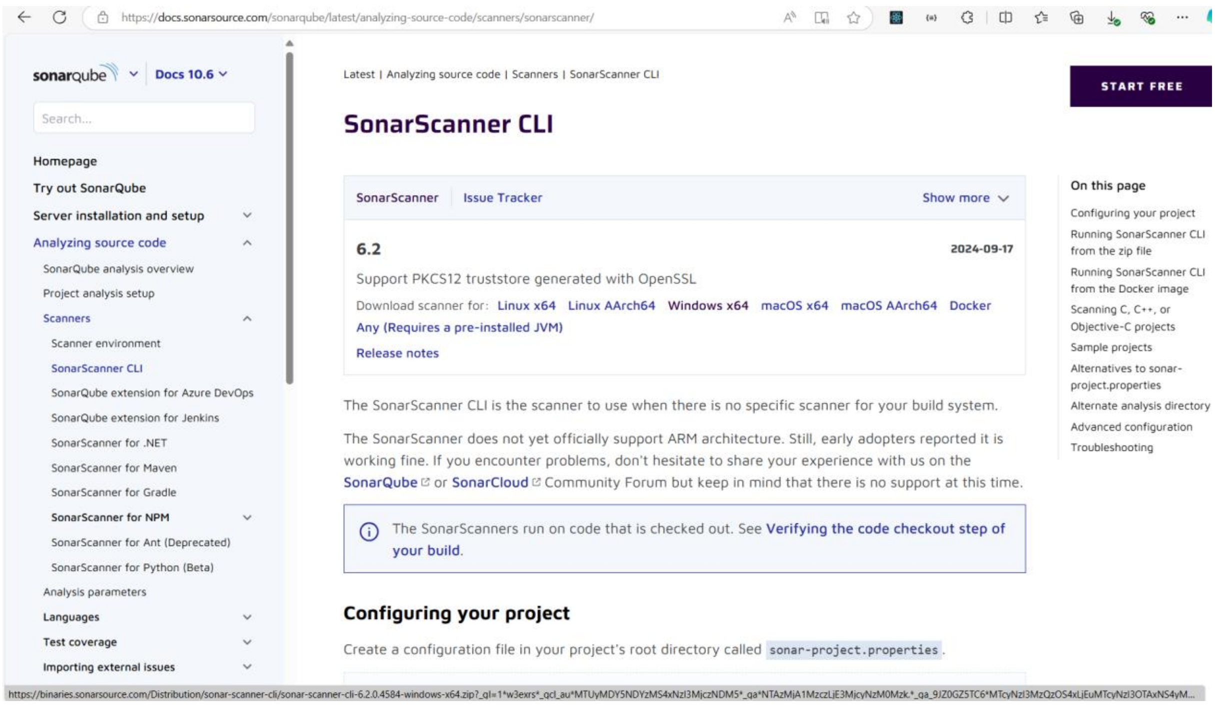
Sneha Patra **D15A** **41**

Experiment 8

Aim: Create a Jenkins CICD 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.



https://docs.sonarsource.com/sonarqube/latest/analyzing-source-code/scanners/sonarscanner/

sonarqube Docs 10.6

Search...

Homepage
Try out SonarQube
Server installation and setup
Analyzing source code
SonarQube analysis overview
Project analysis setup
Scanners
Scanner environment
SonarScanner CLI
SonarQube extension for Azure DevOps
SonarQube extension for Jenkins
SonarScanner for .NET
SonarScanner for Maven
SonarScanner for Gradle
SonarScanner for NPM
SonarScanner for Ant (Deprecated)
SonarScanner for Python (Beta)
Analysis parameters
Languages
Test coverage
Importing external issues

Latest | Analyzing source code | Scanners | SonarScanner CLI

START FREE

SonarScanner CLI

SonarScanner Issue Tracker Show more

6.2 2024-09-17

Support PKCS12 truststore generated with OpenSSL

Download scanner for: Linux x64 Linux AArch64 Windows x64 macOS x64 macOS AArch64 Docker Any (Requires a pre-installed JVM)

Release notes

The SonarScanner CLI is the scanner to use when there is no specific scanner for your build system.

The SonarScanner does not yet officially support ARM architecture. Still, early adopters reported it is working fine. If you encounter problems, don't hesitate to share your experience with us on the SonarQube or SonarCloud Community Forum but keep in mind that there is no support at this time.

The SonarScanners run on code that is checked out. See Verifying the code checkout step of your build.

