



AWS Multi-Region Setup with Git and Jenkins Pipeline



NAME : T MANOJ
EMPID : LYAKE2KHS

Introduction to Multi-Region AWS GIT-JENKINS Setup

Overview

This documentation details a comprehensive continuous integration and continuous deployment (CI/CD) pipeline implemented across two AWS regions. The setup leverages multiple AWS services and open-source tools to create an efficient development workflow that automates code building, testing, and deployment processes.

Services and Tools Used

Amazon Web Services (AWS)

- **EC2 Instances:** Virtual servers in the cloud used to host both the code repository and CI/CD server.
- **Security Groups:** Virtual firewalls that control inbound and outbound traffic to our instances.
- **Multi-Region Architecture:** Utilizing instances in both Ohio and North Virginia regions for geographic distribution and resilience.

Operating Systems

- **Amazon Linux:** A Linux distribution provided by AWS, optimized for the EC2 environment. Used in the Ohio region for the Git repository.
- **Ubuntu:** A popular Debian-based Linux distribution used in the North Virginia region for running Jenkins.

Version Control

- **Git:** Distributed version control system used to track changes in source code during software development.
- **GitHub:** Cloud-based hosting service for Git repositories, facilitating collaboration and code sharing.

CI/CD Tools

- **Jenkins:** An open-source automation server that enables developers to build, test, and deploy their applications.
- **Webhooks:** Used to trigger Jenkins jobs automatically when code is pushed to the GitHub repository.

File Transfer Tools

- **WinSCP:** A free and open-source SFTP, FTP, WebDAV, and SCP client used to securely transfer files between local and remote systems.

System Architecture

The architecture consists of two main components:

1. Source Code Repository (Ohio Region)

- Amazon Linux EC2 instance
- Git server hosting the application code
- Acts as the origin for code pushes to GitHub

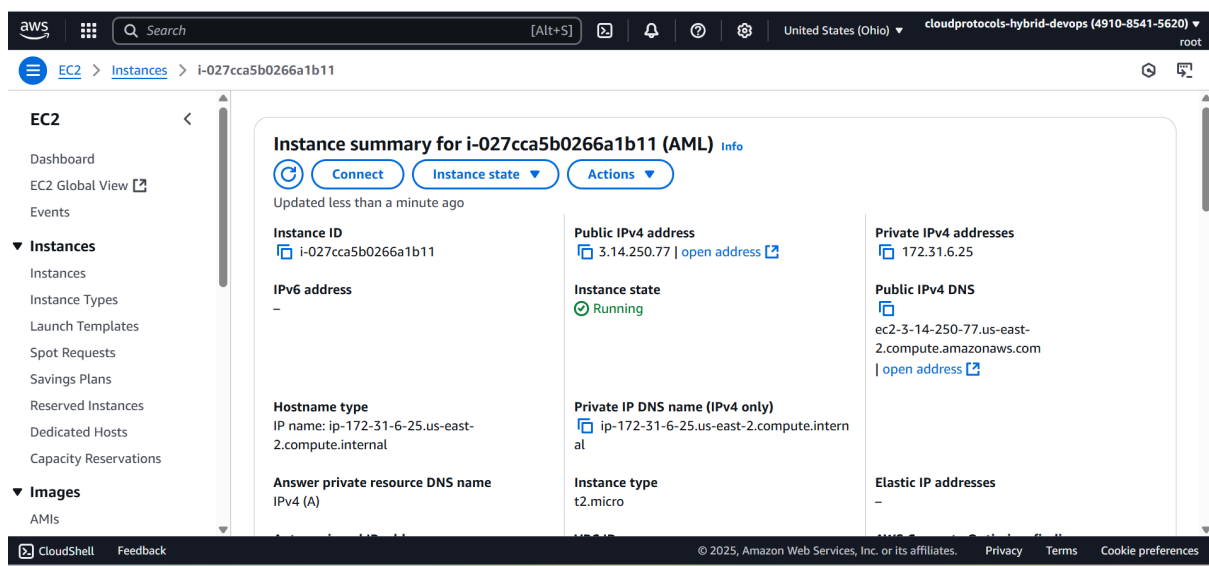
2. CI/CD Server (North Virginia Region)

