

Lab Exercise 19

Setting up Snyc for SAST in Jenkins

Objective: To demonstrate the setup of the Snyc plugin in Jenkins for Static Application Security Testing (SAST), to automatically detect vulnerabilities in their codebase during development, thereby enhancing application security before deployment

Tools required: Snyc

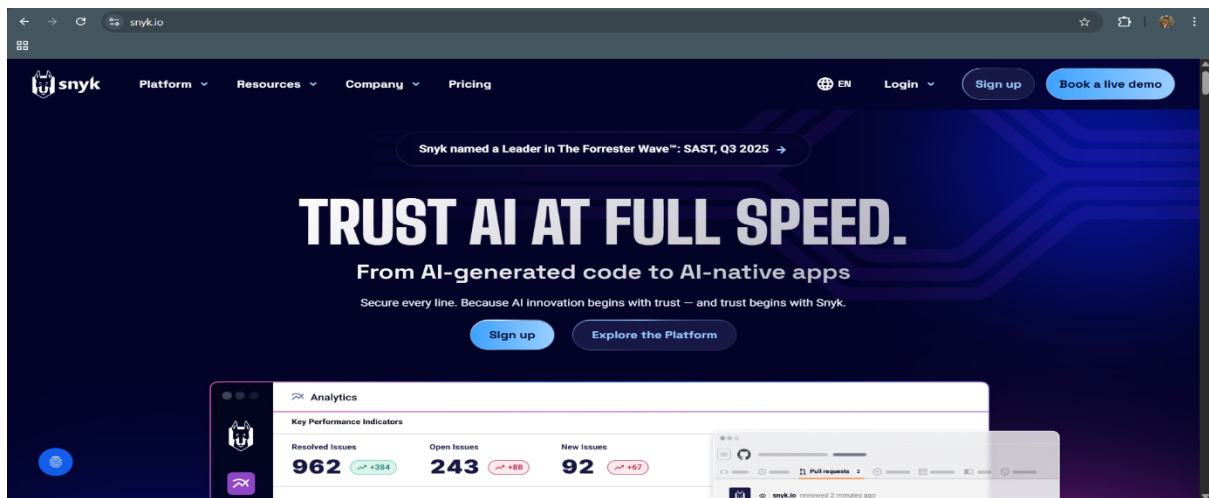
Prerequisites: None

Steps to be followed:

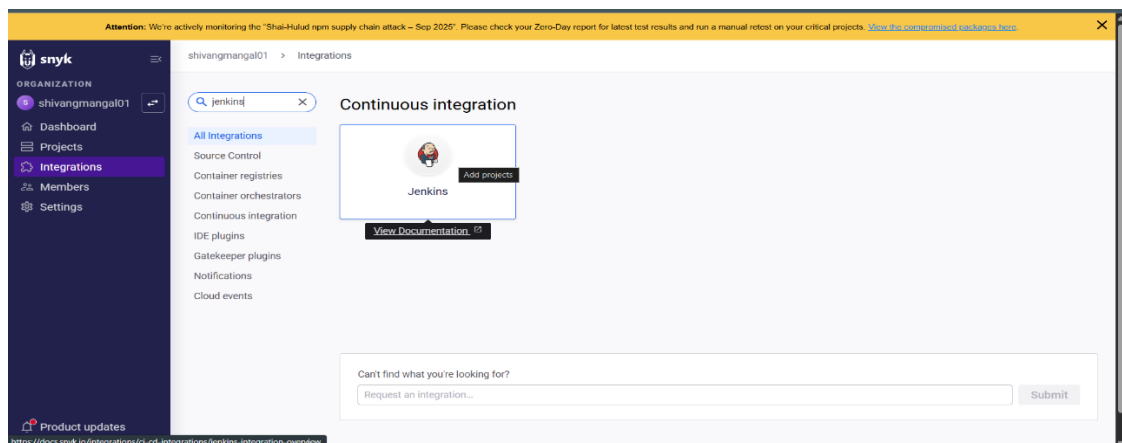
1. Configure Snyc as a SAST scan tool
2. Create and configure a Jenkins job for Snyc integration
3. Manage Snyc API and Jenkins credentials
4. Configure the Jenkins job for scanning

