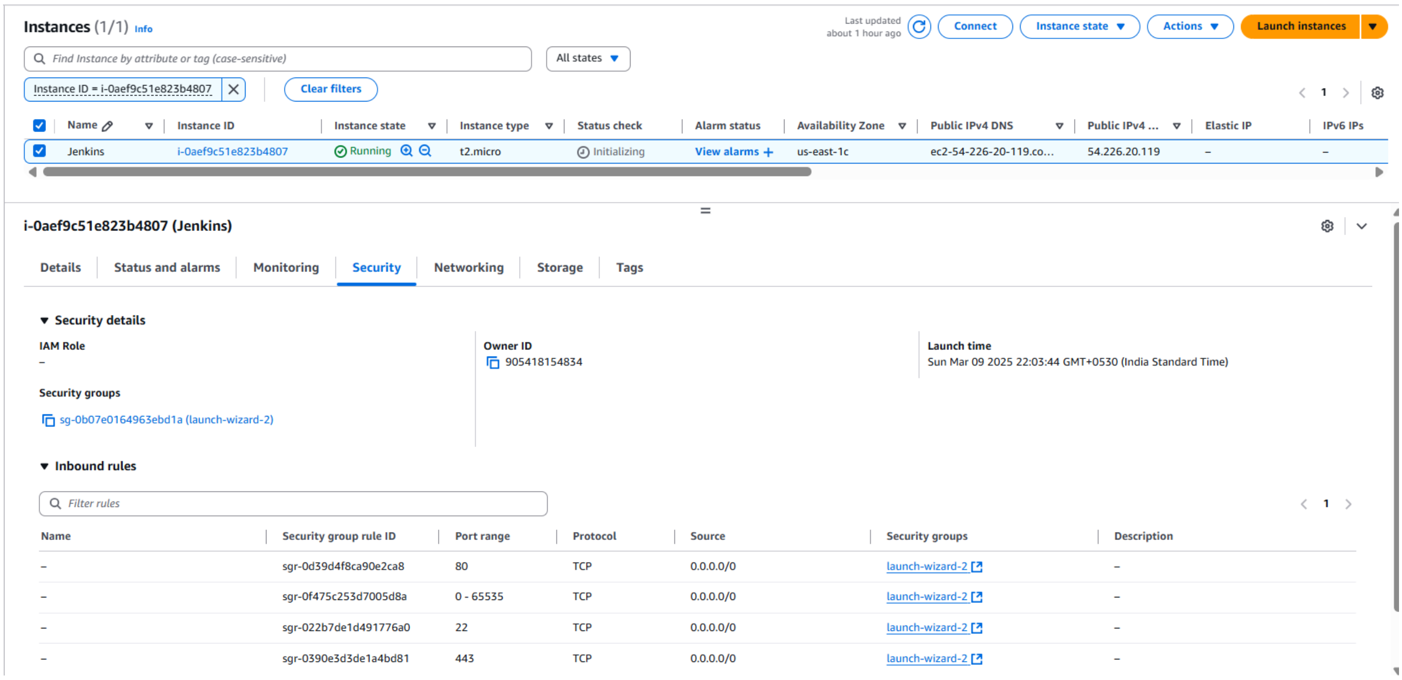
DevOps Assignment – 5 (Day-6)

23951A050H Marreddy Akshaya Varshini

1) Create an EC2 instance & add port 8080 in security groups



Instances (1/1) Info

Last updated about 1 hour ago | Connect | Instance state ▼ | Actions ▼ | Launch instances ▼

Find Instance by attribute or tag (case-sensitive) | All states ▼

Instance ID = i-0aef9c51e823b4807 ✕ | Clear filters

☑	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP	IPv6 IPs
☑	Jenkins	i-0aef9c51e823b4807	Running	t2.micro	Initializing	View alarms +	us-east-1c	ec2-54-226-20-119.co...	54.226.20.119	–	–

i-0aef9c51e823b4807 (Jenkins)

Details | Status and alarms | Monitoring | Security | Networking | Storage | Tags

▼ Security details

IAM Role: –

Owner ID: 905418154834

Launch time: Sun Mar 09 2025 22:03:44 GMT+0530 (India Standard Time)