- Ubuntu EC2 instance
- Jenkins server configured with two jobs:
 - GitBuilder: Triggered by GitHub webhooks to build and test code
 - GitDeploy: Executes after successful GitBuilder job to deploy the application

Infrastructure Overview

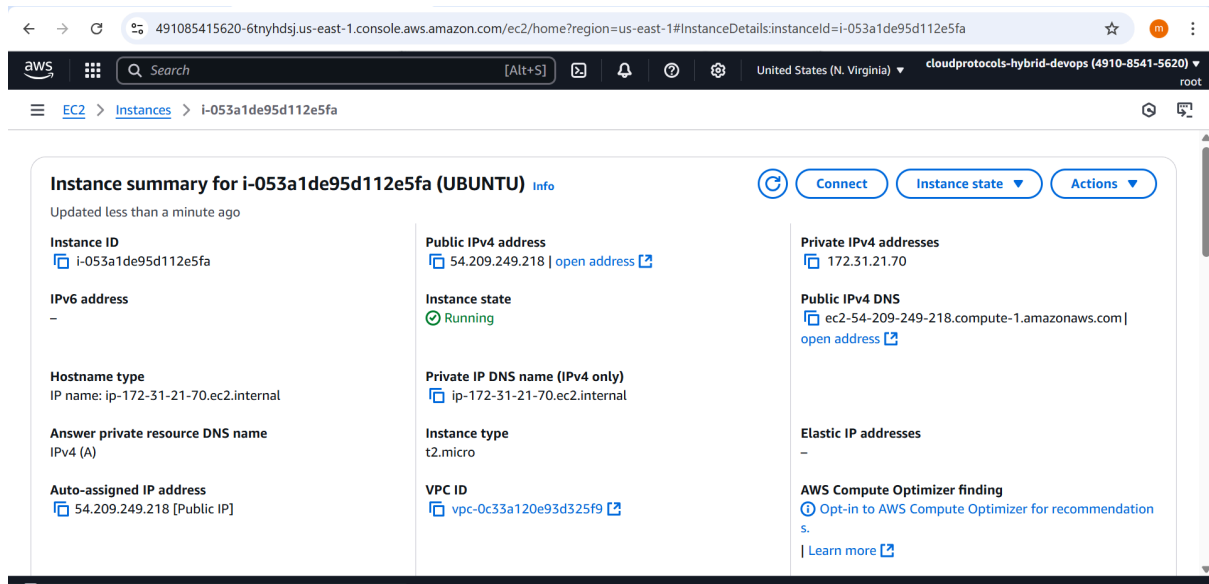
1. Region 1: Ohio

- Operating System: Amazon Linux
- Purpose: Source code repository (Git server)



2. Region 2: North Virginia

- Operating System: Ubuntu
- Purpose: CI/CD server (Jenkins)



Amazon Linux Setup (Ohio Region)

1. Git Installation and Repository Setup

```
# Install Git
sudo yum update -y
sudo yum install git -y

# Verify Git installation
git --version
```

