SUNBEAM
Exploring New Ideas Reaching New Heights

प्रगत संगणन विकास केंद्र
CENTRE FOR DEVELOPMENT OF ADVANCED COMPUTING

Web Application Deployment on AWS with Jenkins CI/CD Pipeline, Nagios Monitoring and Snort IDS

Project By

Saloni Chambhare : 240344223007
Jayesh Mahajan : 240344223013
J.V.Tejeswar Reddy : 240344223014
Rahul Shelke : 240344223034

Under the guidance of

Mr. Sandeep Walvekar
Sunbeam Institute of Information Technology,
Pune (Maharashtra)

Objective

The goal is to securely deploy a web application on Amazon Web Services (AWS) using an automated CI/CD pipeline managed by Jenkins, while ensuring continuous monitoring and intrusion detection using Nagios and Snort. This deployment should emphasize security, automation, and real-time monitoring to safeguard the application against potential threats and ensure high availability.

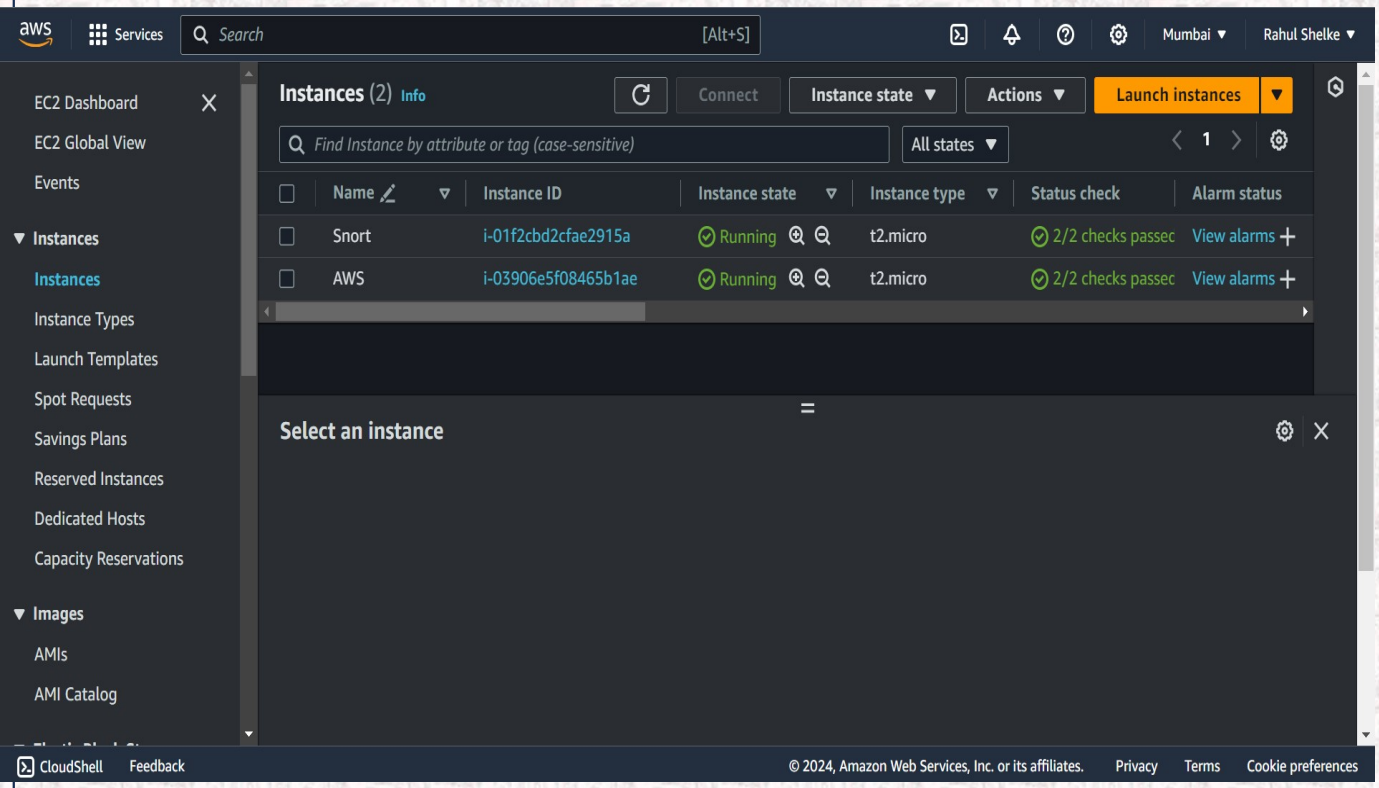
AWS Account Creation

- Visit AWS Website: https://aws.amazon.com/
- Sign Up: Visit the AWS signup page, provide email, account name, and select account type.
- Verify and Pay: Complete phone verification, enter payment details, and choose a support plan.
- Access and Configure: Log in to the AWS Management Console, configure account settings, and start using AWS services.

AWS Instance Creation

- Log In to AWS Management Console
- Access the AWS Management Console
- Navigate to EC2 Dashboard
- Select “EC2” from the “Services” menu to open the EC2 dashboard.
- Launch Instance

AWS Running Instances:



Instance1: AWS Instance

We Configured Jenkins and Nagios on this instance.

