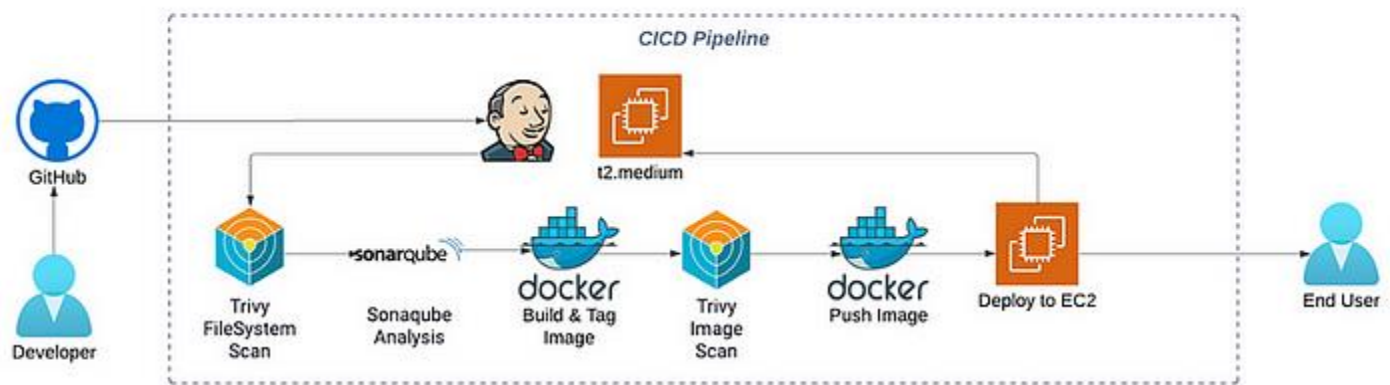


Integrating Trivy and SonarQube with Jenkins Pipeline

1



Lasantha Sanjeewa Silva

Architecture Diagram

In this project, I will create a full CI/CD pipeline using Jenkins, incorporating SonarQube for code quality analysis and Trivy for container security scanning.

SonarQube is used for code quality scans and code coverage, while Trivy is utilized for filesystem and Docker image security scanning.

Step 1

Create an Ubuntu EC2 instance using a t2.medium or larger instance type. Ensure to generate a key pair for connecting to the EC2 instance. Finally, proceed with the instance creation.

EC2 > Instances > Launch an instance

Launch an instance [Info](#)

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags [Info](#)

Name

 [Add additional tags](#)

▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Recents Quick Start

Amazon Linux
aws

macOS
Mac

Ubuntu
ubuntu

Windows
Microsoft

Red Hat
Red Hat

SUSE Linux
SUS

[Browse more AMIs](#)
Including AMIs from AWS, Marketplace and the Community

▼ Summary

Number of instances [Info](#)

Software Image (AMI)
Canonical, Ubuntu, 20.04 LTS, ...[read more](#)
ami-0b4750268a88e78e0

Virtual server type (instance type)
t2.micro

Firewall (security group)
New security group

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

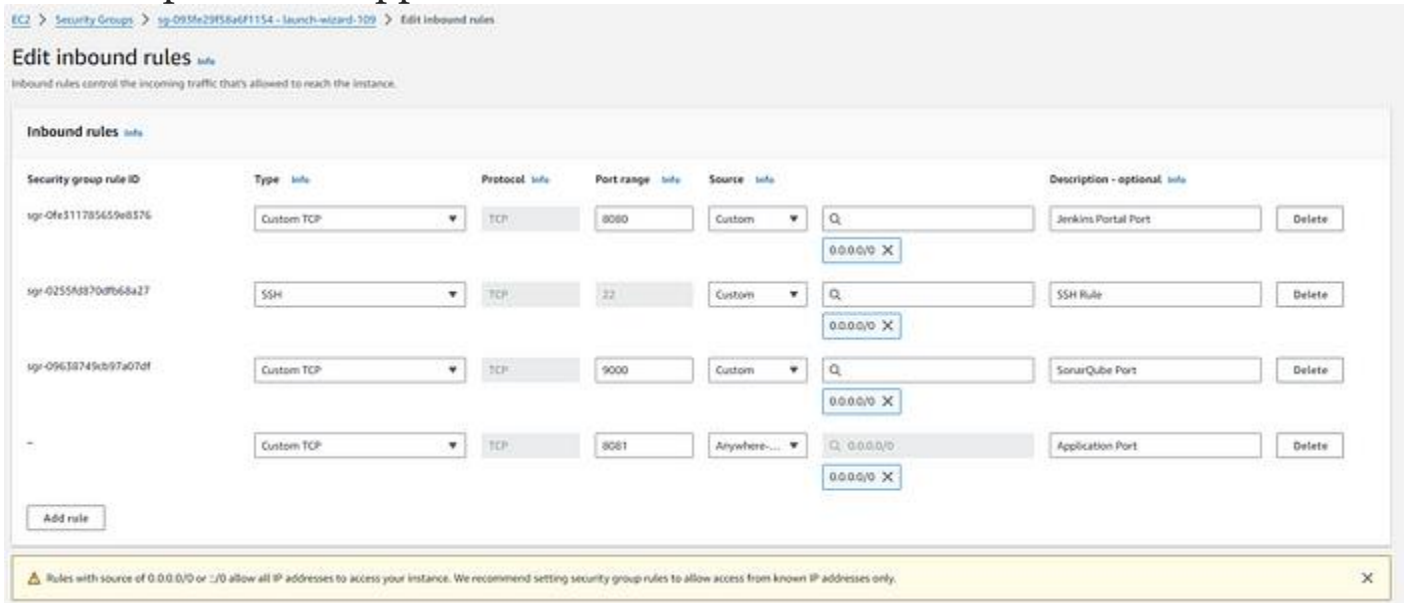
Cancel [Launch instance](#)