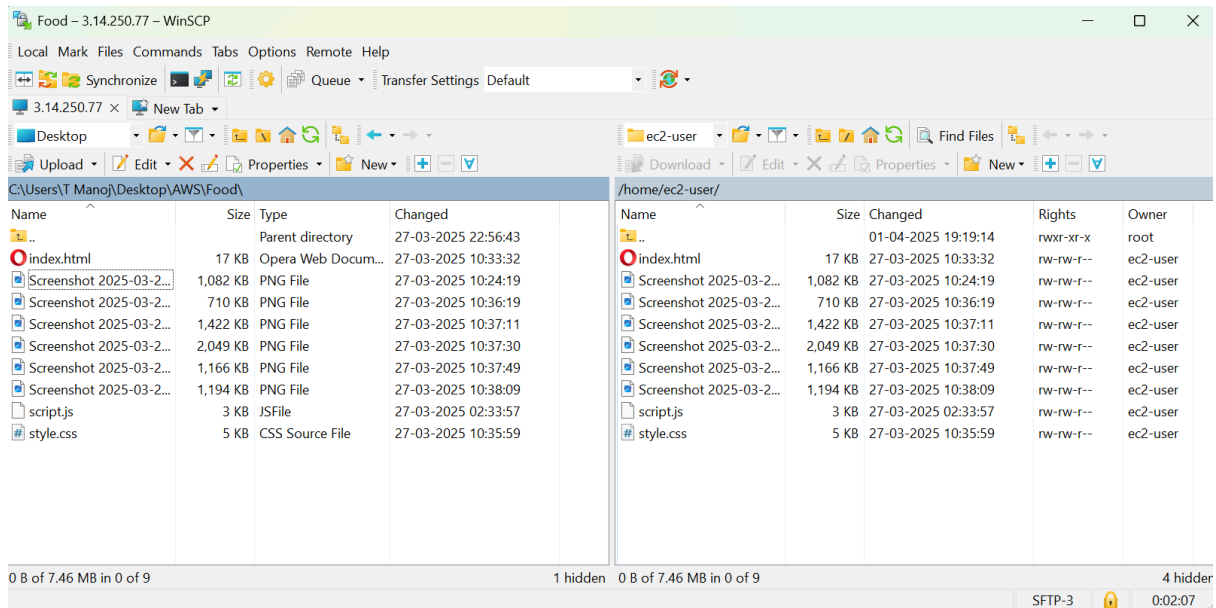
```
Installed:
git-2.47.1-1.amzn2023.0.2.x86_64
git-core-2.47.1-1.amzn2023.0.2.x86_64
git-core-doc-2.47.1-1.amzn2023.0.2.noarch
perl-Error-1:0.17029-5.amzn2023.0.2.noarch
perl-File-Find-1.37-477.amzn2023.0.6.noarch
perl-Git-2.47.1-1.amzn2023.0.2.noarch
perl-TermReadKey-2.38-9.amzn2023.0.2.x86_64
perl-lib-0.65-477.amzn2023.0.6.x86_64

Complete!
[ec2-user@ip-172-31-6-25 ~]$
```

2. File Transfer from Local Machine

- Used WinSCP to copy files from local system to Amazon Linux instance

- Transferred project files to `/path/to/project` directory



3. Git Repository Creation

```
# Navigate to project directory
cd /path/to/project
```

```
[ec2-user@ip-172-31-6-25 ~]$ ll
total 7668
-rw-rw-r--. 1 ec2-user ec2-user 1107363 Mar 27 04:54 'Screenshot 2025-03-27 1024
18.png'
-rw-rw-r--. 1 ec2-user ec2-user 726279 Mar 27 05:06 'Screenshot 2025-03-27 1036
18.png'
-rw-rw-r--. 1 ec2-user ec2-user 1456104 Mar 27 05:07 'Screenshot 2025-03-27 1037
09.png'
-rw-rw-r--. 1 ec2-user ec2-user 2097785 Mar 27 05:07 'Screenshot 2025-03-27 1037
27.png'
-rw-rw-r--. 1 ec2-user ec2-user 1193239 Mar 27 05:07 'Screenshot 2025-03-27 1037
47.png'
-rw-rw-r--. 1 ec2-user ec2-user 1221757 Mar 27 05:08 'Screenshot 2025-03-27 1038
07.png'
-rw-rw-r--. 1 ec2-user ec2-user 16915 Mar 27 05:03 index.html
-rw-rw-r--. 1 ec2-user ec2-user 2695 Mar 26 21:03 script.js
-rw-rw-r--. 1 ec2-user ec2-user 4486 Mar 27 05:05 style.css
[ec2-user@ip-172-31-6-25 ~]$
```

```
# Initialize Git repository
git init
```

```

[ec2-user@ip-172-31-6-25 ~]$ ll
total 7668
-rw-rw-r--. 1 ec2-user ec2-user 1107363 Mar 27 04:54 'Screenshot 2025-03-27 1024
18.png'
-rw-rw-r--. 1 ec2-user ec2-user 726279 Mar 27 05:06 'Screenshot 2025-03-27 1036
18.png'
-rw-rw-r--. 1 ec2-user ec2-user 1456104 Mar 27 05:07 'Screenshot 2025-03-27 1037
09.png'
-rw-rw-r--. 1 ec2-user ec2-user 2097785 Mar 27 05:07 'Screenshot 2025-03-27 1037
27.png'
-rw-rw-r--. 1 ec2-user ec2-user 1193239 Mar 27 05:07 'Screenshot 2025-03-27 1037
47.png'
-rw-rw-r--. 1 ec2-user ec2-user 1221757 Mar 27 05:08 'Screenshot 2025-03-27 1038
07.png'
-rw-rw-r--. 1 ec2-user ec2-user 16915 Mar 27 05:03 index.html
-rw-rw-r--. 1 ec2-user ec2-user 2695 Mar 26 21:03 script.js
-rw-rw-r--. 1 ec2-user ec2-user 4486 Mar 27 05:05 style.css
[ec2-user@ip-172-31-6-25 ~]$ git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint: git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint: git branch -m <name>
Initialized empty Git repository in /home/ec2-user/.git/