Step 1: Configure Snyc as a SAST scan tool

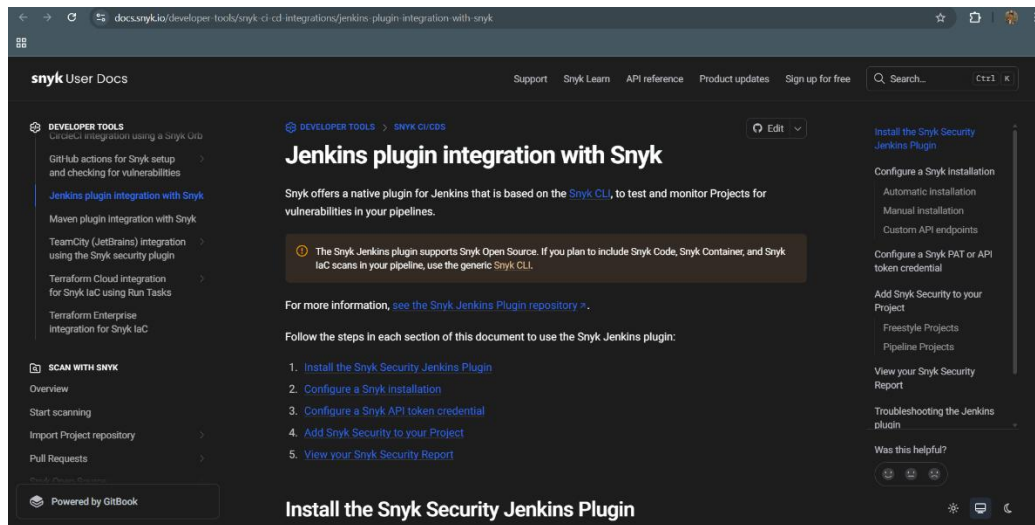
1.1 Visit <https://snyk.io/>, sign up for a new Snyc account, and log in



1.2 Navigate to Integrations and select Jenkins

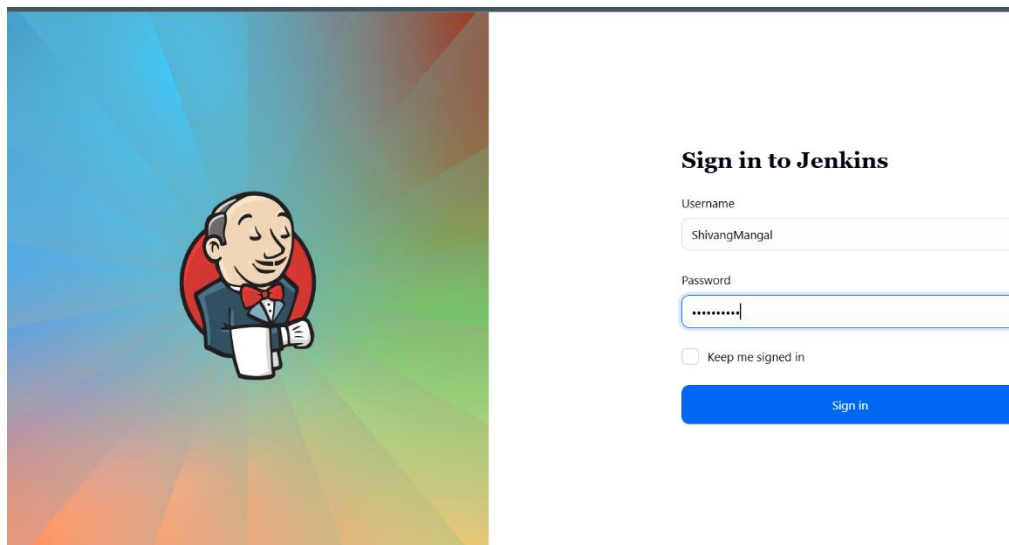


This will direct you to the documentation for integrating Snky with Jenkins.