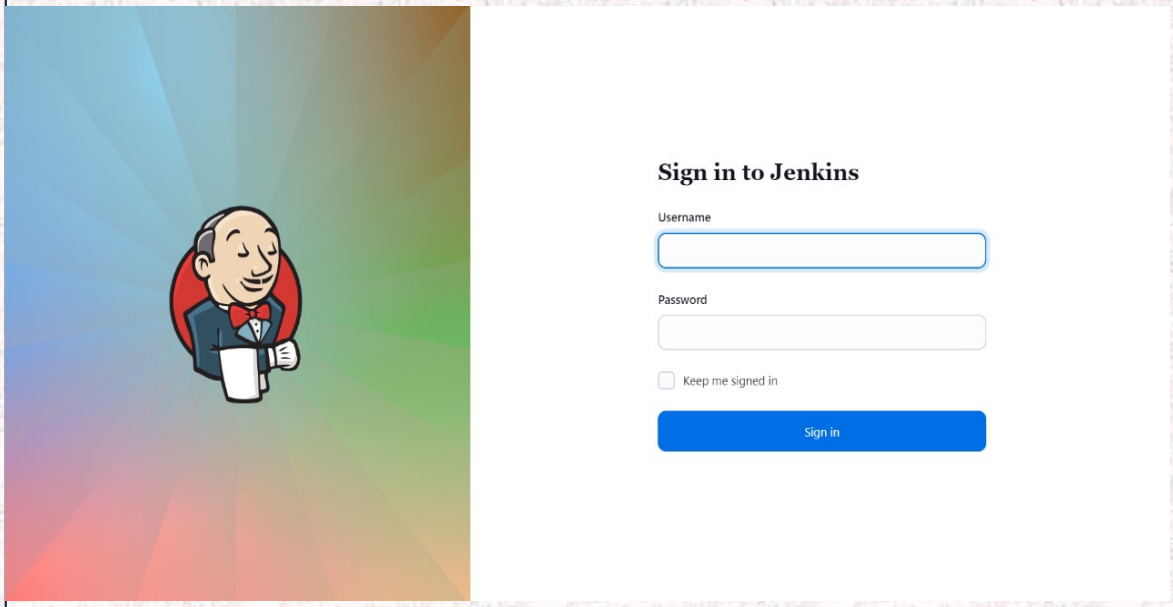
Instance2: Snort

Snort is configured on this instance.

Note:

Both the instances are on same region having same VPC so that we can ping those instances with each other.

Jenkins Installation & Configuration



Install Jenkins:
On Debian/Ubuntu, install Jenkins using apt-get command.

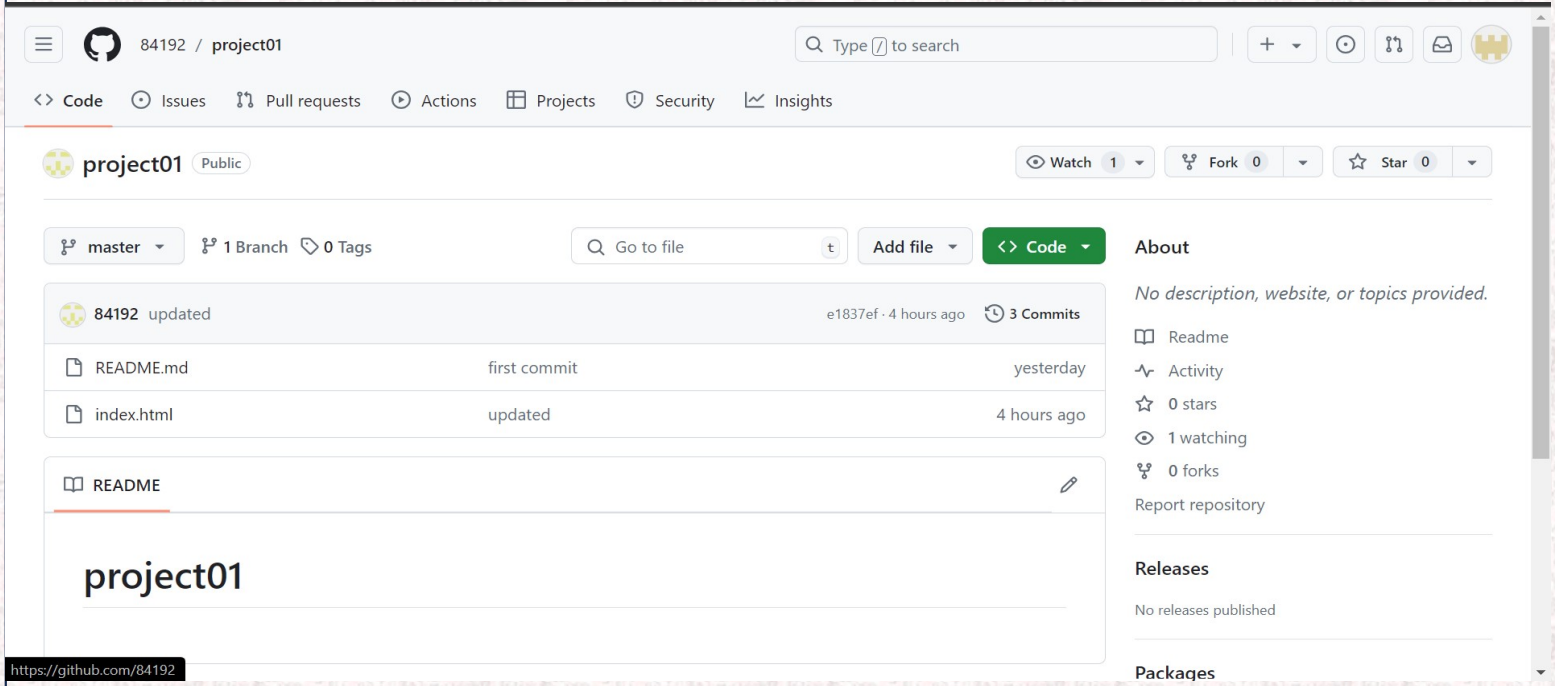
Access Jenkins Web Interface:
Open Jenkins in your browser at http://your-server-ip:8080 and retrieve the initial admin password with sudo cat /var/lib/jenkins/secrets/initialAdminPassword.