EC2 Instance

Step 2

Next, access the EC2 instance, select the security group, and add inbound rules to allow traffic for Jenkins, SonarQube, and the final application.

- 8080 port: Jenkins
- 9000 port: SonarQube
- 8081 port: Web Application



Security Group Inbound Rules

Step 3

Next, connect to the EC2 instance using EC2 Instance Connect or an SSH client. Log in as the admin user and run the following script to install Jenkins.

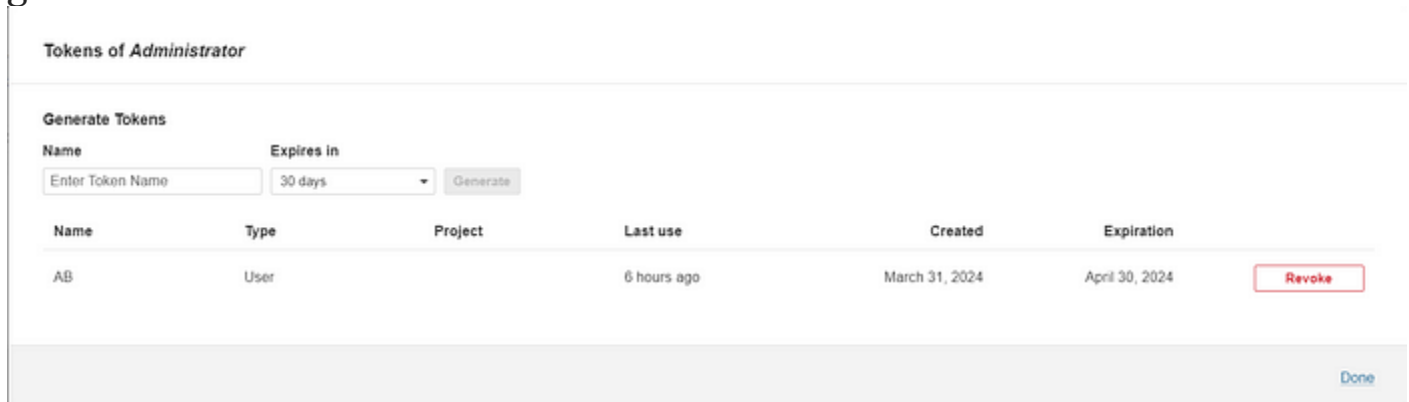
Get the admin password using the following command.

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

Step 4

Next, run SonarQube on the Ubuntu EC2 instance using the script provided below. Since SonarQube will run inside a Docker container, ensure Docker is installed before executing the SonarQube setup.

Next, log in to the SonarQube console at `EC2_Public_IP:9000`, and generate a token for the administrator.



Tokens of Administrator

Generate Tokens

Name: Expires in:

Name	Type	Project	Last use	Created	Expiration	
AB	User		6 hours ago	March 31, 2024	April 30, 2024	<input type="button" value="Revoke"/>

[Done](#)