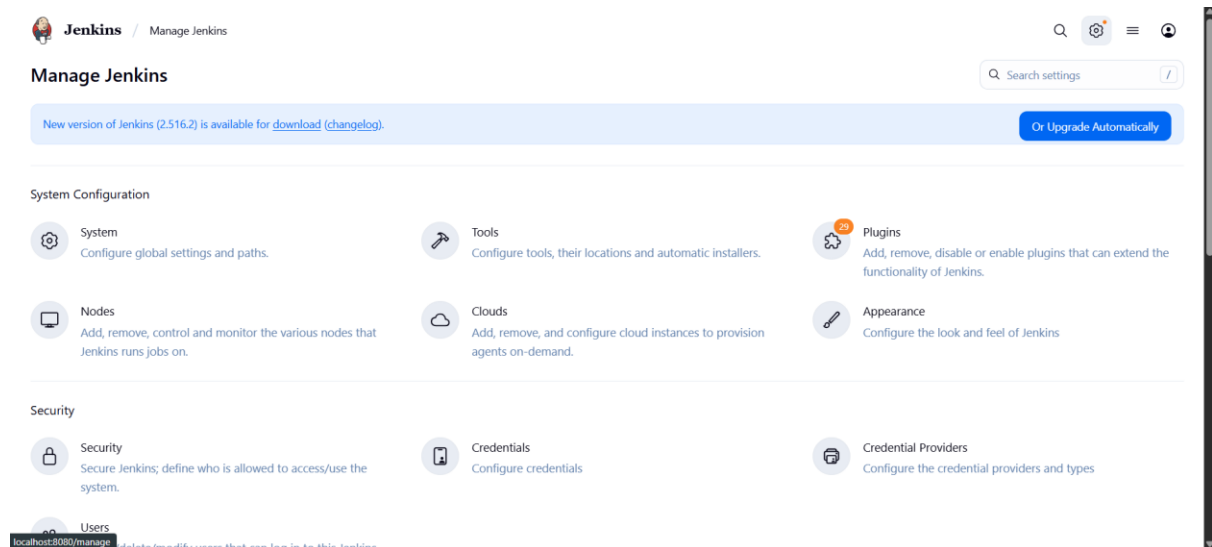
Step 2: Create and configure a Jenkins job for Snky integration

2.1 Open Jenkins and log in to the Jenkins account:



Note: The credentials for accessing Jenkins in the lab are Username: **admin** and Password: **admin**.

2.2 To install the Snky plugin, navigate to Manage Jenkins and click Available Plugins, search for Snky Security plugin, and then click Install



The Jenkins Manage Jenkins page shows a notification for a new version (2.516.2) and a list of configuration categories. The categories include System Configuration (System, Tools, Plugins, Nodes, Clouds, Appearance) and Security (Security, Credentials, Credential Providers, Users). The Users category is highlighted with a red box and a tooltip that reads 'delete/delete users that can log in to this Jenkins'.

Jenkins / Manage Jenkins

Search settings

New version of Jenkins (2.516.2) is available for [download](#) ([changelog](#)). [Or Upgrade Automatically](#)