```

```

# Add all files to Git
git add .

```

```

# Commit changes
git commit -m "Initial commit"

```

```

[ec2-user@ip-172-31-6-25 ~]$ git add .
[ec2-user@ip-172-31-6-25 ~]$ git commit -m "initial commit"
[master (root-commit) 914a9b9] initial commit
Committer: EC2 Default User <ec2-user@ip-172-31-6-25.us-east-2.compute.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. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

13 files changed, 371 insertions(+)
create mode 100644 .bash_logout
create mode 100644 .bash_profile
create mode 100644 .bashrc
create mode 100644 .ssh/authorized_keys
create mode 100644 Screenshot 2025-03-27 102418.png
create mode 100644 Screenshot 2025-03-27 103618.png
create mode 100644 Screenshot 2025-03-27 103709.png
create mode 100644 Screenshot 2025-03-27 103727.png
create mode 100644 Screenshot 2025-03-27 103747.png
create mode 100644 Screenshot 2025-03-27 103807.png
create mode 100644 index.html
create mode 100644 script.js
create mode 100644 style.css
[ec2-user@ip-172-31-6-25 ~]$

```

```

# Configure remote repository (GitHub)
git remote add origin https://github.com/username/repository.git

```

```
ec2-user@ip-172-31-6-25 ~]$ git remote add origin https://github.com/tupakulamanoj/JENKINS.git
ec2-user@ip-172-31-6-25 ~]$ git remote -v
origin  https://github.com/tupakulamanoj/JENKINS.git (fetch)
origin  https://github.com/tupakulamanoj/JENKINS.git (push)
ec2-user@ip-172-31-6-25 ~]$
```

```
# Push code to remote repository
git push -u origin main
```

Ubuntu Setup (North Virginia Region)

1. Java Installation (Prerequisite for Jenkins)

```
# Update package index
sudo apt update

# Install OpenJDK 17
sudo apt install openjdk-17-jdk -y

# Verify Java installation
java -version
```

```
ubuntu@ip-172-31-21-70:~$ sudo apt update
sudo apt install openjdk-17-jdk -y
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Ign:5 https://pkg.jenkins.io/debian-stable binary/ InRelease
Hit:6 https://pkg.jenkins.io/debian-stable binary/ Release
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [979 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Components [151 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [1046 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Components [365 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Components [212 B]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Components [940 B]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 Components [7060 B]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Components [15.7 kB]
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/restricted amd64 Components [212 B]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 Components [212 B]
Get:18 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [731 kB]
Get:19 http://security.ubuntu.com/ubuntu noble-security/main amd64 Components [8996 B]
Get:20 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Packages [827 kB]
Get:21 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Components [52.0 kB]
Get:22 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Components [212 B]
Get:23 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Components [212 B]
Fetched 4563 kB in 2s (2708 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
34 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
openjdk-17-jdk is already the newest version (17.0.14+7-1~24.04).
0 upgraded, 0 newly installed, 0 to remove and 34 not upgraded.
ubuntu@ip-172-31-21-70:~$
```