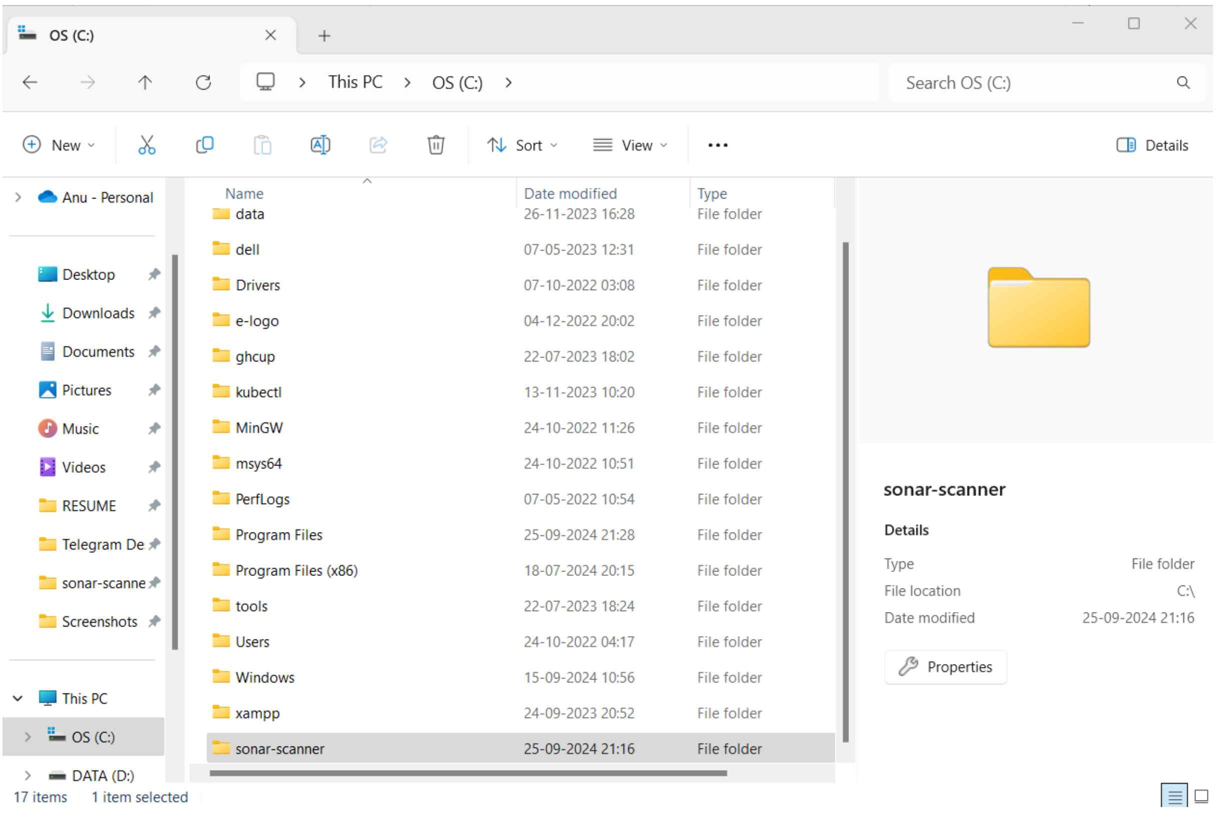
Configuring your project

Create a configuration file in your project's root directory called `sonar-project.properties`.

On this page
Configuring your project
Running SonarScanner CLI from the zip file
Running SonarScanner CLI from the Docker image
Scanning C, C++, or Objective-C projects
Sample projects
Alternatives to sonar-project.properties
Alternate analysis directory
Advanced configuration
Troubleshooting

https://binaries.sonarsource.com/Distribution/sonar-scanner-cli/sonar-scanner-cli-6.2.0.4584-windows-x64.zip?_gl=1*w3exrs*_gcl_au*MTUyMDY5NDYzMS4xNzI3MjczNDM5*_ga*NTAzMjA1MzczLjE3MjcyNzM0Mzk.*_ga_9JZ0GZ5TC6*MTcyNzI3MzQzOS4xLjEuMTcyNzI3OTAxNS4yM...

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



OS (C:)

This PC > OS (C:) >

Search OS (C:)

New Sort View Details

Anu - Personal
Desktop
Downloads
Documents
Pictures
Music
Videos
RESUME
Telegram De
sonar-scanne
Screenshots
This PC
OS (C:)
DATA (D:)

Name	Date modified	Type
data	26-11-2023 16:28	File folder
dell	07-05-2023 12:31	File folder
Drivers	07-10-2022 03:08	File folder
e-logo	04-12-2022 20:02	File folder
ghcup	22-07-2023 18:02	File folder
kubectl	13-11-2023 10:20	File folder
MinGW	24-10-2022 11:26	File folder
msys64	24-10-2022 10:51	File folder
PerfLogs	07-05-2022 10:54	File folder
Program Files	25-09-2024 21:28	File folder
Program Files (x86)	18-07-2024 20:15	File folder
tools	22-07-2023 18:24	File folder
Users	24-10-2022 04:17	File folder
Windows	15-09-2024 10:56	File folder
xampp	24-09-2023 20:52	File folder
sonar-scanner	25-09-2024 21:16	File folder