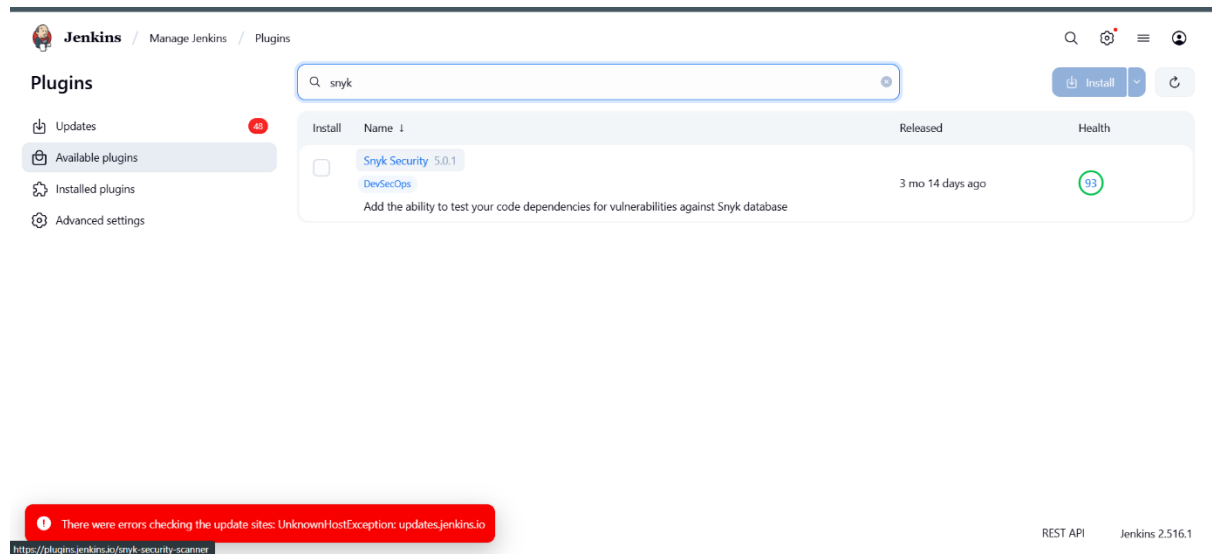
System Configuration

- System**: Configure global settings and paths.
- Tools**: Configure tools, their locations and automatic installers.
- Plugins**: Add, remove, disable or enable plugins that can extend the functionality of Jenkins.
- Nodes**: Add, remove, control and monitor the various nodes that Jenkins runs jobs on.
- Clouds**: Add, remove, and configure cloud instances to provision agents on-demand.
- Appearance**: Configure the look and feel of Jenkins

Security

- Security**: Secure Jenkins; define who is allowed to access/use the system.
- Credentials**: Configure credentials
- Credential Providers**: Configure the credential providers and types
- Users**: Create/delete/users that can log in to this Jenkins.

localhost:8080/manage delete/delete users that can log in to this Jenkins



The Jenkins Manage Jenkins Plugins page shows a search bar with 'snyk' and a table of available plugins. The table has columns for Install, Name, Released, and Health. The 'Snyk Security' plugin is listed with version 5.0.1, released 3 months and 14 days ago, and a health score of 93. The 'DevSecOps' plugin is also listed. A red error message is displayed at the bottom: 'There were errors checking the update sites: UnknownHostException: updates.jenkins.io'.

Jenkins / Manage Jenkins / Plugins

Search: snyk

[Install](#) [Refresh](#)

Install	Name	Released	Health
<input type="checkbox"/>	Snyk Security 5.0.1 DevSecOps Add the ability to test your code dependencies for vulnerabilities against Snyk database	3 mo 14 days ago	93

Updates (40)

Available plugins

Installed plugins

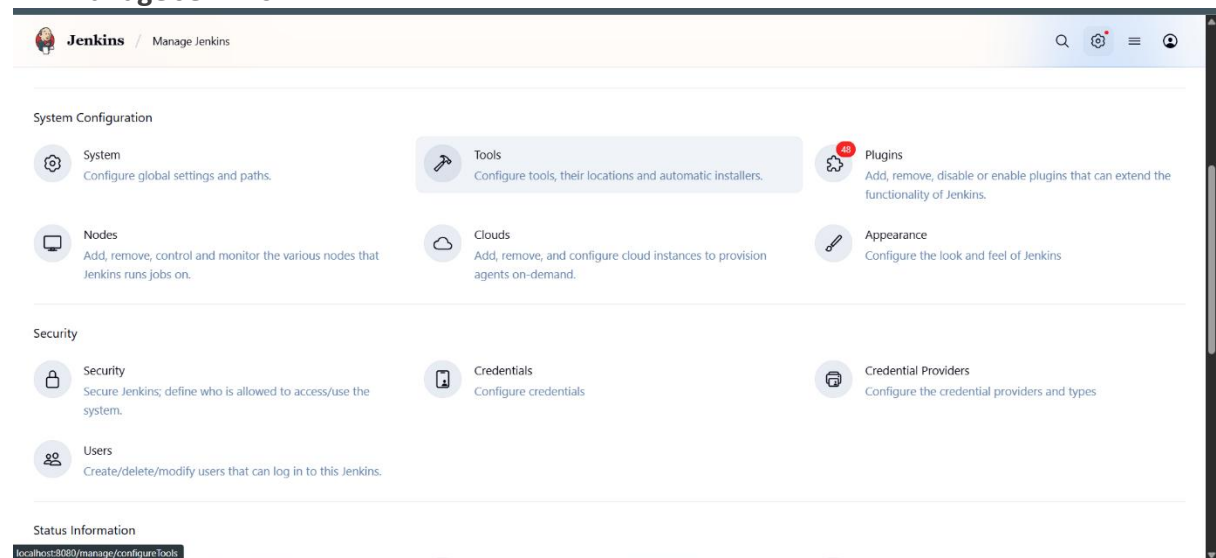
Advanced settings

There were errors checking the update sites: UnknownHostException: updates.jenkins.io