Complete Initial Setup:
Use the admin password to unlock Jenkins, install suggested plugins, and set up an admin user through the setup wizard.

SUNBEAM
Exploring New Ideas Reaching New Heights

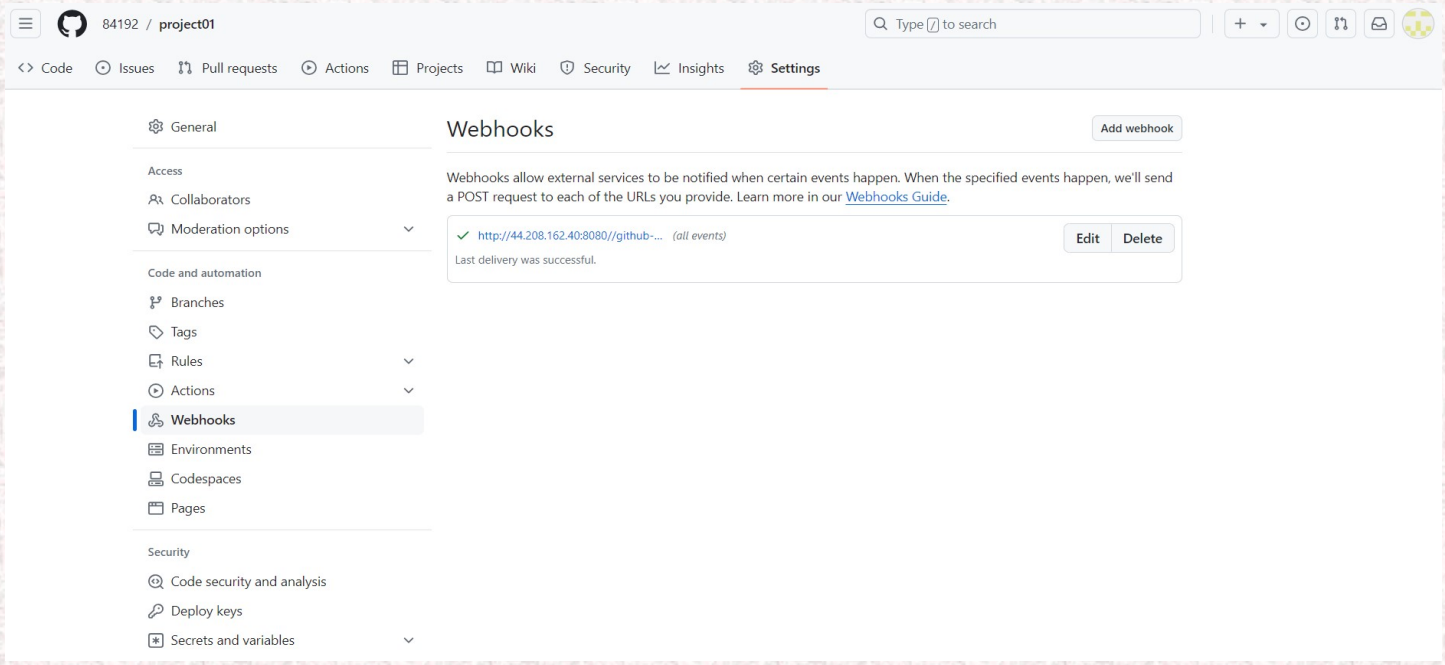
प्रगत संगणन विकास केंद्र
CENTRE FOR DEVELOPMENT OF ADVANCED COMPUTING

GitHub:



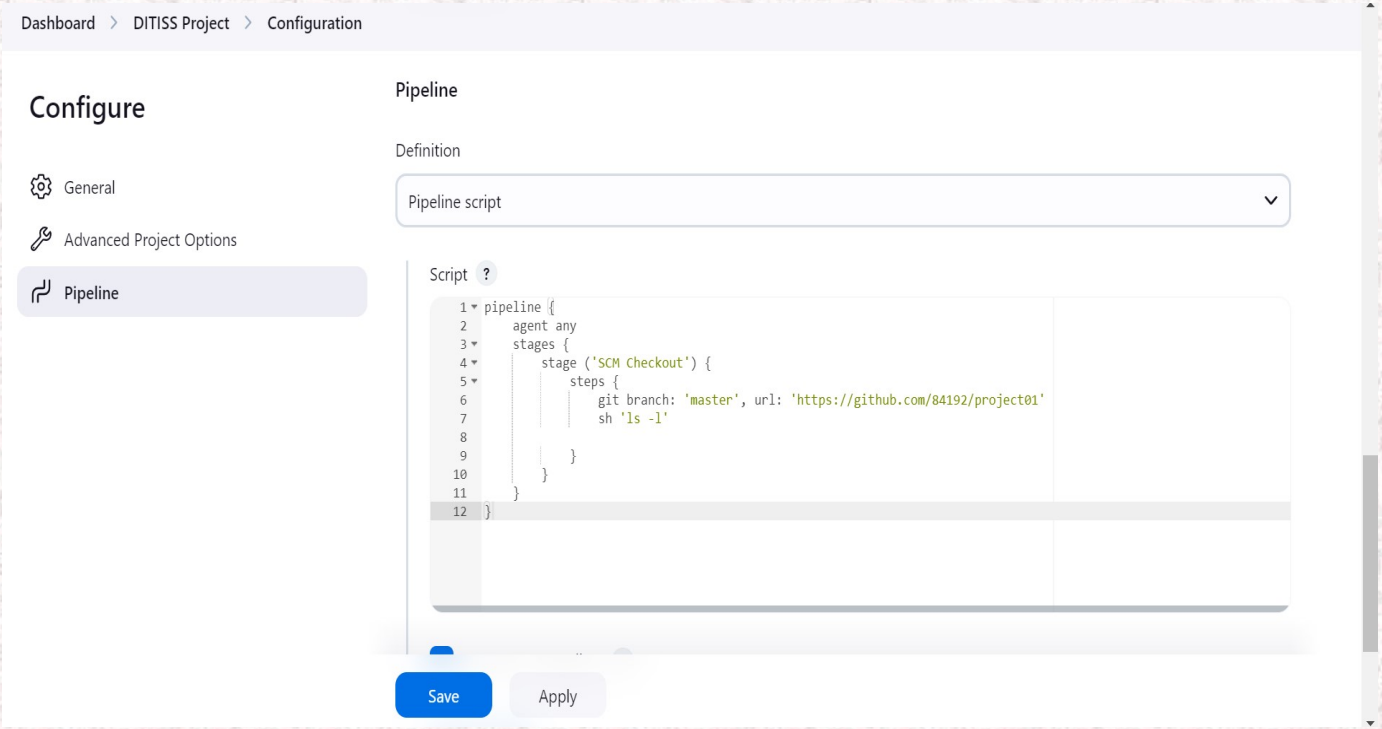
- Create a Account on GitHub
- Create a New Repository in GitHub
- Push the index.html file which is Created with the html Code of Static Website.