2. Jenkins Installation

```
# Import Jenkins repository key
sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \
https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key
```

```
ubuntu@ip-172-31-21-70:~$ sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \
https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key
--2025-04-01 13:53:30-- https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key
Resolving pkg.jenkins.io (pkg.jenkins.io)... 146.75.38.133, 2a04:4e42:78::645
Connecting to pkg.jenkins.io (pkg.jenkins.io)|146.75.38.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3175 (3.1K) [application/pgp-keys]
Saving to: '/usr/share/keyrings/jenkins-keyring.asc'

/usr/share/keyrings/jenkins-keyring.asc 100%[=====>] 3.10K --.-KB/s in 0s

2025-04-01 13:53:30 (62.4 MB/s) - '/usr/share/keyrings/jenkins-keyring.asc' saved [3175/3175]

ubuntu@ip-172-31-21-70:~$
```

```
# Add Jenkins repository to sources list
echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \
https://pkg.jenkins.io/debian-stable binary/ | sudo tee \
/etc/apt/sources.list.d/jenkins.list > /dev/null
```

```
ubuntu@ip-172-31-21-70: ~
ubuntu@ip-172-31-21-70:~$ echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \
https://pkg.jenkins.io/debian-stable binary/ | sudo tee \
/etc/apt/sources.list.d/jenkins.list > /dev/null
ubuntu@ip-172-31-21-70:~$
```

```
# Update package index
sudo apt-get update
```

```
ubuntu@ip-172-31-21-70:~$ sudo apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Ign:5 https://pkg.jenkins.io/debian-stable binary/ InRelease
Hit:6 https://pkg.jenkins.io/debian-stable binary/ Release
Reading package lists... Done
```

```
# Install Jenkins
sudo apt-get install jenkins -y
```

```
ubuntu@ip-172-31-21-70:~$ sudo apt-get install jenkins
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
jenkins is already the newest version (2.492.2).
0 upgraded, 0 newly installed, 0 to remove and 34 not upgraded.
ubuntu@ip-172-31-21-70:~$
```

```
# Check Jenkins status
sudo systemctl status jenkins
```



```

jenkins.service - Jenkins Continuous Integration Server
   Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; preset: enabled)
   Active: active (running) since Tue 2025-04-01 06:49:49 UTC; 7h ago
     Main PID: 4854 (java)
       Tasks: 38 (limit: 1129)
      Memory: 435.3M (peak: 473.2M)
         CPU: 1min 32.405s
    CGroup: /system.slice/jenkins.service
            └─4854 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkins/war --httpPort=8080

Apr 01 06:51:39 ip-172-31-21-70 jenkins[4854]: 2025-04-01 06:51:39.756+0000 [id=263] INFO jenkins.InitReactorRunner$1#onAttained: Started all
Apr 01 06:51:39 ip-172-31-21-70 jenkins[4854]: 2025-04-01 06:51:39.761+0000 [id=263] INFO jenkins.InitReactorRunner$1#onAttained: Augmented all
Apr 01 06:51:40 ip-172-31-21-70 jenkins[4854]: 2025-04-01 06:51:40.225+0000 [id=263] INFO jenkins.InitReactorRunner$1#onAttained: System confi
Apr 01 06:51:40 ip-172-31-21-70 jenkins[4854]: 2025-04-01 06:51:40.225+0000 [id=263] INFO jenkins.InitReactorRunner$1#onAttained: System confi
Apr 01 06:51:40 ip-172-31-21-70 jenkins[4854]: 2025-04-01 06:51:40.226+0000 [id=263] INFO jenkins.InitReactorRunner$1#onAttained: Loaded all
Apr 01 06:51:40 ip-172-31-21-70 jenkins[4854]: 2025-04-01 06:51:40.249+0000 [id=263] INFO jenkins.InitReactorRunner$1#onAttained: Configuratio
Apr 01 06:51:40 ip-172-31-21-70 jenkins[4854]: 2025-04-01 06:51:40.485+0000 [id=263] INFO jenkins.InitReactorRunner$1#onAttained: Completed inst
Apr 01 06:51:40 ip-172-31-21-70 jenkins[4854]: 2025-04-01 06:51:40.489+0000 [id=82] INFO h.m.UpdateCenter$CompleteBatchJob#run: Completed inst
Apr 01 06:53:57 ip-172-31-21-70 jenkins[4854]: 2025-04-01 06:53:57.349+0000 [id=15] WARNING hudson.security.csrf.CrumbFilter#doFilter: Found is
Apr 01 06:53:57 ip-172-31-21-70 jenkins[4854]: 2025-04-01 06:53:57.350+0000 [id=15] WARNING hudson.security.csrf.CrumbFilter#doFilter: No val
lines 1-20/20 (END)