https://plugins.jenkins.io/snyk-security-scanner

REST API Jenkins 2.516.1

2.3 To configure Maven and Snyk in the Global Tool Configuration, click on Tools inside Manage Jenkins



The Jenkins Manage Jenkins Tools page shows the 'Tools' configuration section. The 'Tools' category is highlighted with a red box. The 'Tools' section includes a list of tools and their locations. The 'Tools' section is highlighted with a red box and a tooltip that reads 'Configure tools, their locations and automatic installers'.

Jenkins / Manage Jenkins

Search

System Configuration

- System**: Configure global settings and paths.
- Tools**: Configure tools, their locations and automatic installers.
- Plugins**: Add, remove, disable or enable plugins that can extend the functionality of Jenkins.
- Nodes**: Add, remove, control and monitor the various nodes that Jenkins runs jobs on.
- Clouds**: Add, remove, and configure cloud instances to provision agents on-demand.
- Appearance**: Configure the look and feel of Jenkins

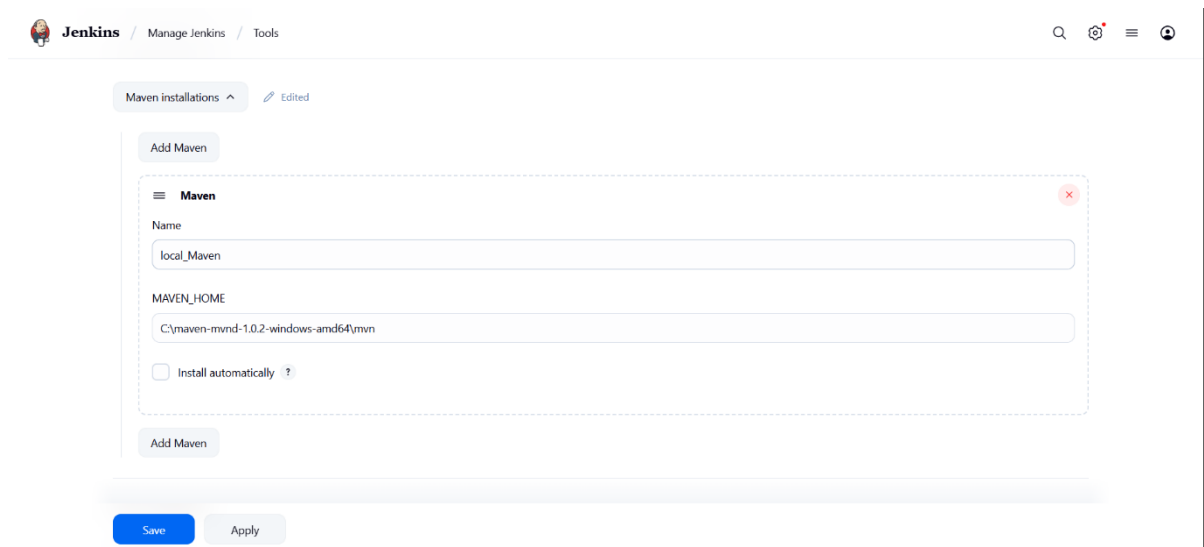
Security

- Security**: Secure Jenkins; define who is allowed to access/use the system.
- Credentials**: Configure credentials
- Credential Providers**: Configure the credential providers and types
- Users**: Create/delete/users that can log in to this Jenkins.