Webhook

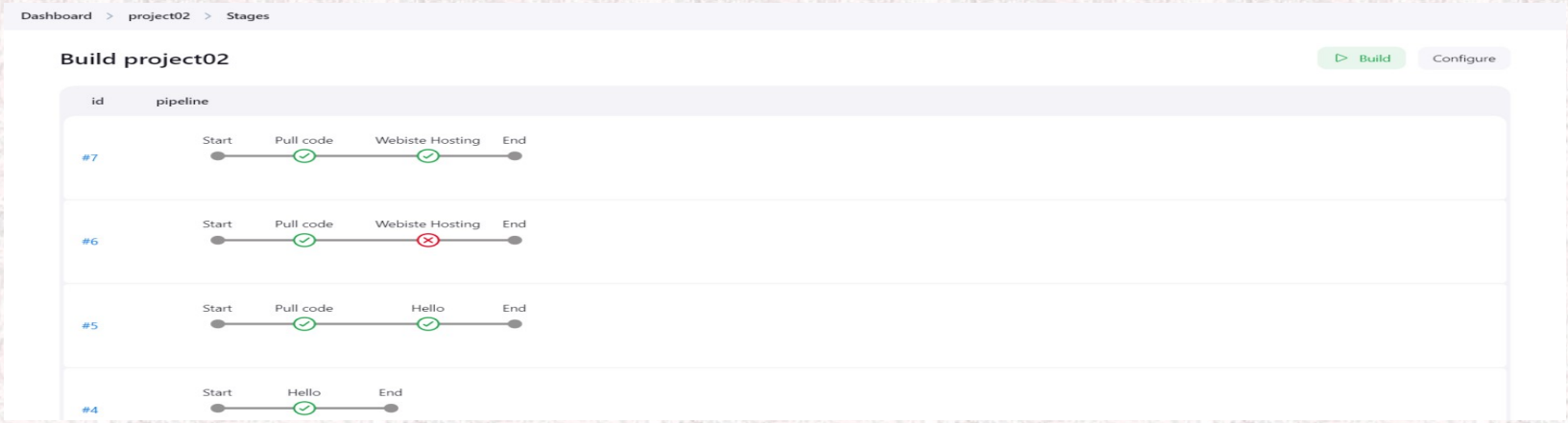


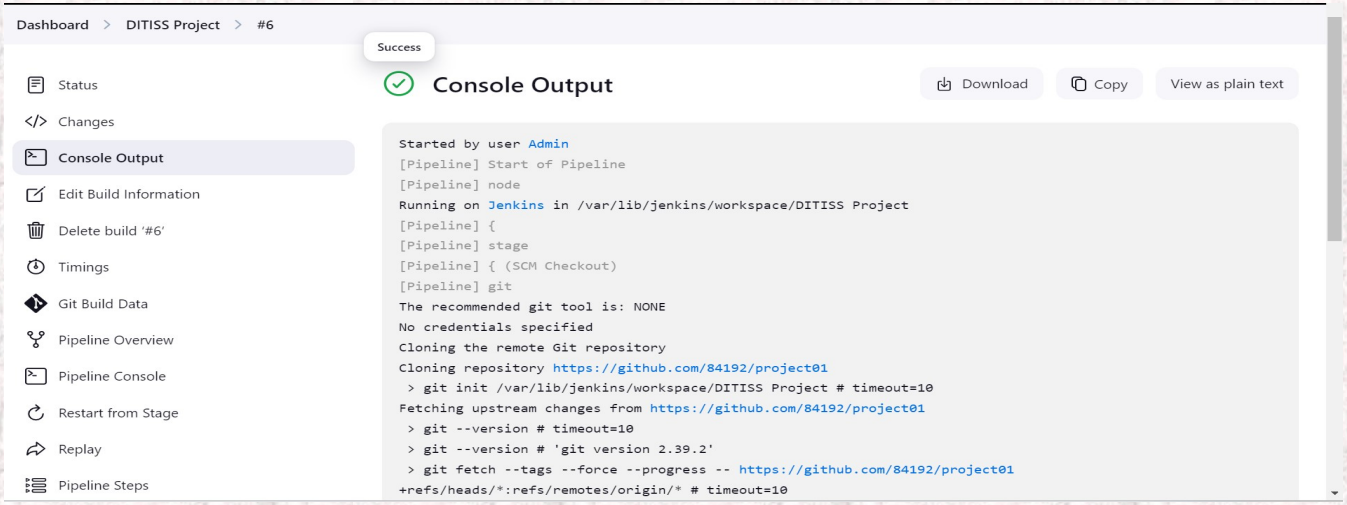
- For Webhook Creation in GitHub, go to the settings, then click on the Webhook.
- Add Webhook
- Paste Payload URL
  Example:
  http://44.208.162.40:8080//github-webhook/
- Select Content Type: Application/x-www-form-urlencoded
- Click on Create Webhook

