

Aim: To understand Static Analysis SAST process and learn to integrate Jenkins SAST to SonarQube/GitLab.

Run docker -v command. Use this command to check if docker is installed and running on your system

```
C:\Users\Aditya>docker -v  
Docker version 27.2.0, build 3ab4256
```

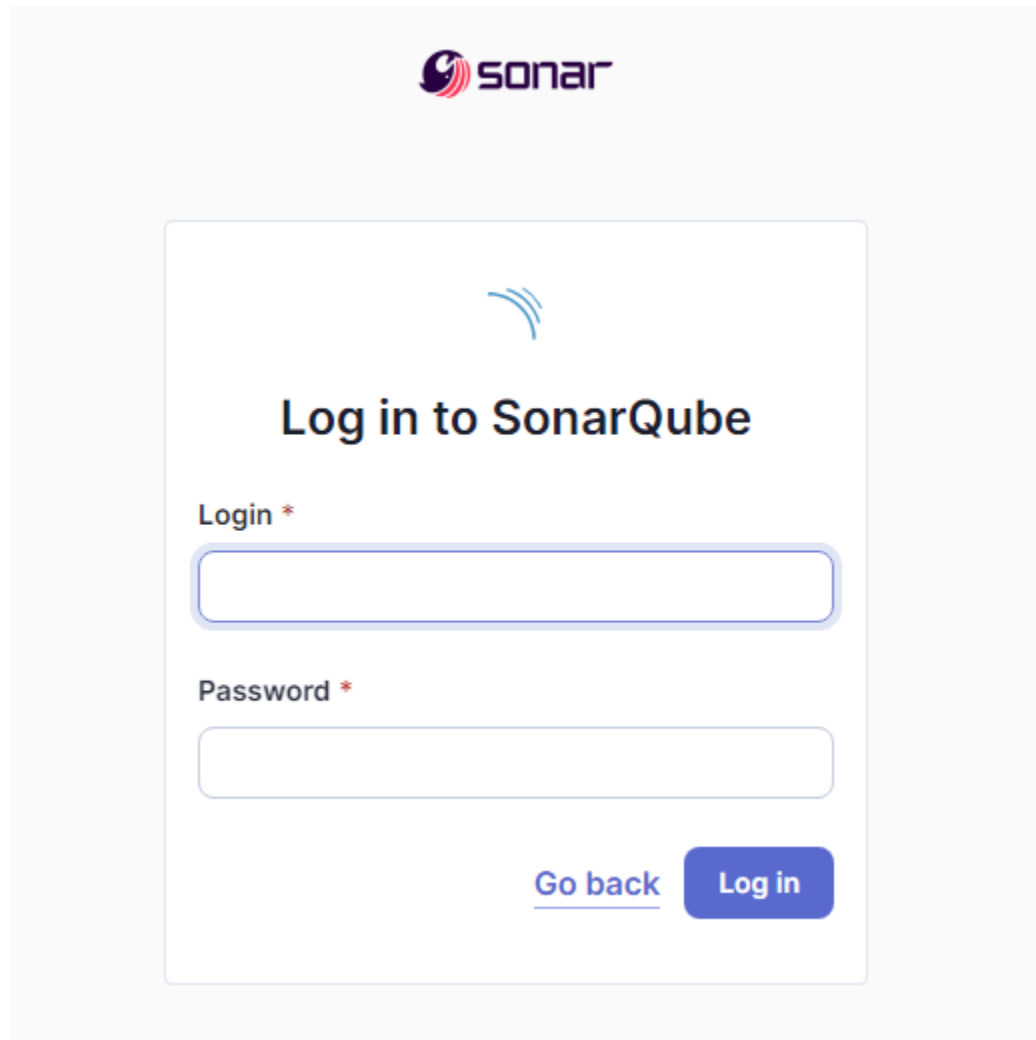
docker pull sonarqube This command helps you to install an image of SonarQube that can be used on the local system without actually installing the SonarQube installer.

```
C:\Users\Aditya>docker pull sonarqube  
Using default tag: latest  
latest: Pulling from library/sonarqube  
4f4fb700ef54: Download complete  
bd819c9b5ead: Download complete  
80338217a4ab: Download complete  
7478e0ac0f23: Download complete  
00a925ab929a: Download complete  
7b87d6fa783d: Download complete  
a5fd5c7e184: Download complete  
7d9a34308537: Download complete  
Digest: sha256:72e9feec71242af83faf65f95a40d5e3bb2822a6c3b2cda8568790f3d31aecde  
Status: Downloaded newer image for sonarqube:latest  
docker.io/library/sonarqube:latest
```


