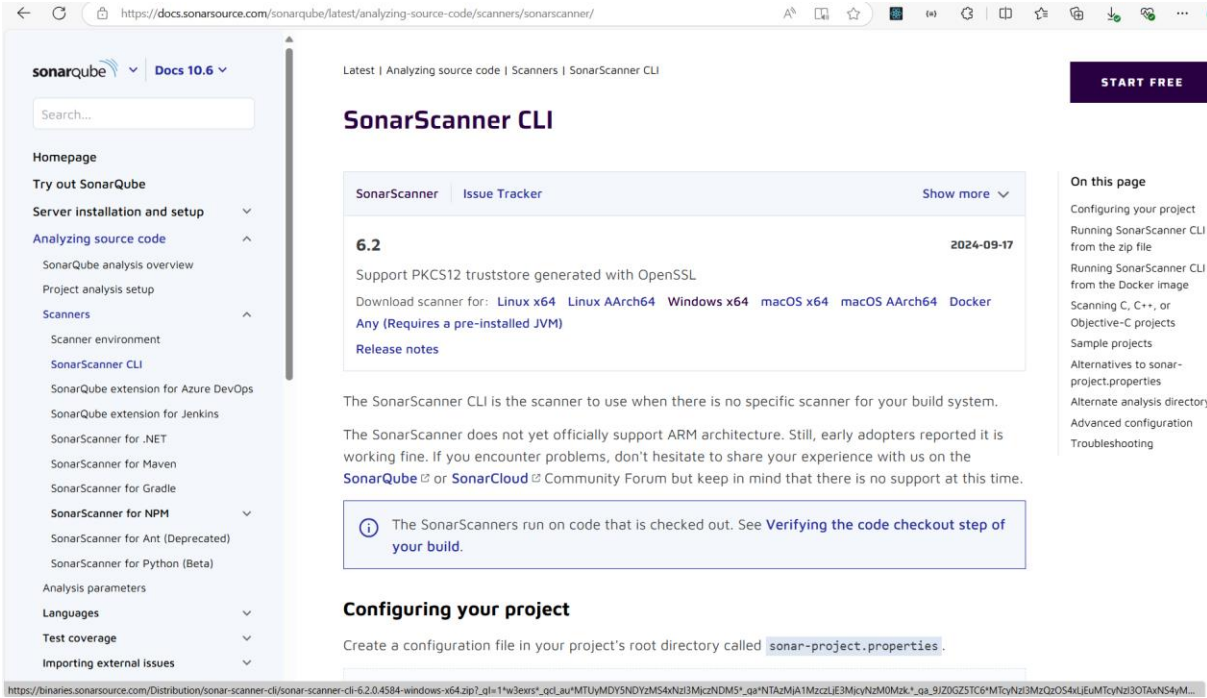


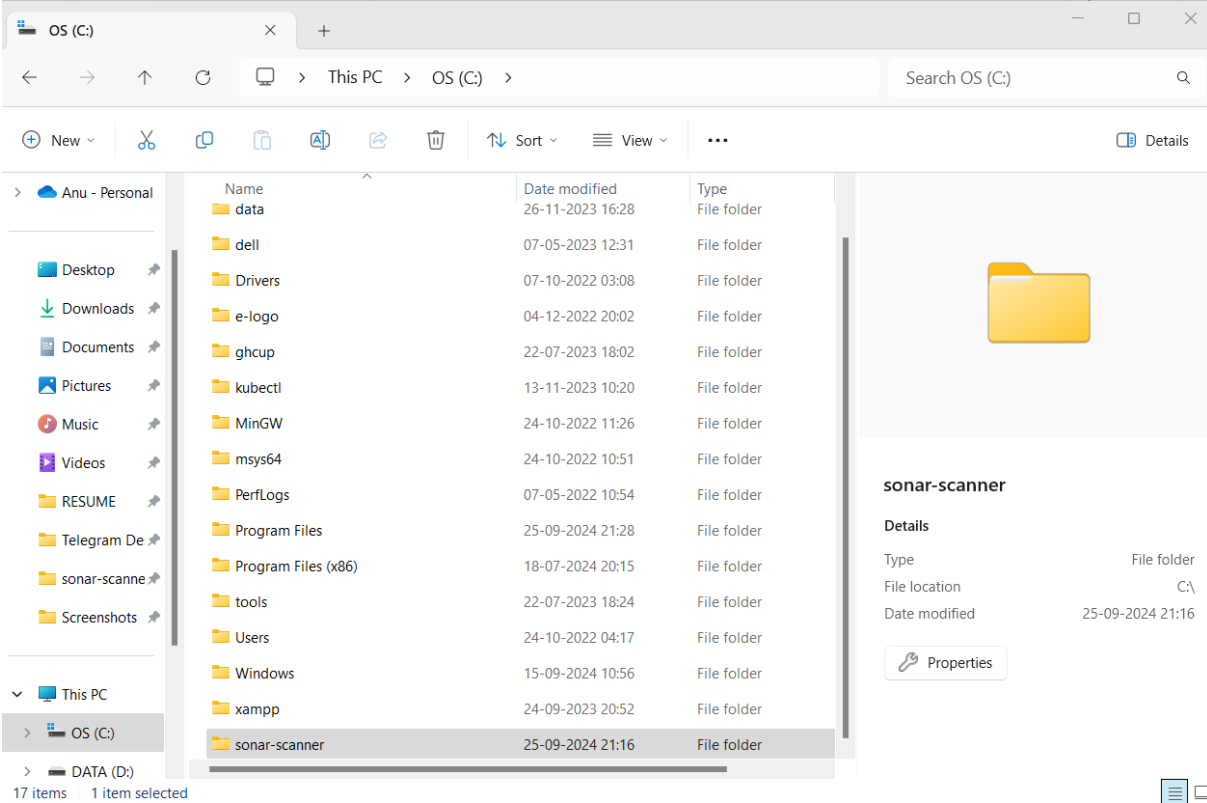
Experiment 8

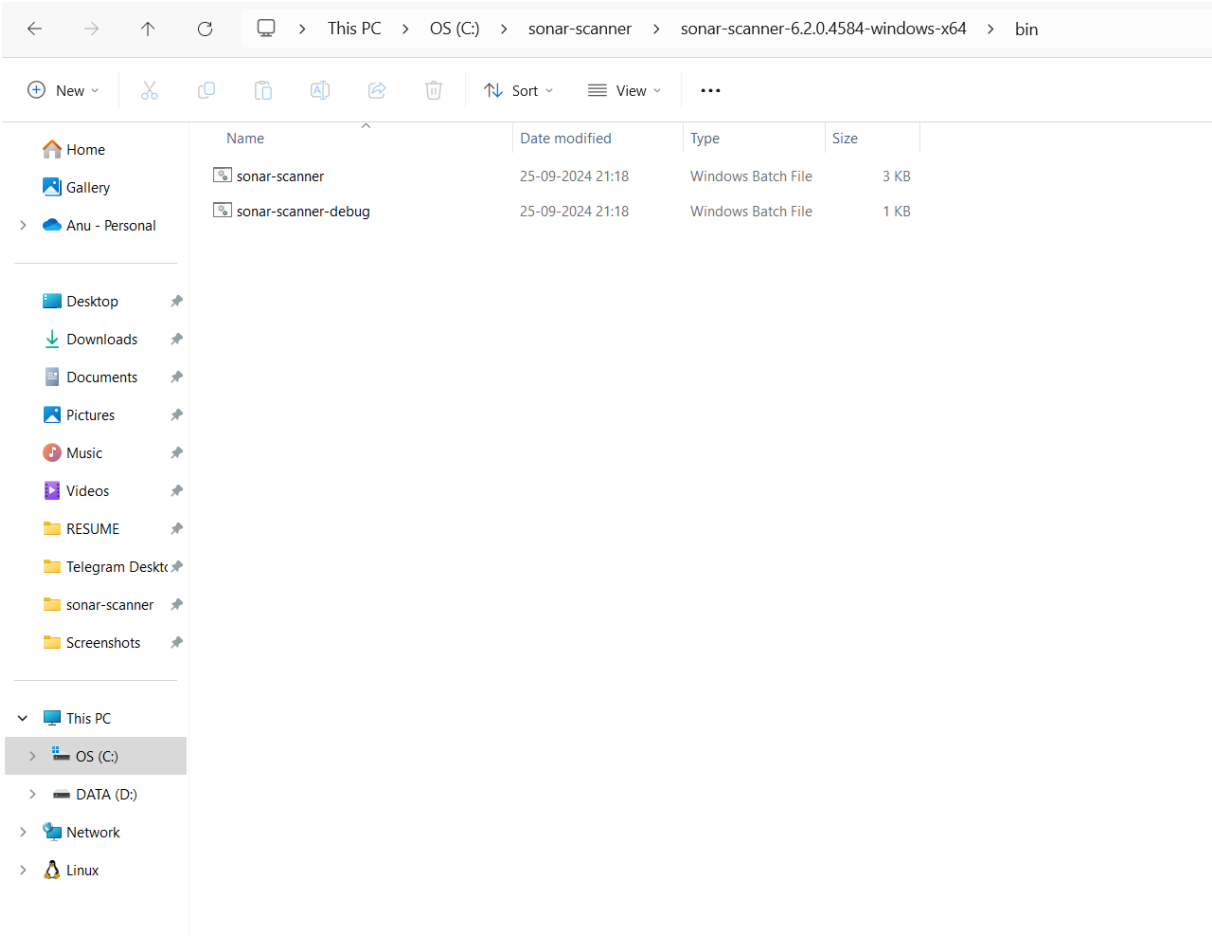
Aim: Create a Jenkins CI/CD Pipeline with SonarQube / GitLab Integration to perform a static analysis of the code to detect bugs, code smells, and security vulnerabilities on a sample Web /Java / Python application.

Step 1 : Visit the following link to download the SonarScanner CLI - <https://docs.sonarsource.com/sonarqube/latest/analyzing-source-code/scanners/sonarscanner/> and then click on Windows x-64 to download the zip file.