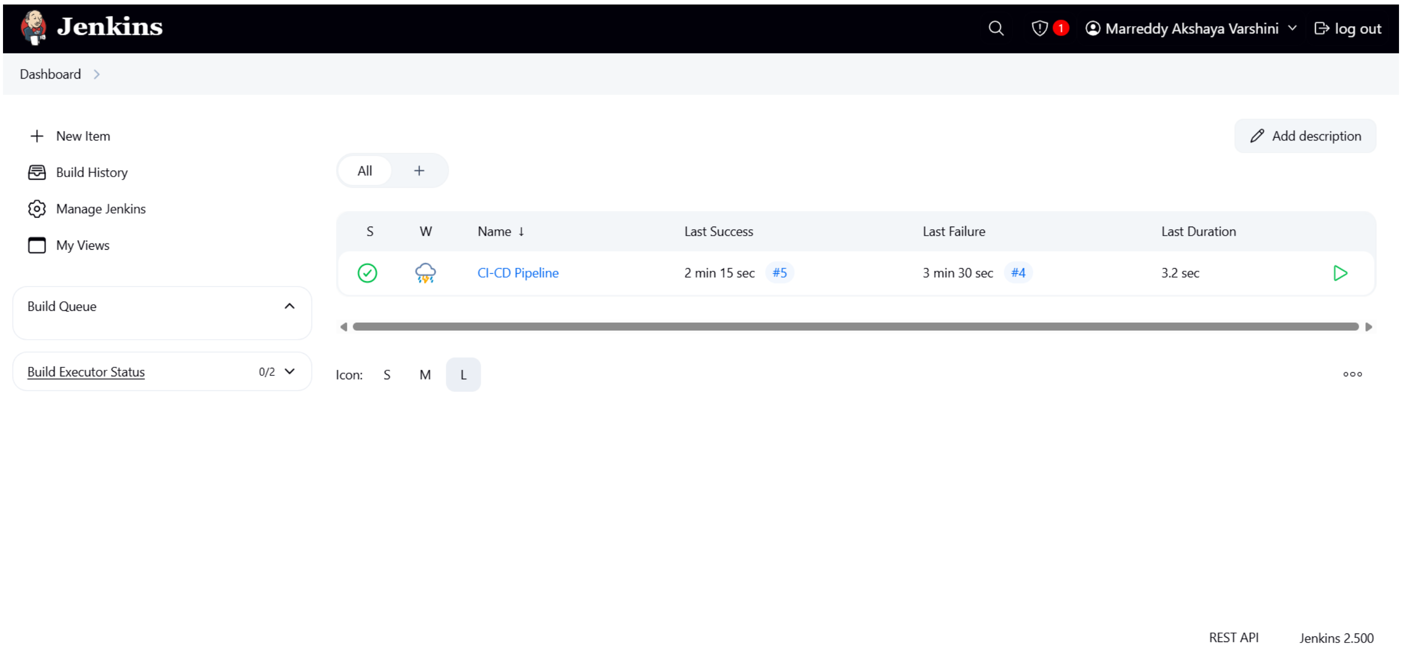
Security groups: sg-0b07e0164963ebd1a (launch-wizard-2)