Run SonarQube image: docker run -d --name sonarqube -e SONAR\_ES\_BOOTSTRAP\_CHECKS\_DISABLE=true -p 9000:9000 sonarqube:latest This command will run the SonarQube image that was just installed using docker


```
C:\Users\Aditya>docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest  
102a6e51b67968db1df41bc8c6c3ce068f11506acadbab2d306470aede30908
```

Once the SonarQube image is started, you can go to <http://localhost:9000> to find the SonarQube that has started



The image shows the SonarQube login page. At the top, there is the Sonar logo. Below it, the text "Log in to SonarQube" is displayed. There are two input fields: "Login \*" and "Password \*". Below the "Login \*" field is a blue button labeled "Log in". To the left of the "Log in" button is a link labeled "Go back".






## Log in to SonarQube

Login \*

Password \*

[Go back](#) Log in


After changing the password, you will be directed to this screen. Click on Create a Local Project.


 [Projects](#) [Issues](#) [Rules](#) [Quality Profiles](#) [Quality Gates](#) [Administration](#) [More](#) [Q](#)


## How do you want to create your project?


Do you want to benefit from all of SonarQube's features (like repository import and Pull Request decoration)?  
Create your project from your favorite DevOps platform.

First, you need to set up a DevOps platform configuration.

 Import from Azure DevOps Setup

 Import from Bitbucket Cloud

 Import from GitHub Setup

 Import from GitLab

Are you just testing or have an advanced use-case? Create a local project.

[Create a local project](#)

Give the project a display name and project key

1 of 2

## Create a local project

Project display name \*



Project key \*



Main branch name \*

The name of your project's default branch [Learn More](#) 

[Cancel](#)[Next](#)

Set up the project as required and click on create.

2 of 2

×

## Set up project for Clean as You Code

The new code definition sets which part of your code will be considered new code. This helps you focus attention on the most recent changes to your project, enabling you to follow the Clean as You Code methodology. Learn more: [Defining New Code](#)

### Choose the baseline for new code for this project

☒ Use the global setting

**Previous version**

Any code that has changed since the previous version is considered new code.  
Recommended for projects following regular versions or releases.

☐ Define a specific setting for this project

☐ Previous version

Any code that has changed since the previous version is considered new code.  
Recommended for projects following regular versions or releases.

☐ Number of days

Any code that has changed in the last x days is considered new code. If no action is taken on a new issue after x days, this issue will become part of the overall code.  
Recommended for projects following continuous delivery.