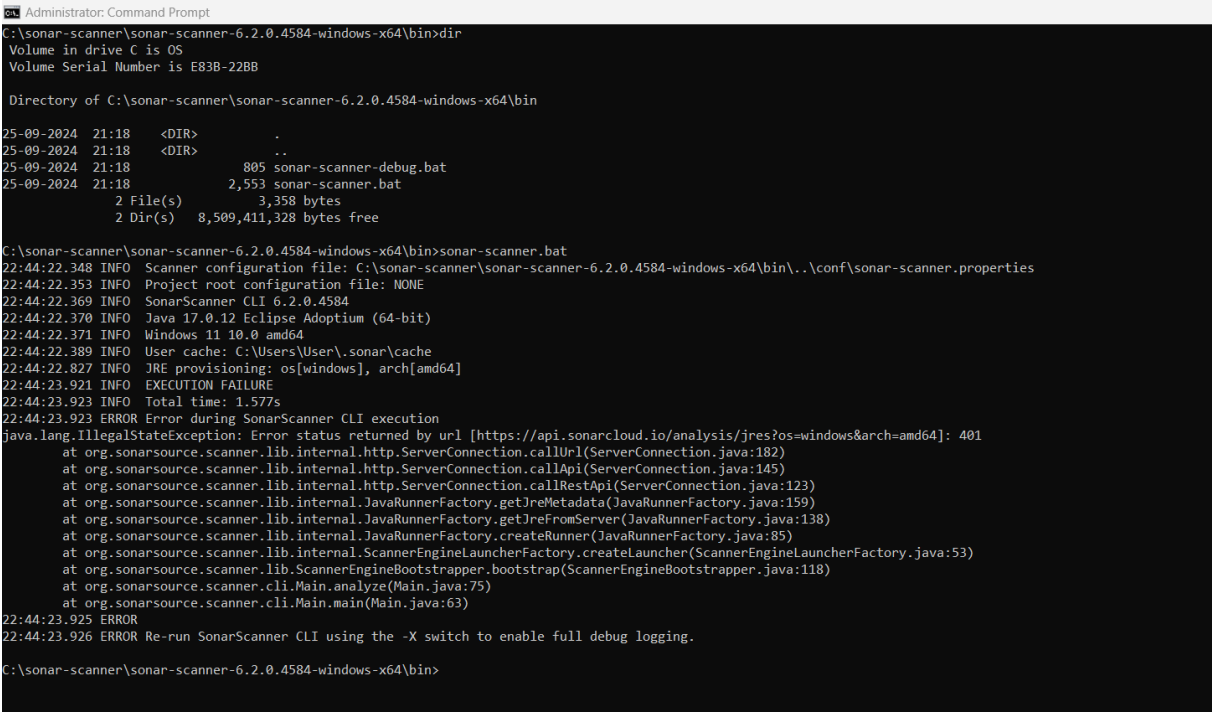


Step 2: Extract the content in C drive and name the folder sonar-scanner






Step 3: Open Command Prompt and run as administrator and run the following commands –
cd C:\sonar-scanner\sonar-scanner-6.2.0.4584-windows-x64\bin
dir
sonar-scanner.bat




Step 4: Open Jenkins and create a pipeline and name the pipeline SonarQube Pipeline and then click on okay.

Enter an item name


** 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.


Branch Pipeline

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

Step 5: In the configuration, under the Pipeline Section write the following Pipeline Script -

```
node {
```

```
    stage('Cloning the GitHub Repo') {
        git 'https://github.com/shazforiot/MSBuild_firstproject.git'
    }

    stage('SonarQube analysis') {
        withSonarQubeEnv('sonarqube') {
            bat "C:/sonar-scanner/sonar-scanner-6.2.0.4584-windows-x64/bin/sonar-scanner.bat \
                -D sonar.login=admin \
                -D sonar.password=sonarqube \
                -D sonar.projectKey=sonarqube-test \
                -D sonar.exclusions=vendor/**,resources/**,**/*.java \
                -D sonar.host.url=http://127.0.0.1:9000/"
        }
    }
}
```

Then click on the save button.

Dashboard > SonarQube Pipeline > Configuration

Configure

General

Advanced Project Options

Pipeline

Pipeline

Definition

Pipeline script

Script ?

```
1 - node {
2   stage('Cloning the GitHub Repo') {
3     git 'https://github.com/shazforiot/MSBuild_firstproject.git'
4   }
5   stage('SonarQube analysis') {
6     withSonarQubeEnv('sonarqube') {
7       bat "C:/sonar-scanner/sonar-scanner-6.2.0.4584-windows-x64/bin/sonar-scanner.bat \
8         -D sonar.login=admin \
9         -D sonar.password=sonarqube \
10        -D sonar.projectKey=sonarqube-test \
11        -D sonar.exclusions=vendor/**,resources/**,**/*.java \
12        -D sonar.host.url=http://127.0.0.1:9000/"
13      }
14    }
15  }
16 }
```

☒ Use Groovy Sandbox ?

[Pipeline Syntax](#)

Save

Apply

REST API Jenkins 2.452.3

Step 6: Now, click on Build Now and the build is successful.