Pipeline Creation:

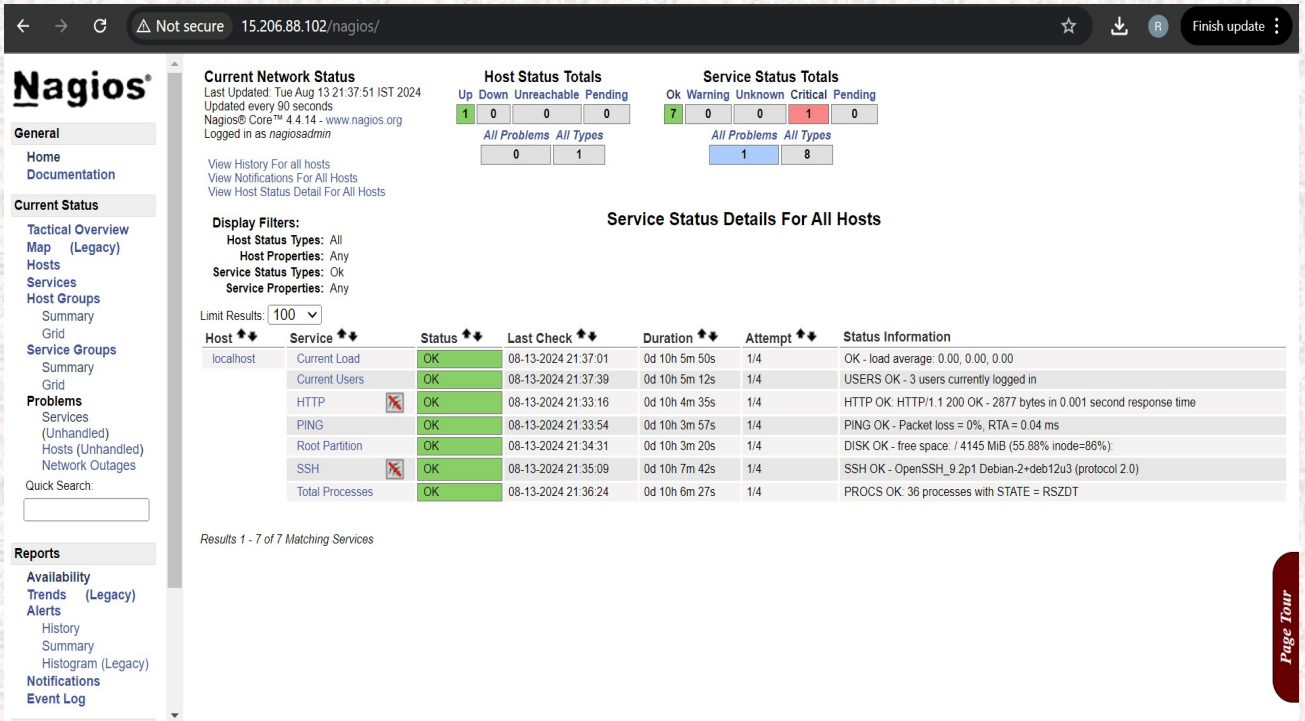


- In Jenkins, click on New Item
- Enter Item Name
- Select Pipeline Project
- Enter Project Description
- Select GitHub Project and Paste its https URL.
- Build Triggers, Click on GitHub Hook Trigger GITScm Pooling.
- Select Pipeline Script & write your Pipeline Script over there.
- Apply & Save
- After that, click on Build
- Check the Console Output
- If there is any Error, Troubleshoot it and Resolve it.


Dashboard > project02 > Stages
Build project02
Build
Configure
id
pipeline
#7
Start
Pull code
Webiste Hosting
End
#6
Start
Pull code
Webiste Hosting
End
#5
Start
Pull code
Hello
End
#4
Start
Hello
End