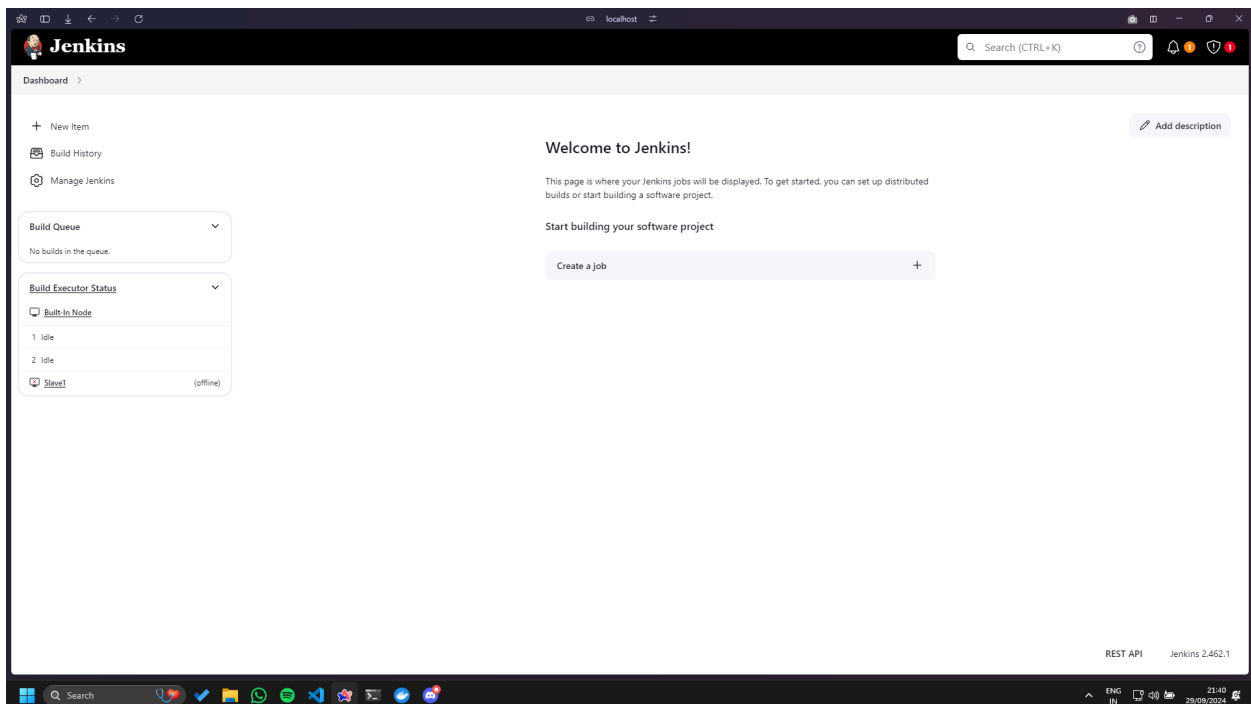
☐ Reference branch

Choose a branch as the baseline for the new code.  
Recommended for projects using feature branches.

Back

Create project

Open Jenkins on whichever port it is installed. (<http://localhost:>).



Go to manage jenkins → Search for Sonarqube Scanner for Jenkins and install it.



Now, go to Manage Jenkins → System. Under Sonarqube servers, add a server. Add server authentication token if needed.

SonarQube installations  
List of SonarQube installations

Name

Server URL

Default is http://localhost:9000

Server authentication token

SonarQube authentication token. Mandatory when anonymous access is disabled.

- none -

+ Add

Advanced

Add SonarQube

Save

Apply

Go to Manage Jenkins → Tools. Go to SonarQube scanner, choose the latest configuration and choose install automatically

SonarQube Scanner installations

Add SonarQube Scanner

SonarQube Scanner

Name

☒ Install automatically

Install from Maven Central

Version

SonarQube Scanner 6.2.0.4584

Add Installer

Add SonarQube Scanner

After configuration, create a New Item → choose a freestyle project.

## New Item

Enter an item name

sonarcube30

Select an item type



### 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.

Use this github repository in Source Code Management.

[https://github.com/shazforiot/MSBuild\\_firstproject](https://github.com/shazforiot/MSBuild_firstproject) It is a sample hello-world project with no vulnerabilities.

Source Code Management

☐ None

☒ Git ?

Repositories ?

Repository URL ?

[https://github.com/shazforiot/MSBuild\\_firstproject](https://github.com/shazforiot/MSBuild_firstproject)

Credentials ?

- none -

+ Add ▾

Advanced ▾

Add Repository

Under Build Steps, enter Sonarqube Scanner, enter these Analysis properties. Mention the SonarQube Project Key, Login, Password, Source path and Host URL.

#### Build Steps

Execute SonarQube Scanner

JDK ?  
JDK to be used for this SonarQube analysis  
(Inherit From Job)

Path to project properties ?

Analysis properties ?  

```
sonar.projectkey=30exp7
sonar.login=admin
sonar.password=Aditya@123
sonar.hosturl=http://localhost:9000
sonar.sources=
```

Additional arguments ?

JVM Options ?

Now, you need to grant the locl user (here admin user) permissions to Execute the Analysis stage on SoanrQube. For this, go to <http://localhost:9000/admin/permissions> and check the 'Execute Analysis' checkbox under Administrator.

sonarqube

Projects Issues Rules Quality Profiles Quality Gates Administration More

Administration

Configuration Security Projects System Marketplace

Global Permissions

Grant and revoke permissions to make changes at the global level. These permissions include editing Quality Profiles, executing analysis, and performing global system administration.

All Users Groups Search for users or groups...

	Administer System ?	Administer ?	Execute Analysis ?	Create ?
<div>sonar-administrators</div> <div>System administrators</div>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> Quality Gates <input checked="" type="checkbox"/> Quality Profiles	<input type="checkbox"/>	<input checked="" type="checkbox"/> Projects
<div>sonar-users</div> <div>Every authenticated user automatically belongs to this group</div>	<input type="checkbox"/>	<input type="checkbox"/> Quality Gates <input type="checkbox"/> Quality Profiles	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> Projects
<div>Anyone DEPRECATED</div> <div>Anybody who browses the application belongs to this group. If authentication is not enforced, assigned permissions also apply to non-authenticated users.</div>	<input type="checkbox"/>	<input type="checkbox"/> Quality Gates <input type="checkbox"/> Quality Profiles	<input type="checkbox"/>	<input type="checkbox"/> Projects
<div>Administrator admin</div>	<input checked="" type="checkbox"/>	<input type="checkbox"/> Quality Gates <input type="checkbox"/> Quality Profiles	<input checked="" type="checkbox"/>	<input type="checkbox"/> Projects


4 of 4 shown


Embedded database should be used for evaluation purposes only  
The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.