▼ Inbound rules

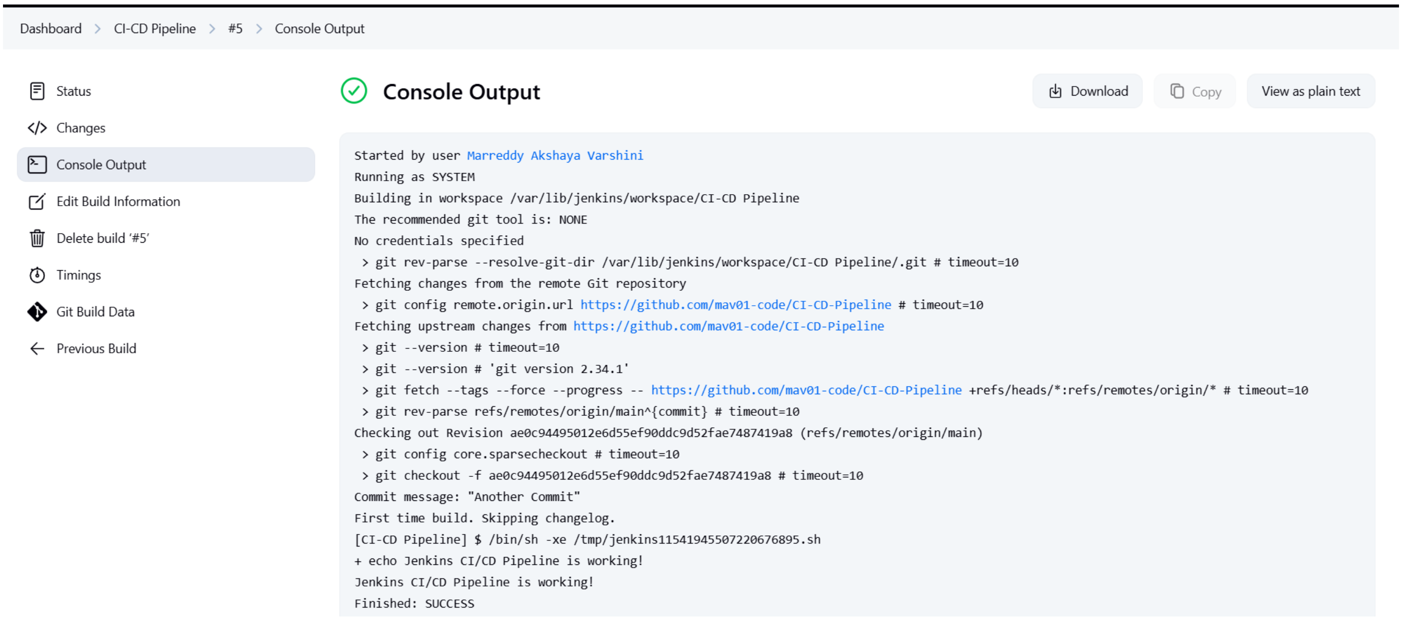
Filter rules

Name	Security group rule ID	Port range	Protocol	Source	Security groups	Description
–	sgr-0d39d4f8ca90e2ca8	80	TCP	0.0.0.0/0	launch-wizard-2	–
–	sgr-0f475c253d7005d8a	0 - 65535	TCP	0.0.0.0/0	launch-wizard-2	–
–	sgr-022b7de1d491776a0	22	TCP	0.0.0.0/0	launch-wizard-2	–
–	sgr-0390e3d3de1a4bd81	443	TCP	0.0.0.0/0	launch-wizard-2	–

2) Open your VM and update
3) Install jdk and Jenkins
   - sudo apt install fontconfig openjdk-17-jre
   - sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \ https://pkg.jenkins.io/debian/jenkins.io-2023.key
   - echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc]" \ https://pkg.jenkins.io/debian binary/ | sudo tee \ /etc/apt/sources.list.d/jenkins.list > /dev/null
   - sudo apt-get update
   - sudo apt-get install Jenkins
4) Open Jenkins on your browser using IPv4 address:8080
   - http://54.226.20.119:8080/
5) Unlock Jenkins and login with your initial password
   - sudo cat /var/lib/jenkins/secrets/initialAdminPassword
6) Install required plugins
7) Create a git repository and make sure to create a personal access token
8) Create new item on Jenkins
9) Add a name and select freestyle project
10) Add your github repo link in the source code management
11) Specify branch – master/main
12) Save it and click on build now
13) Add a webhook on your github repo
14) Whenever there's a commit, codebuild will automatically be triggered



S	W	Name ↓	Last Success	Last Failure	Last Duration
✓		CI-CD Pipeline	2 min 15 sec #5	3 min 30 sec #4	3.2 sec



Console Output

```
Started by user Marreddy Akshaya Varshini
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/CI-CD Pipeline
The recommended git tool is: NONE
No credentials specified
 > git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/CI-CD Pipeline/.git # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/mav01-code/CI-CD-Pipeline # timeout=10
Fetching upstream changes from https://github.com/mav01-code/CI-CD-Pipeline
 > git --version # timeout=10
 > git --version # 'git version 2.34.1'
 > git fetch --tags --force --progress -- https://github.com/mav01-code/CI-CD-Pipeline +refs/heads/*:refs/remotes/origin/* # timeout=10
 > git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision ae0c94495012e6d55ef90ddc9d52fae7487419a8 (refs/remotes/origin/main)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f ae0c94495012e6d55ef90ddc9d52fae7487419a8 # timeout=10
Commit message: "Another Commit"
First time build. Skipping changelog.
[CI-CD Pipeline] $ /bin/sh -xe /tmp/jenkins11541945507220676895.sh
+ echo Jenkins CI/CD Pipeline is working!
Jenkins CI/CD Pipeline is working!
Finished: SUCCESS
```