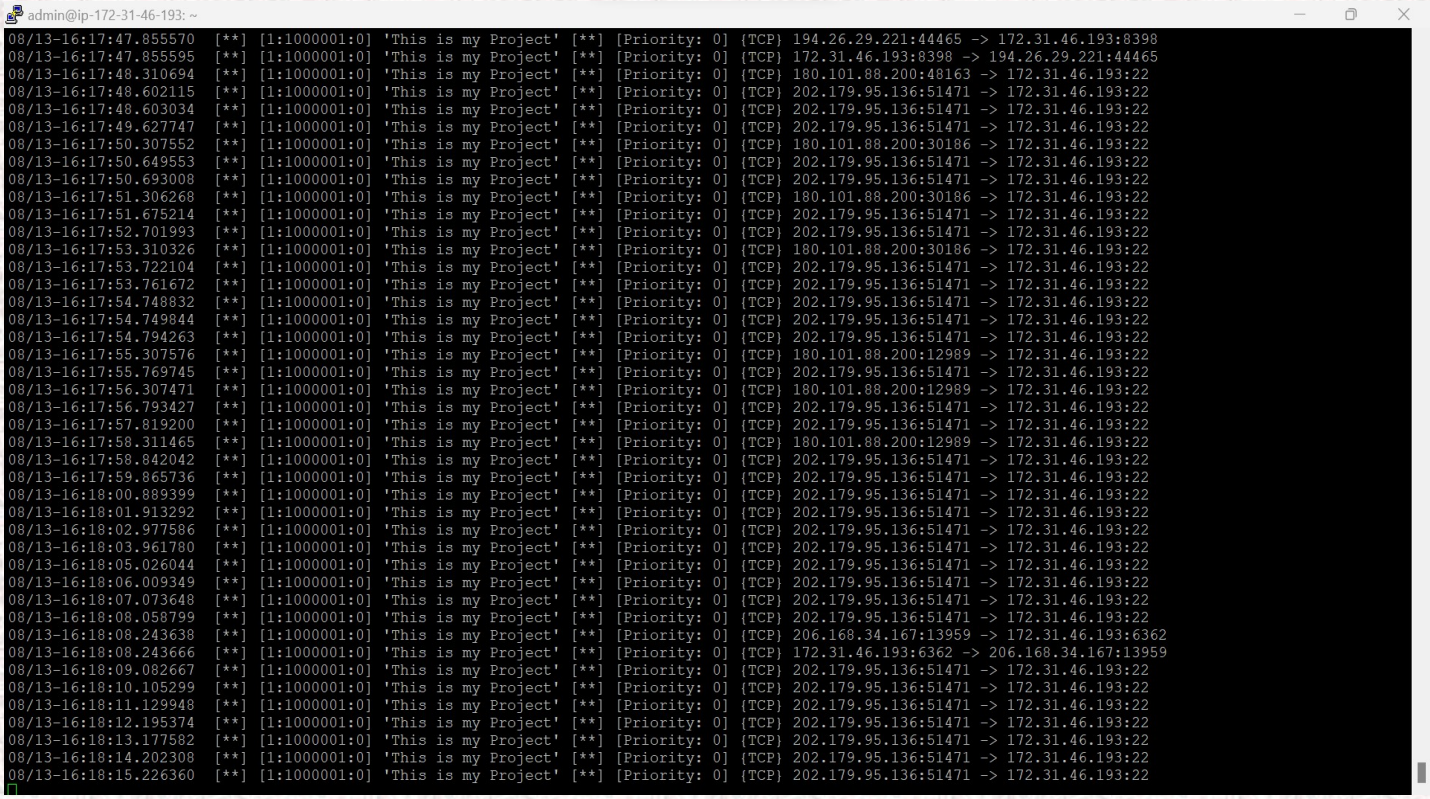

Dashboard > DITISS Project > #6
Success
Status
Changes
Console Output
Edit Build Information
Delete build '#6'
Timings
Git Build Data
Pipeline Overview
Pipeline Console
Restart from Stage
Replay
Pipeline Steps
Console Output
Download
Copy
View as plain text
Started by user Admin
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/DITISS Project
[Pipeline] {
[Pipeline] stage
[Pipeline] { (SCM Checkout)
[Pipeline] git
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning repository https://github.com/84192/project01
> git init /var/lib/jenkins/workspace/DITISS Project # timeout=10
Fetching upstream changes from https://github.com/84192/project01
> git --version # timeout=10
> git --version # 'git version 2.39.2'
> git fetch --tags --force --progress -- https://github.com/84192/project01
+refs/heads/*:refs/remotes/origin/* # timeout=10

Nagios



- Nagios offers comprehensive monitoring for IT infrastructure, including servers, networks, applications, and services.
- Its robust alerting system notifies administrators of issues promptly, enabling quick response to critical issues.
- Nagios proactively identifies potential problems, minimizing downtime and enhancing system reliability.
- Centralized management simplifies administration tasks, improving efficiency across multiple systems and services.
- Highly scalable, Nagios supports distributed monitoring setups, effectively accommodating growing infrastructure needs.

