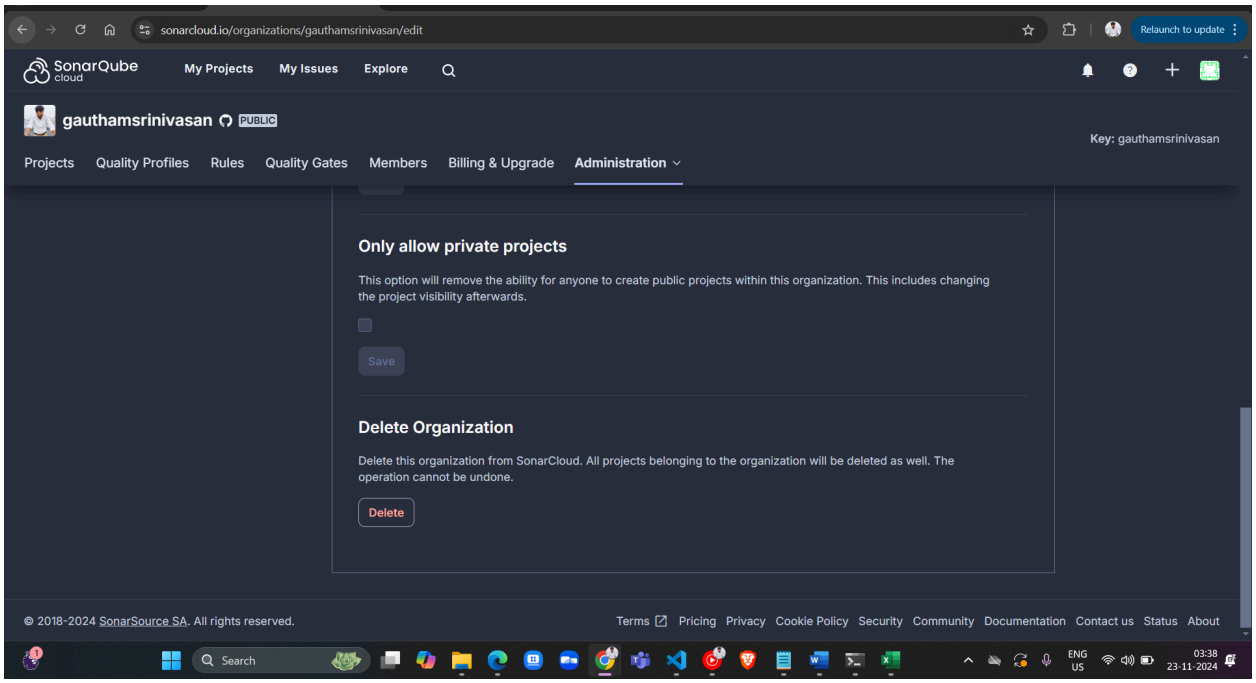
- Delete the Registry



- Delete the created user



- Delete SonarQube organization



Conclusion

This project successfully demonstrates the integration of Jenkins, Terraform, AWS, Kubernetes, SonarCloud, Snyk, OWASP ZAP, and Jira for a comprehensive DevSecOps pipeline. By automating the entire workflow, from infrastructure provisioning to vulnerability scanning and automated issue tracking, we ensure seamless security, scalability, and efficiency in the development lifecycle. The process improves software quality, enhances security posture, and facilitates faster, automated delivery with minimal manual intervention, aligning with best practices in modern DevSecOps.