```

3. Jenkins Initial Configuration

Get Jenkins initial admin password

```
sudo cat /var/lib/jenkins/secrets/initialAdminPassword
```

- Open Jenkins in web browser: <http://<Ubuntu-Instance-IP>:8080>
- Enter the initial admin password
- Install suggested plugins
- Create admin user and complete initial setup

Jenkins Jobs Configuration

1. GitBuilder Job Setup

- From Jenkins dashboard, click "New Item"
- Enter "GitBuilder" as job name and select "Freestyle project"

Dashboard > build > Configuration

Configure

- General
- Source Code Management
- Triggers
- Environment
- Build Steps
- Post-build Actions

Description

EVENT TRIGGER FOR GIT

Plain text Preview

☐ Discard old builds ?

☐ GitHub project

☐ This project is parameterised ?

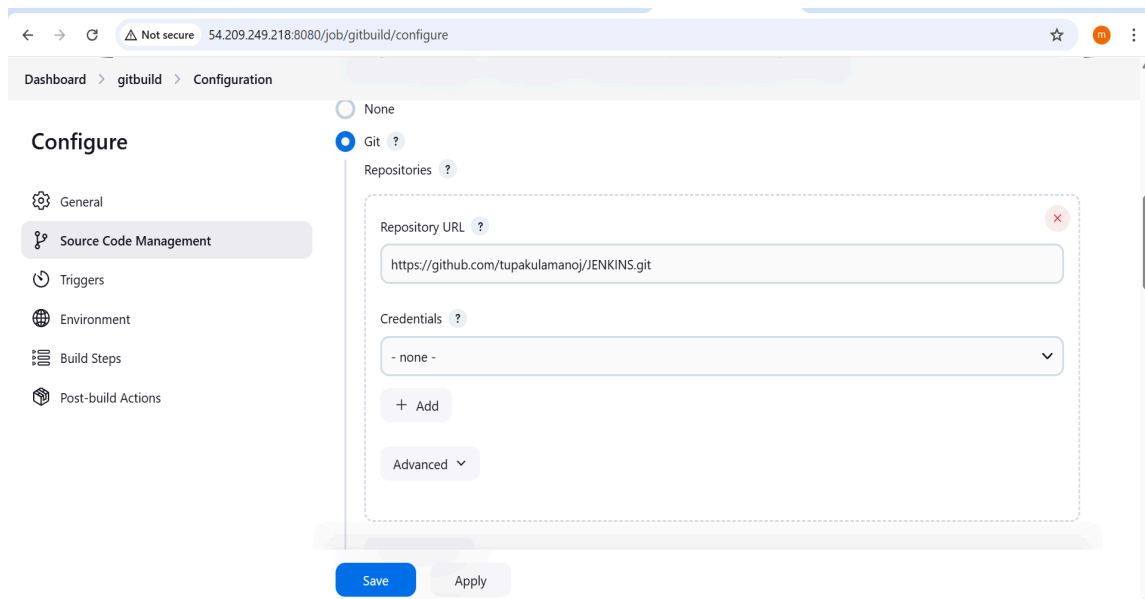
☐ Throttle builds ?

☐ Execute concurrent builds if necessary ?