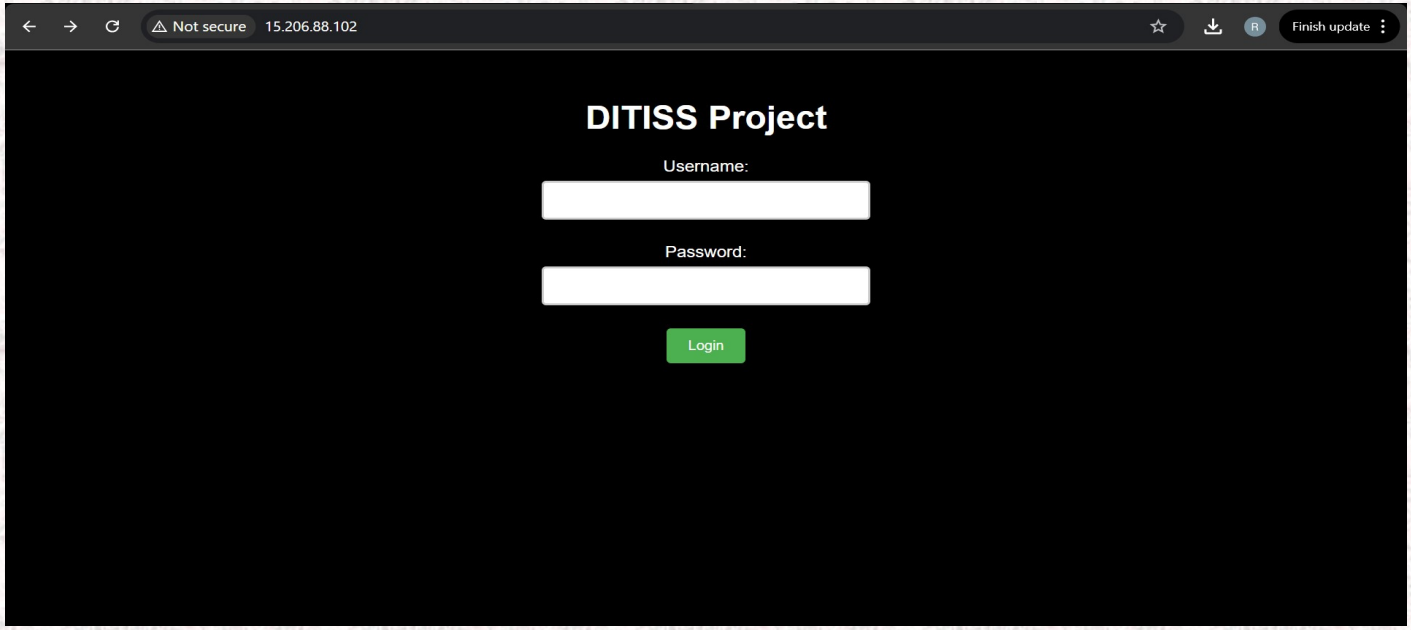
Snort



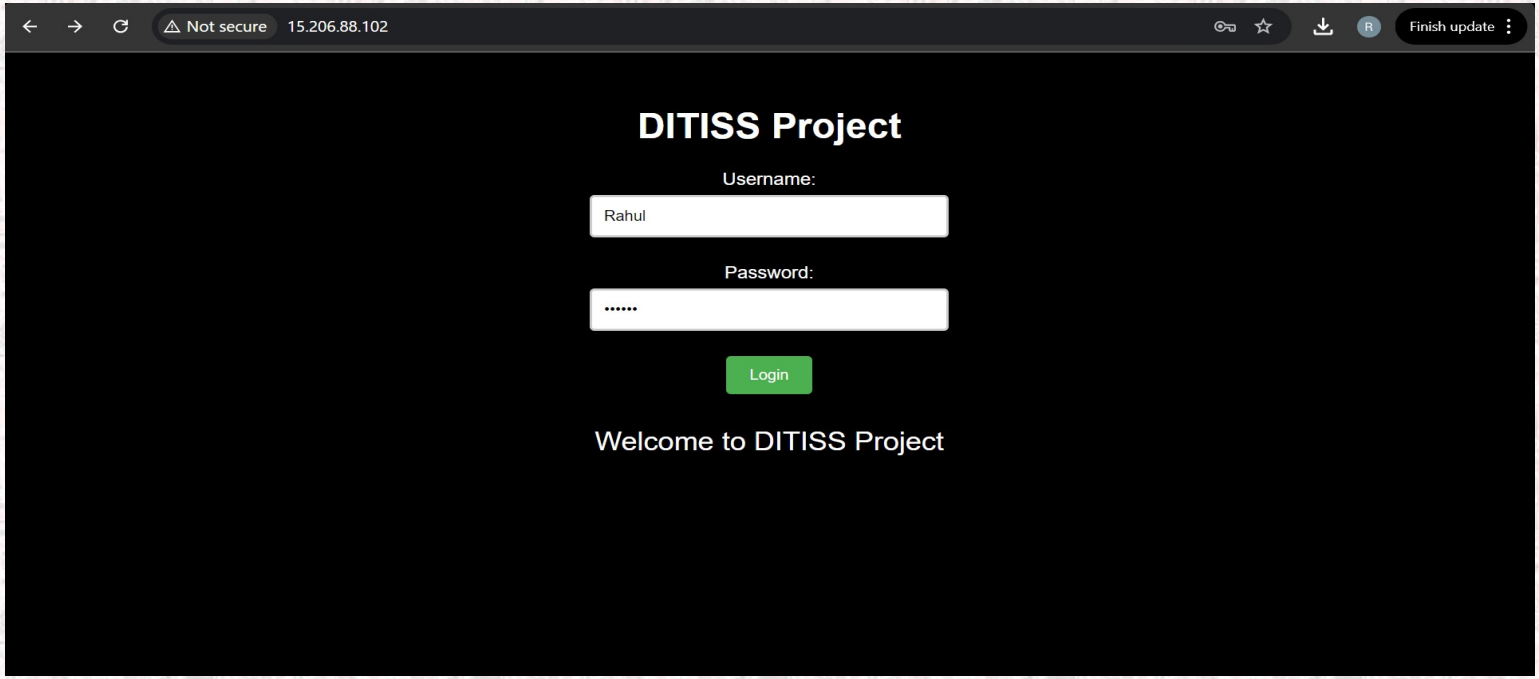
Snort Does the following work:

- **Detect Intrusions:** Identify and alert on malicious or unauthorized network activities, such as attacks or breaches.
- **Prevent Attacks:** Block or mitigate threats by using IPS capabilities to prevent harmful traffic from reaching the network.
- **Analyze Network Traffic:** Provide detailed logs and reports for forensic analysis and network security assessments.
- **Enhance Security:** Improve overall network security by integrating with other security measures and offering customizable rules for detecting a wide range of threats.

Static Website Output



- The Output of our Project, that is a **Static Website** will look like this.