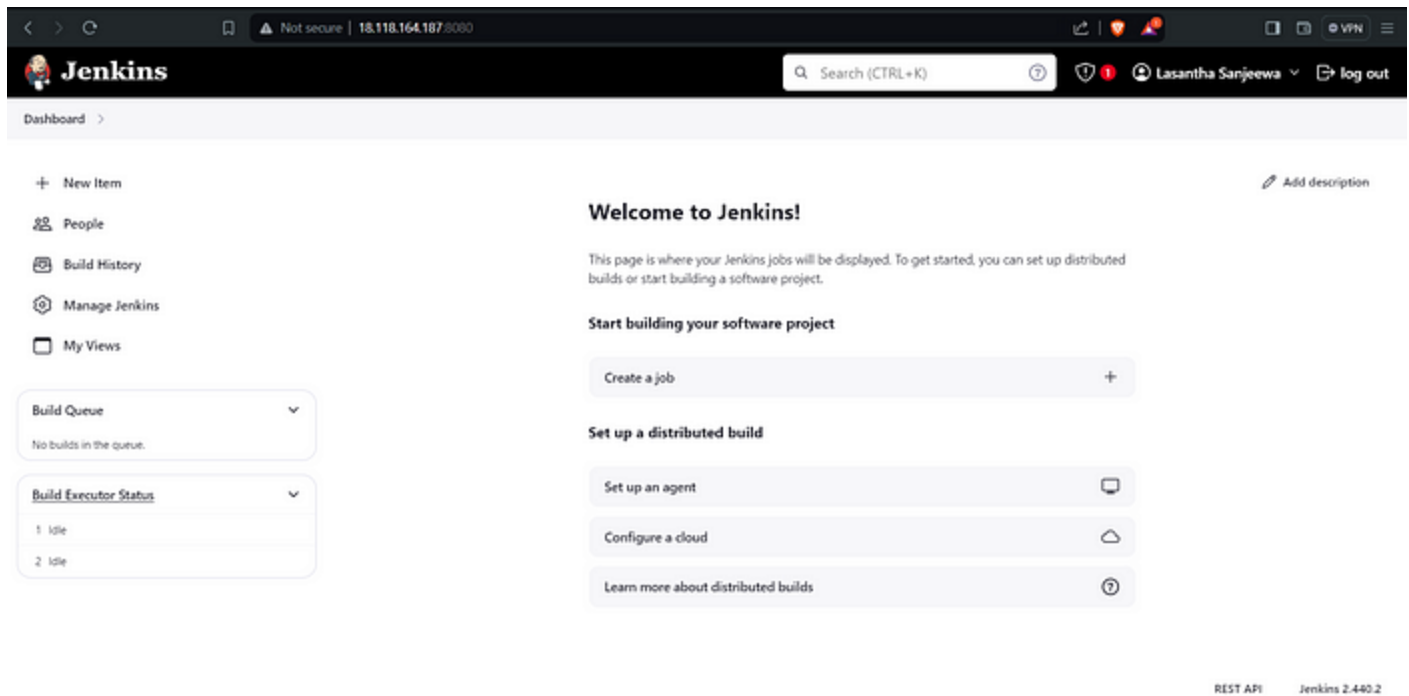
SonarQube Admin Token

Step 5

Install Trivy using the following command. Trivy is used for filesystem scanning and container image scanning.

Step 6

Next, log in to Jenkins at `Public_IP:8080` and create a user account.



Install SonarQube, Docker, and the Docker Pipeline plugin in Jenkins.

Step 7

In Jenkins global settings, configure the SonarQube and Docker installations.



SonarQube Installation

Docker

Name

docker

☒

Install automatically ?

Download from docker.com

Docker version ?

latest

Add Installer

Add Docker

Docker Installation

Step 8

Add two Jenkins credentials: one for Docker and one for SonarQube.

Global credentials (unrestricted)

+ Add Credentials

Credentials that should be available irrespective of domain specification to requirements matching.

ID	Name	Kind	Description
 docker-cred	lasanthasanjeeva/*****	Username with password	
 sonar	sonar	Secret text	

Icon:

S

M

L

Global Credentials


Step 9

Next, click 'Create Job' and select 'Pipeline' as the template.


Enter an item name

jenkins-pipeline


» Required field




Freestyle project
Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.




Maven project
Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.




Pipeline
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.




Multi-configuration project
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.



Folder
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.



Multibranch Pipeline
Creates a set of Pipeline projects according to detected branches in one SCM repository.



Organization Folder
Creates a set of multibranch project subfolders by scanning for repositories.

OK

If you want to create a new item from other existing, you can use this option:

Create Jenkins Job

Next, enable 'Discard Old Builds' and set the 'Max # of Builds' to 2.

General

Enabled 

Description

Plain text [Preview](#)

☒ Discard old builds 

Strategy

Log Rotation

Days to keep builds

if not empty, build records are only kept up to this number of days

Max # of builds to keep

if not empty, only up to this number of build records are kept

Advanced 

Discard Old Builds

Next, go to the last option and select ‘Pipeline Script from SCM.’ Choose ‘Git’ as the repository type and select your Git credentials. Ensure that the Jenkinsfile inside the Git repository includes the repository URL. You can either clone or fork my sample Git repository for this purpose.

Pipeline

Definition

Pipeline script from SCM

SCM ?

Git

Repositories ?

Repository URL ?

https://github.com/sanju2/jenkins-cicd-pipeline.git

Credentials ?

none

+ Add

Advanced

Add Repository

Branches to build ?

Branch Specifier (blank for 'any') ?

*/main

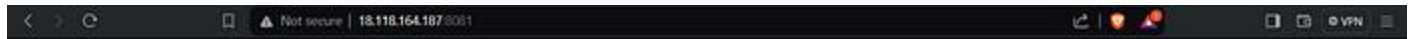
Pipeline Script from SCM

Finally, save the pipeline configuration and click 'Build Now' to start the build process.



Jenkins Pipeline

Finally, you should see the application running.



Jenkins CICD Demo With Trivy & SonarQube - V1

Website

Git Repository: <https://github.com/sanju2/jenkins-cicd-pipeline.git>

Thanks for reading the Article.

Connect with me

LinkedIn <https://www.linkedin.com/in/lasanthasilva>

Twitter <https://twitter.com/LasanthaSilva96>