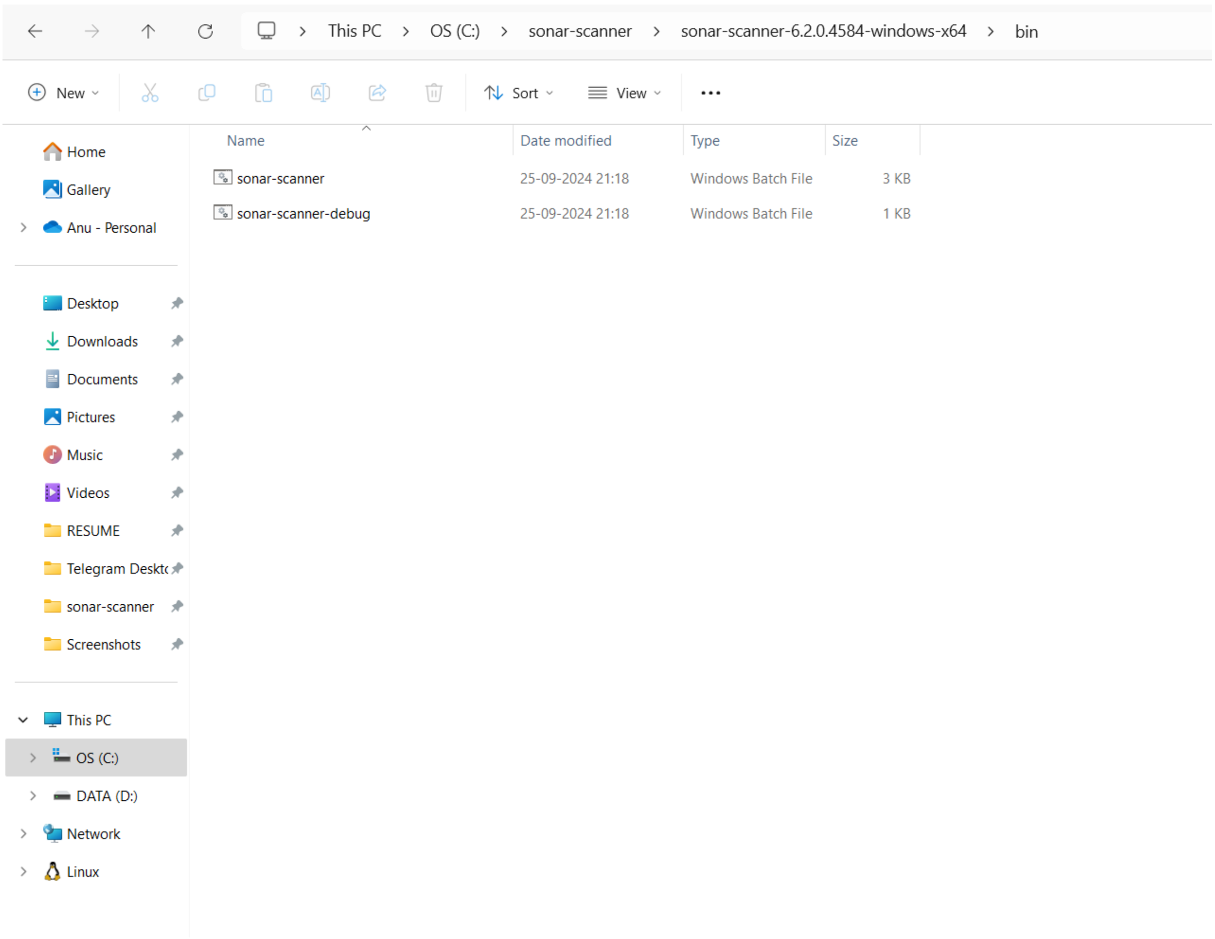
sonar-scanner

Details

Type	File folder
File location	C:\
Date modified	25-09-2024 21:16

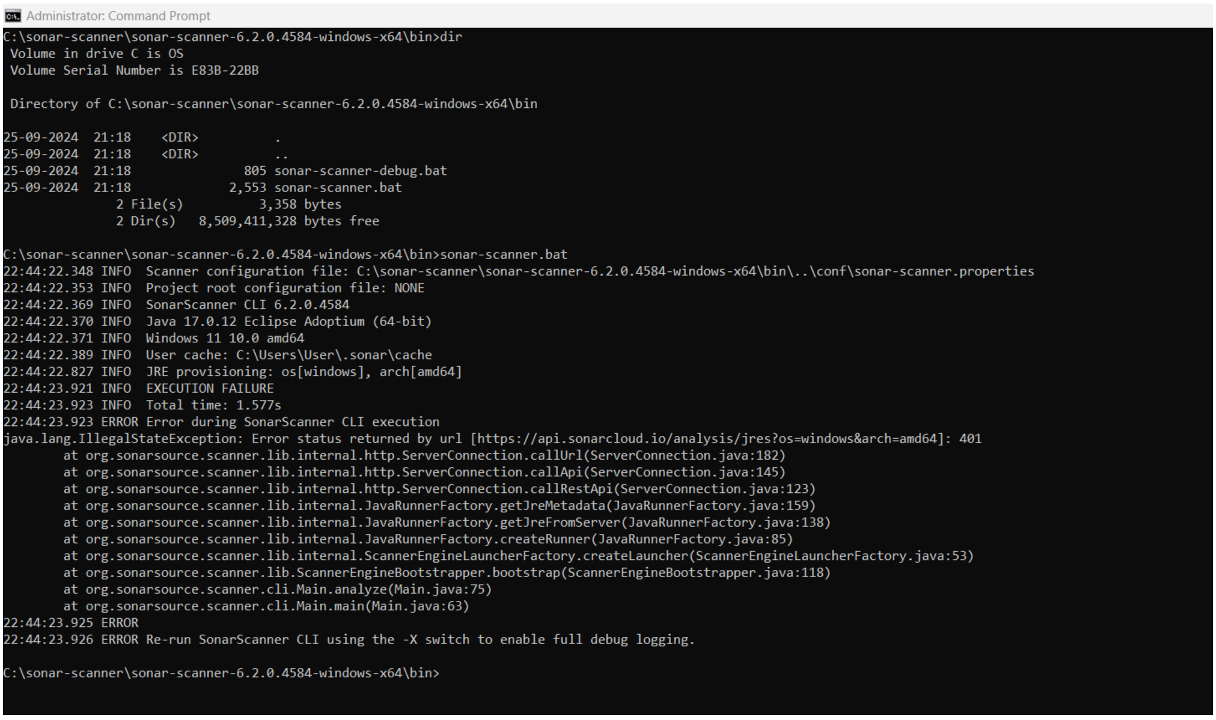
Properties

17 items 1 item selected