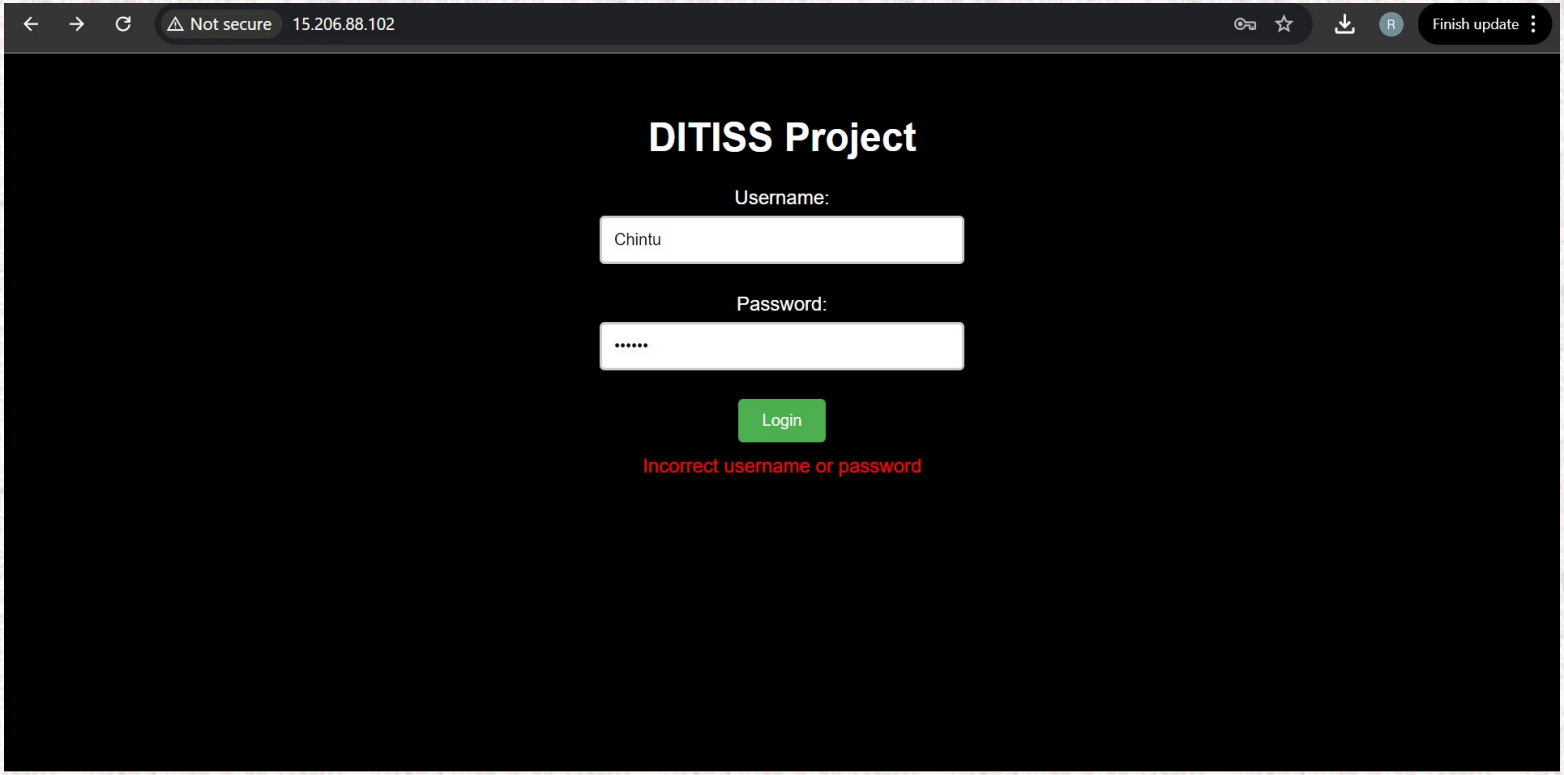


We have added Credentials of our 4 Group Members with Usernames:

- **Rahul**
- **Saloni**
- **Tejeswar**
- **Jayesh**

Each Member has given a particular Password.

If the Credentials Entered are Correct, it will show "**Welcome to DITISS Project**".



If the Credentials are not Correct, it will show "**Incorrect username or password**"

Conclusion:

Hence, we have successfully deployed a highly available and secure web server environment on Amazon Web Services (AWS). And ensured the reliability, performance, and security of the web application while maintaining efficient development and operational processes.

Future Scope:

The future scope for Secure Web Application Deployment on AWS with Jenkins CI/CD, Nagios monitoring, and Snort IDS involves advancing automation, security, and scalability. Enhanced integration of Jenkins with Infrastructure-as-Code (IaC) and serverless technologies will streamline deployments and improve efficiency. Advanced threat detection and response can be achieved by integrating Snort with machine learning and other security tools, while comprehensive monitoring solutions combining Nagios with real-time analytics will offer better visibility and proactive management. Additionally, adopting containerization, improving disaster recovery strategies, and ensuring compliance with evolving regulations will further bolster the resilience and security of web applications.

SUNBEAM
Exploring New Ideas Reaching New Heights

प्रगत संगणन विकास केंद्र
CENTRE FOR DEVELOPMENT OF ADVANCED COMPUTING

CERTIFICATE

This is to certify that the project report entitled **"Web Application Deployment on AWS with Jenkins CI/CD Pipeline, Nagios Monitoring and Snort IDS"**, submitted by **Rahul Shelke*(PRN Number:* 240344223034*)*** is the authentic work completed under our careful supervision and guidance. This project fulfills the requirements for the award of Post Graduate Diploma in IT Infrastructure, Systems, and Security (PG-DITISS) at Sunbeam Institute of Information Technology, Pune (M.S.).

Place: Pune (Maharashtra)

Date: 14-08-2024

Mr. Sandeep Walvekar
(Guide)

Mr. Nitin Kudale
(CEO)

Mr. Vishal Salunkhe
(Course Coordinator)

Sunbeam Institute of Information Technology

Pune (M.S.) – 411057


Thank You!