Jenkins

Q Search (CTRL+K)

Anuprita Mhapankar log out

Dashboard > SonarQube Pipeline >

Status

Changes

Build Now

Configure

Delete Pipeline

Full Stage View

SonarQube

Stages

Rename

Pipeline Syntax

Build History

trend

Filter...

#10

Sep 25, 2024, 9:55 PM

#9

Sep 25, 2024, 9:52 PM

#8

SonarQube Pipeline

Add description

Disable Project

Stage View

Average stage times:
(Average full run time: ~22s)

	Cloning the GitHub Repo	SonarQube analysis
#10	1s	20s
#9	19s	549ms failed
#8	1s	1s failed
#7	1s	957ms failed

Dashboard > SonarQube Pipeline > #10

```
21:56:16.244 INFO ----- Run sensors on project
21:56:16.428 INFO Sensor C# [csharp]
21:56:16.429 WARN Your project contains C# files which cannot be analyzed with the scanner you are using. To analyze C# or VB.NET, you must use the
SonarScanner for .NET 5.x or higher, see https://redirect.sonarsource.com/doc/install-configure-scanner-msbuild.html
21:56:16.429 INFO Sensor C# [csharp] (done) | time=1ms
21:56:16.430 INFO Sensor Analysis Warnings import [csharp]
21:56:16.432 INFO Sensor Analysis Warnings import [csharp] (done) | time=2ms
21:56:16.432 INFO Sensor C# File Caching Sensor [csharp]
21:56:16.432 WARN Incremental PR analysis: Could not determine common base path, cache will not be computed. Consider setting
'sonar.projectBaseDir' property.
21:56:16.432 INFO Sensor C# File Caching Sensor [csharp] (done) | time=1ms
21:56:16.433 INFO Sensor Zero Coverage Sensor
21:56:16.450 INFO Sensor Zero Coverage Sensor (done) | time=16ms
21:56:16.494 INFO CPD Executor Calculating CPD for 0 files
21:56:16.494 INFO CPD Executor CPD calculation finished (done) | time=0ms
21:56:16.530 INFO SCM revision ID 'f2bc042c04c6e72427c380bcae6d6fee7b49adf'
21:56:16.704 INFO Analysis report generated in 178ms, dir size=200.5 kB
21:56:16.773 INFO Analysis report compressed in 68ms, zip size=21.9 kB
21:56:16.930 INFO Analysis report uploaded in 155ms
21:56:16.931 INFO ANALYSIS SUCCESSFUL, you can find the results at: http://127.0.0.1:9000/dashboard?id=sonarqube-test
21:56:16.931 INFO Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
21:56:16.932 INFO More about the report processing at http://127.0.0.1:9000/api/ce/task?id=af67fb15-719a-4b23-8f38-5edc7a765dae
21:56:16.940 INFO Analysis total time: 16.723 s
21:56:16.943 INFO SonarScanner Engine completed successfully
21:56:17.042 INFO EXECUTION SUCCESS
21:56:17.044 INFO Total time: 19.574s

[Pipeline] }
[Pipeline] // withSonarQubeEnv
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

Step 7: Now, visit <http://127.0.0.1:9000/dashboard?id=sonarqube-test> to see the result.

← ↻ ⓘ 127.0.0.1:9000/dashboard?id=sonarqube-test&codeScope=overall ☆

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration More 🔍

☆ sonarqube-test / ⓘ main ✓ ?

Overview Issues Security Hotspots Measures Code Activity Project Settings ▾ Project Information

✓ Quality Gate ⓘ

Passed

⚠ The last analysis has warnings. [See details](#)

New Code

Overall Code

Security

0 Open issues

0 H 0 M 0 L

A

Reliability

0 Open issues

0 H 0 M 0 L

A

Maintainability

0 Open issues

0 H 0 M 0 L

A

Accepted issues

0

Valid issues that were not fixed

Coverage

On 0 lines to cover.

Duplications

0.0%

On 86 lines.

Security Hotspots

0

A