Advanced ▾

Save Apply

- Configure source code management:
 - Select Git
 - Enter repository URL: <https://github.com/username/repository.git>



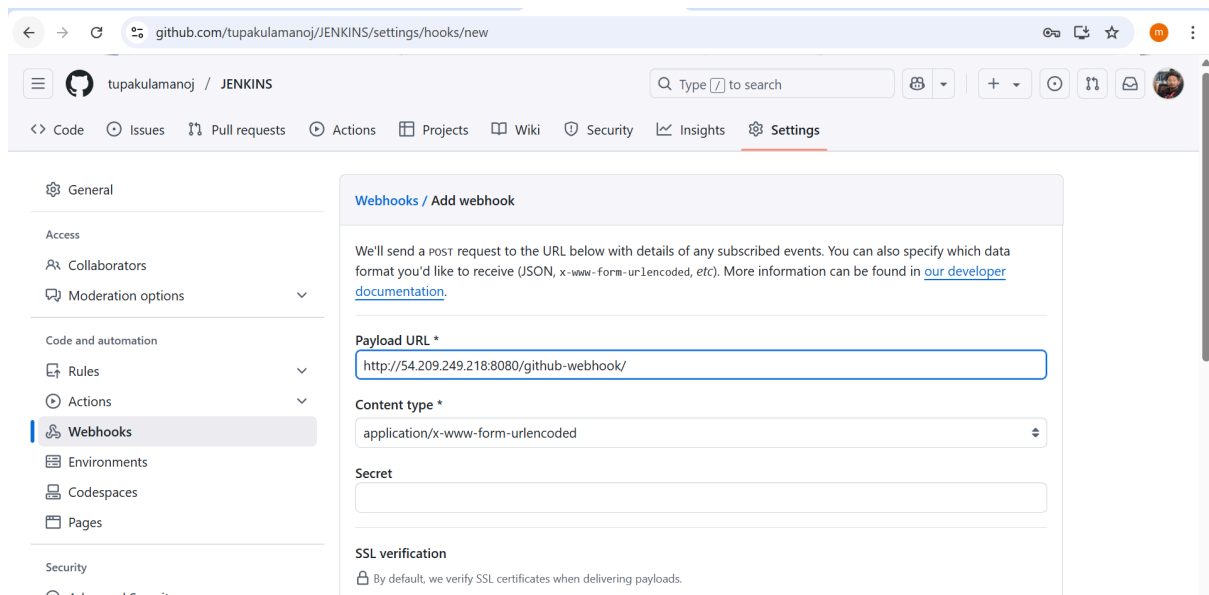
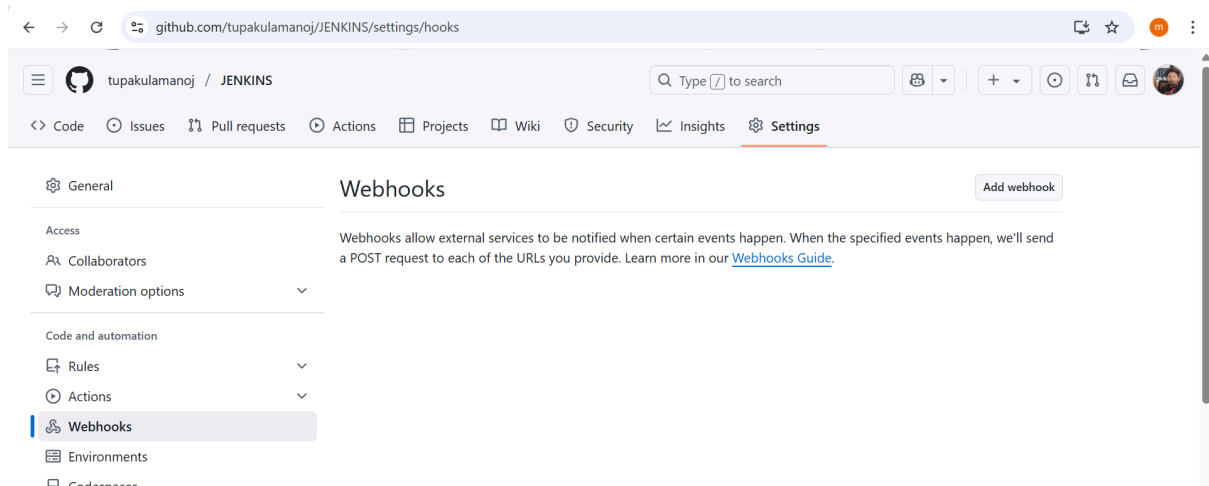
- Configure build triggers:
 - Check "GitHub hook trigger for GITScm polling"

Triggers

Set up automated actions that start your build based on specific events, like code changes or scheduled times.