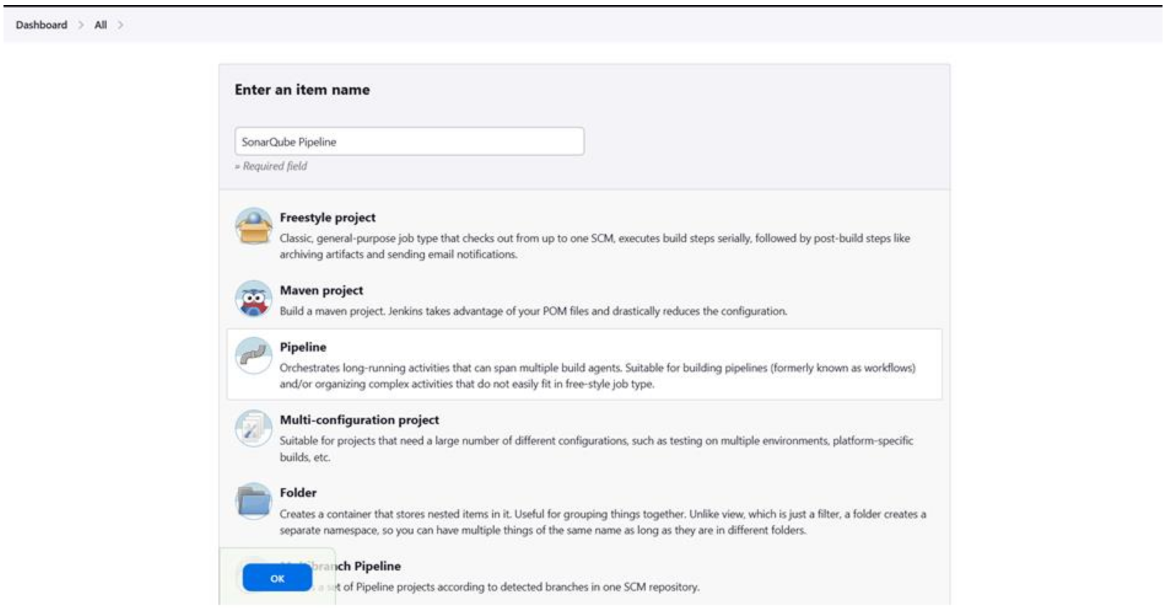
Name	Date modified	Type	Size
sonar-scanner	25-09-2024 21:18	Windows Batch File	3 KB
sonar-scanner-debug	25-09-2024 21:18	Windows Batch File	1 KB