SonarQube™ technology is powered by SonarSource SA


Community Edition v10.6 (92116) ACTIVE LGPL v3 Community Documentation Plugins Web API


Go back to jenkins. Go to the job you had just built and click on Build Now.


 Status


 Changes


 Workspace

 Build Now


 Configure

 Delete Project

 SonarQube

 Rename

# sonarcube30



SonarQube

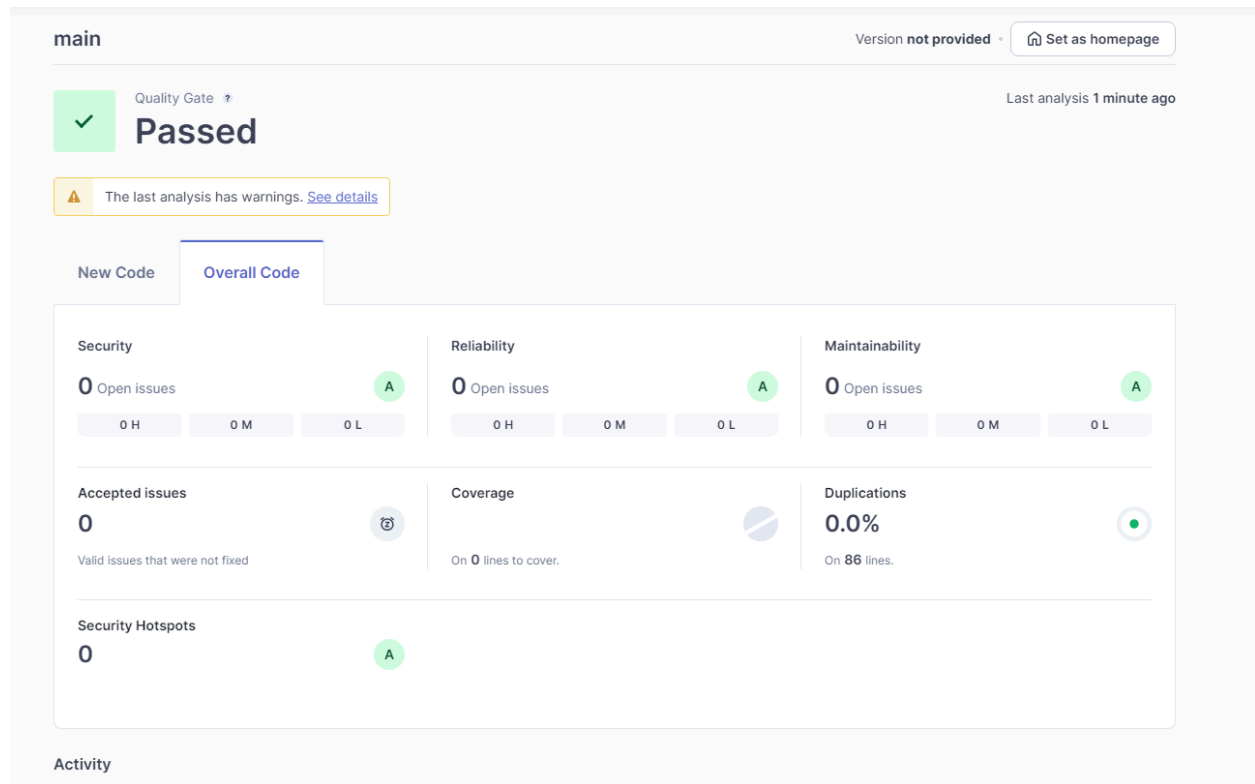
## Permalinks

Check the console output

```
22:04:00.599 INFO SonarScanner Engine completed successfully
22:04:00.665 INFO EXECUTION SUCCESS
22:04:00.667 INFO Total time: 16.886s
Finished: SUCCESS
```



Once the build is complete, go back to SonarQube and check the project linked.



Conclusion: In this experiment, we have learned how to perform Jenkins SAST using SonarQube. For this, we used a docker image of SonarQube so as to not install it locally on our system. After installing the required configurations on Jenkins, using a coe from a gihub repository, we analyze its code using SonarQube. Once we build the project, we can see that SonarQube project displays that the code has no errors.