- ☐ Trigger builds remotely (e.g., from scripts) ?
- ☐ Build after other projects are built ?
- ☐ Build periodically ?
- ☒ GitHub hook trigger for GITScm polling ?
- ☐ Poll SCM ?

- Configure webhook in GitHub:
 - Go to GitHub repository → Settings → Webhooks → Add webhook
 - Payload URL:
`http://<Ubuntu-Instance-IP>:8080/github-webhook/`
 - Content type: `application/json`
 - Events: Select "Just the push event"
 - Save webhook



2. GitDeploy Job Setup

- From Jenkins dashboard, click "New Item"
- Enter "GitDeploy" as job name and select "Freestyle project"
- Configure build triggers:
 - Check "Build after other projects are built"
 - Enter "GitBuilder" in the project name field

Add deployment build steps (e.g., shell commands)

Dashboard > gitdeploy > Configuration

Configure

Set up automated actions that start your build based on specific events, like code changes or scheduled times.

- ☐ Trigger builds remotely (e.g., from scripts) ?
- ☒ Build after other projects are built ?
 - Projects to watch: gitbuilder,
 - ☒ Trigger only if build is stable
 - ☐ Trigger even if the build is unstable
 - ☐ Trigger even if the build fails
 - ☐ Always trigger, even if the build is aborted
- ☐ Build periodically ?
- ☐ GitHub hook trigger for GITScm polling ?
- ☐ Poll SCM ?

[Save](#) [Apply](#)

CI/CD Workflow

- Developer pushes code to GitHub repository (from Amazon Linux in Ohio)

```
[ec2-user@ip-172-31-6-25 hello]$ git push -u origin master
Username for 'https://github.com': tupakulamanoj
Password for 'https://tupakulamanoj@github.com':
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 353 bytes | 353.00 KiB/s, done.
Total 4 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/tupakulamanoj/JENKINS.git
 90250b4..d0ae74c  master -> master
branch 'master' set up to track 'origin/master'.
```

- GitHub webhook triggers the GitBuilder job in Jenkins
- GitBuilder job:
 - Pulls latest code from GitHub
 - Builds and tests the application

S	W	Name ↓	Last Success	Last Failure	Last Duration
✓	☀	gitbuilder	57 sec #4	N/A	0.24 sec
✓	☁	gitdeploy	47 sec #5	5 min 32 sec #2	14 ms

- After GitBuilder completes successfully, GitDeploy job is automatically triggered
- GitDeploy job:
 - Deploys the application to the target environment
 - Restarts services if needed

← → ↻ Not secure 54.209.249.218:8080/job/gitdeploy/5/console ☆ m ⋮

Jenkins 🔍 🔔 1 🛡️ 1 👤 T MANOJ ⌵ 🚪 log out

Dashboard > gitdeploy > #5 > Console Output

📄 Status

</> Changes

📄 Console Output

📝 Edit Build Information

🗑️ Delete build '#5'

🕒 Timings

← Previous Build

✓ Console Output

📄 Download 📄 Copy View as plain text

```
Started by upstream project "gitbuilder" build number 4
originally caused by:
  Started by GitHub push by tupakulamanoj
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/gitdeploy
[gitdeploy] $ /bin/sh -xe /tmp/jenkins8306825398615575877.sh
+ uptime
15:36:47 up 8:56, 2 users, load average: 0.00, 0.00, 0.00
Finished: SUCCESS
```

REST API Jenkins 2.492.2

Security Considerations

- Ensure both AWS instances have appropriate security groups configured
- Configure Jenkins security (authentication, authorization, etc.)
- Use SSH keys or secure credentials for deployment