Step 3: Then now open Cmd 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



```
C:\sonar-scanner\sonar-scanner-6.2.0.4584-windows-x64\bin>dir
 Volume in drive C is OS
 Volume Serial Number is E83B-22BB

 Directory of C:\sonar-scanner\sonar-scanner-6.2.0.4584-windows-x64\bin

25-09-2024  21:18    <DIR>          .
25-09-2024  21:18    <DIR>          ..
25-09-2024  21:18               805 sonar-scanner-debug.bat
25-09-2024  21:18             2,553 sonar-scanner.bat
               2 File(s)          3,358 bytes
               2 Dir(s)   8,509,411,328 bytes free

C:\sonar-scanner\sonar-scanner-6.2.0.4584-windows-x64\bin>sonar-scanner.bat
22:44:22.348 INFO  Scanner configuration file: C:\sonar-scanner\sonar-scanner-6.2.0.4584-windows-x64\bin\..\conf\sonar-scanner.properties
22:44:22.353 INFO  Project root configuration file: NONE
22:44:22.369 INFO  SonarScanner CLI 6.2.0.4584
22:44:22.370 INFO  Java 17.0.12 Eclipse Adoptium (64-bit)
22:44:22.371 INFO  Windows 11 10.0 amd64
22:44:22.389 INFO  User cache: C:\Users\User\.sonar\cache
22:44:22.827 INFO  JRE provisioning: os[windows], arch[amd64]
22:44:23.921 INFO  EXECUTION FAILURE
22:44:23.923 INFO  Total time: 1.577s
22:44:23.923 ERROR Error during SonarScanner CLI execution
java.lang.IllegalStateException: Error status returned by url [https://api.sonarcloud.io/analysis/jres?os=windows&arch=amd64]: 401
        at org.sonarsource.scanner.lib.internal.http.ServerConnection.callUrl(ServerConnection.java:182)
        at org.sonarsource.scanner.lib.internal.http.ServerConnection.callApi(ServerConnection.java:145)
        at org.sonarsource.scanner.lib.internal.http.ServerConnection.callRestApi(ServerConnection.java:123)
        at org.sonarsource.scanner.lib.internal.JavaRunnerFactory.getJreMetadata(JavaRunnerFactory.java:159)
        at org.sonarsource.scanner.lib.internal.JavaRunnerFactory.getJreFromServer(JavaRunnerFactory.java:138)
        at org.sonarsource.scanner.lib.internal.JavaRunnerFactory.createRunner(JavaRunnerFactory.java:85)
        at org.sonarsource.scanner.lib.internal.ScannerEngineLauncherFactory.createLauncher(ScannerEngineLauncherFactory.java:53)
        at org.sonarsource.scanner.lib.ScannerEngineBootstrapper.bootstrap(ScannerEngineBootstrapper.java:118)
        at org.sonarsource.scanner.cli.Main.analyze(Main.java:75)
        at org.sonarsource.scanner.cli.Main.main(Main.java:63)
22:44:23.925 ERROR
22:44:23.926 ERROR Re-run SonarScanner CLI using the -X switch to enable full debug logging.

C:\sonar-scanner\sonar-scanner-6.2.0.4584-windows-x64\bin>
```

Step 4: Open Jenkins and create a pipeline and name the pipeline SonarQube Pipeline


Dashboard > All >

Enter an item name

SonarQube 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.

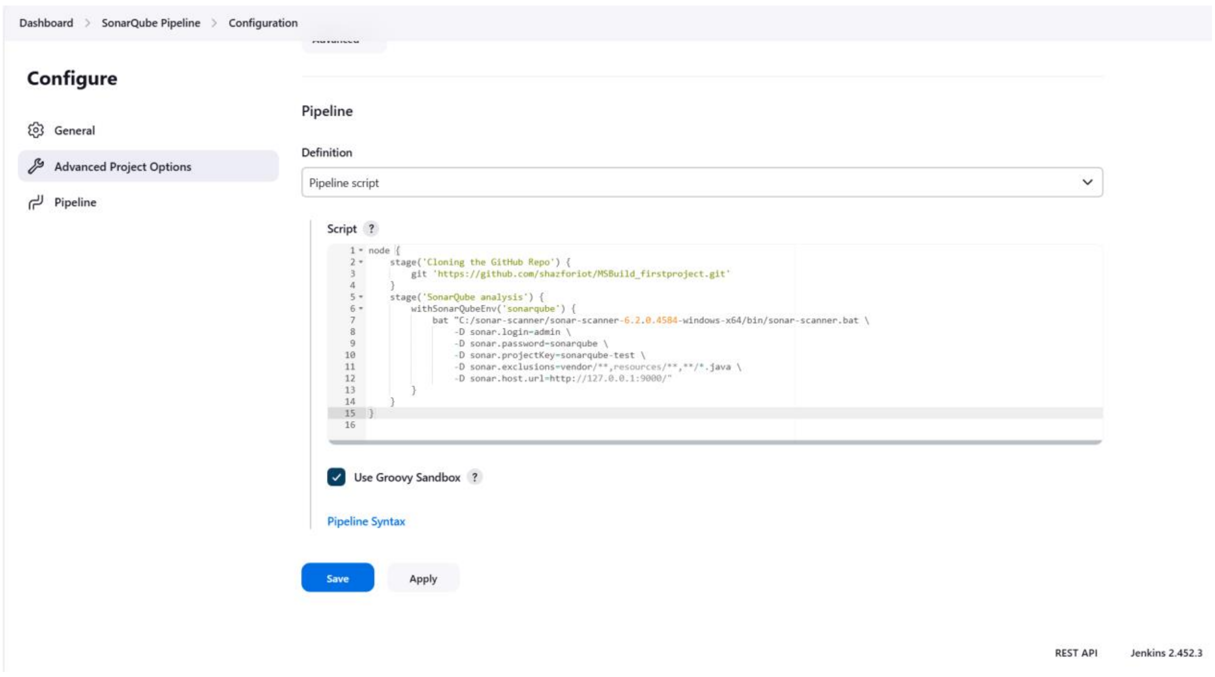
Folder
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.

ranch Pipeline
t 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

Pipeline

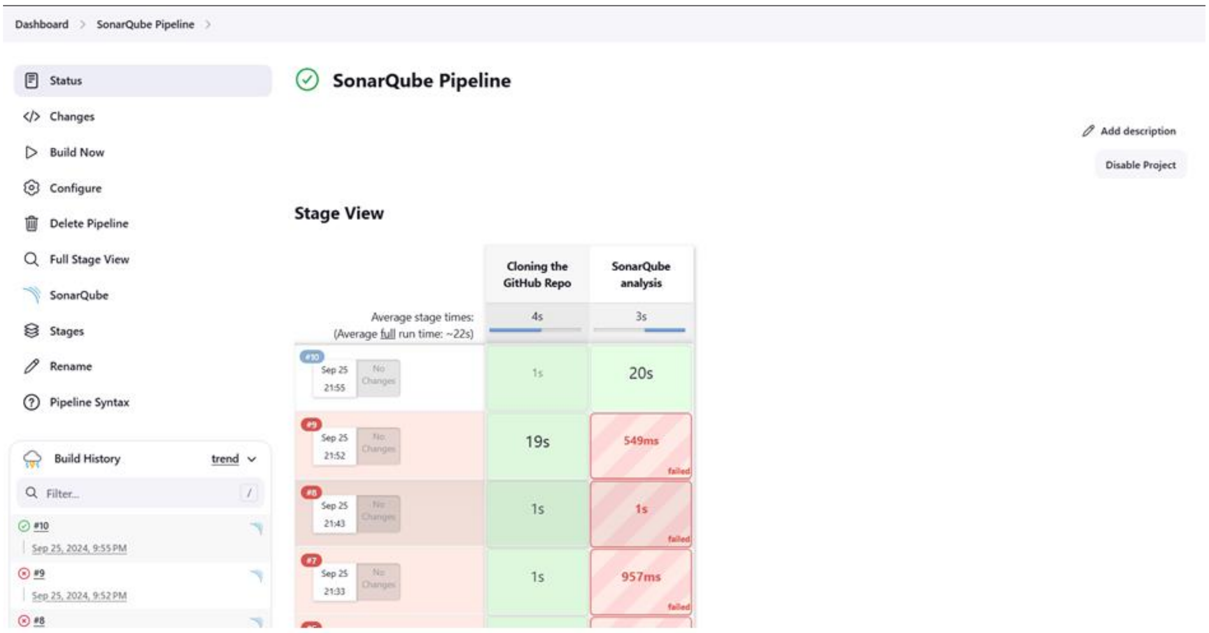
Definition: Pipeline script

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/"
        }
    }
}
```

Use Groovy Sandbox ✓

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



Dashboard > SonarQube Pipeline >

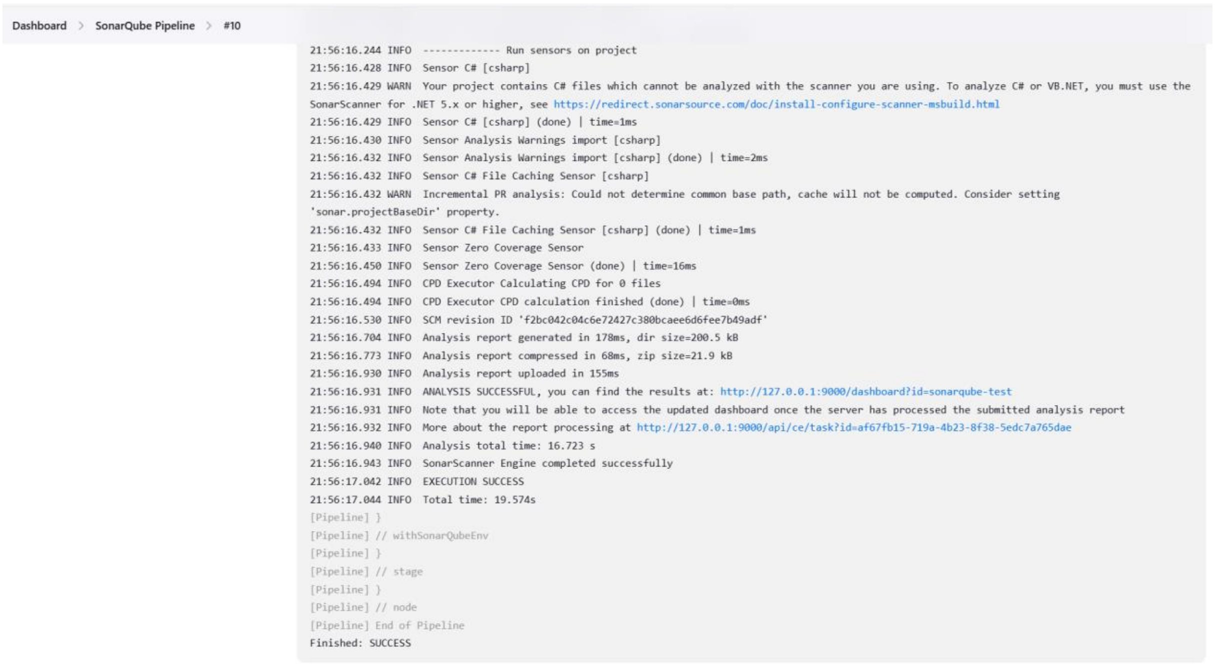
SonarQube Pipeline

Stage View

	Cloning the GitHub Repo	SonarQube analysis
Average stage times: (Average full run time: ~22s)	4s	3s
#10 Sep 25 21:55 No Changes	1s	20s
#9 Sep 25 21:52 No Changes	19s	549ms failed
#8 Sep 25 21:43 No Changes	1s	1s failed
#7 Sep 25 21:33 No Changes	1s	957ms failed

Build History trend

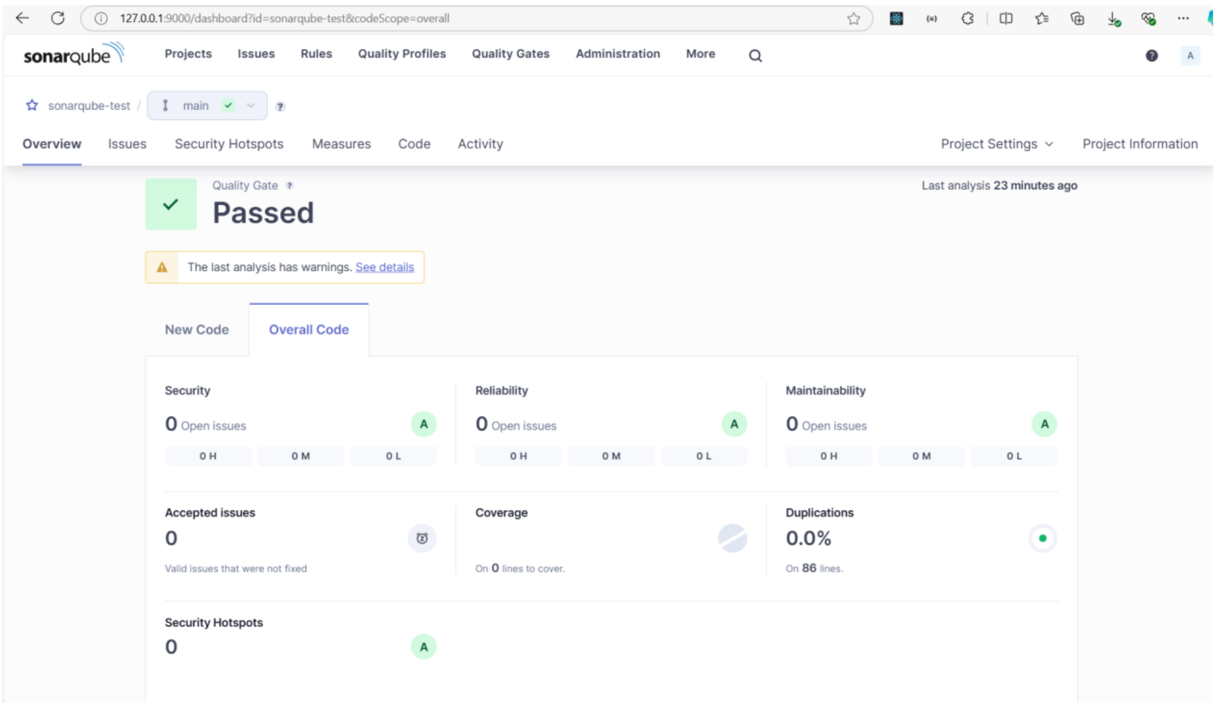
#10 Sep 25, 2024, 9:55 PM
#9 Sep 25, 2024, 9:52 PM
#8



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 'f2bc042c04c6e72427c380bcaee6d6fee7b49adf'
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, ypo can visit http://127.0.0.1:9000/dashboard?id=sonarqube-test to see the result.



Quality Gate

Passed

Last analysis **23 minutes** ago

The last analysis has warnings. See details

Security
0 Open issues A
0 H | 0 M | 0 L

Reliability
0 Open issues A
0 H | 0 M | 0 L

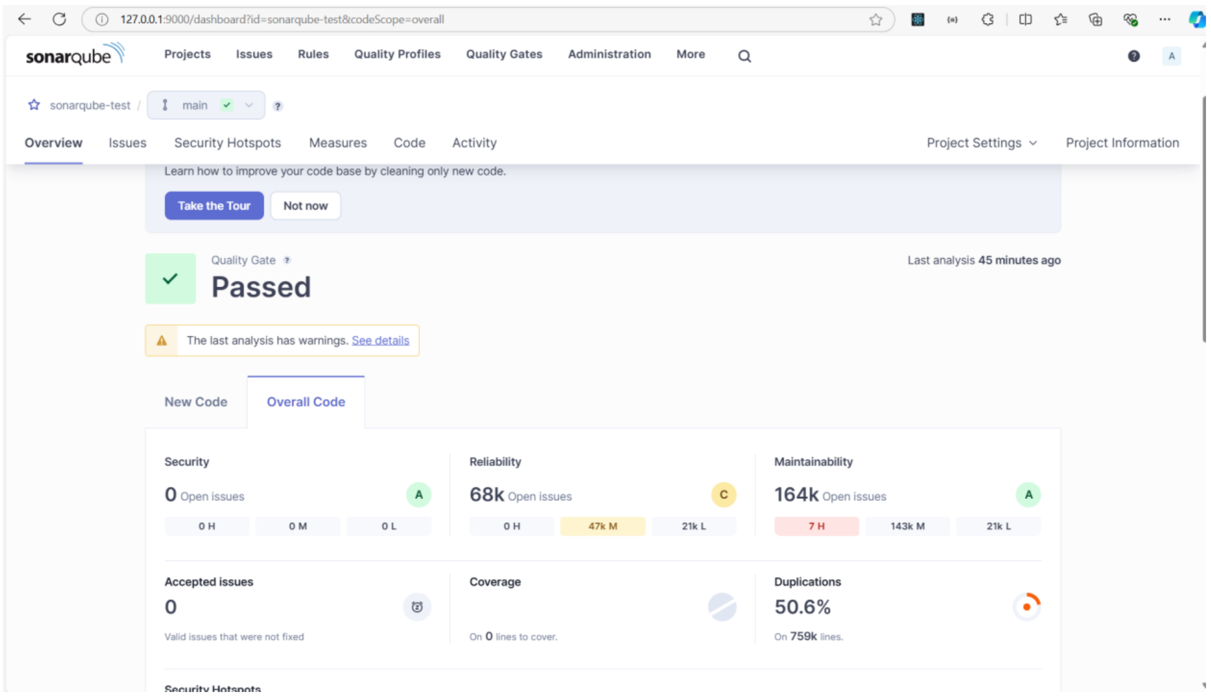
Maintainability
0 Open issues A
0 H | 0 M | 0 L

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



Learn how to improve your code base by cleaning only new code.

Quality Gate

Passed

Last analysis **45 minutes** ago

The last analysis has warnings. See details

Security
0 Open issues A
0 H | 0 M | 0 L

Reliability
68k Open issues C
0 H | 47k M | 21k L

Maintainability
164k Open issues A
7 H | 143k M | 21k L

Accepted issues
0
Valid issues that were not fixed

Coverage
On **0** lines to cover.

Duplications
50.6%
On **759k** lines.

Security Hotspots

Conclusion:

In this experiment, we performed a static analysis of the code to detect bugs, code smells, andsecurity vulnerabilities on our sample Java application.