Status Information

localhost:8080/manage/configureTools

2.4 To add Maven, click on Add Maven under Maven installations and enter Maven as the Name

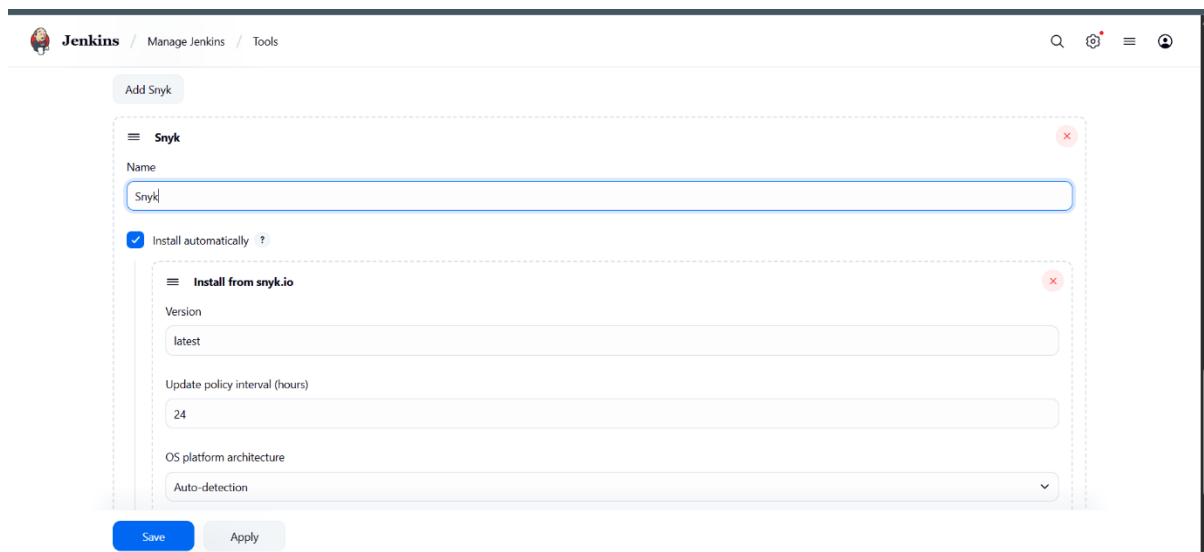


The screenshot shows the Jenkins 'Manage Jenkins' page, specifically the 'Tools' tab. Under the 'Maven installations' section, there is an 'Add Maven' button. Below it, a form is displayed with the following fields:

- Name:** local_Maven
- MAVEN_HOME:** C:\maven-mvnd-1.0.2-windows-amd64\mvn
- Install automatically:** ☐

At the bottom of the form, there are 'Save' and 'Apply' buttons.

2.5 To add Snky, click on Add Snky under Snky Installations, add Name as Snky, and click on the Save button




The screenshot shows the Jenkins 'Manage Jenkins' page, specifically the 'Tools' tab. Under the 'Snky installations' section, there is an 'Add Snky' button. Below it, a form is displayed with the following fields:

- Name:** Snky
- Install automatically:** ☒
- Install from snyk.io:**
 - Version:** latest
 - Update policy interval (hours):** 24
 - OS platform architecture:** Auto-detection

At the bottom of the form, there are 'Save' and 'Apply' buttons.

Step 3: Manage Snky API and Jenkins credentials

3.1 To retrieve your Snky API token, go to Account Settings in your Snky account, click on Click to show under the Auth Token key field, and copy the token for further reference



ORGANIZATION

shivangmangal01

Dashboard

Projects

Integrations

Members

Settings

Shivang

shivangmangal01@gmail.com

Account settings

Notification preferences

Share with a friend

Log out

https://app.snyk.io/account

shivangmangal01 > Dashboard

Start securing your code

Connect your code
Connect Snyk to your code to fix issues and vulnerabilities.

[Choose integration](#)

☐ Add and scan your first project
Import your code to see how Snyk surfaces issues, problematic dependencies, and vulnerabilities.

Invite team members


Collaborate on projects and build secure applications together

[Copy invite link](#)

Use Snyk in the command line

Learn how to install our command line tool to scan your code locally

[Learn more](#)



ORGANIZATION

shivangmangal01

Dashboard

Projects

Integrations

Members

Settings

Product updates

Help

Shivang

https://app.snyk.io/account

Account > General

Account settings

General

Authorized Snyk Apps

Notifications

Share with a friend

Auth Token

Use this token to authenticate the Snyk CLI and in CI/CD pipelines. Learn more about authenticating CLI in our docs.

KEY	CREATED	
click to show	25 September 2025, 16:23:29	Revoke & Regenerate

Authorized Applications


List of applications you have authorized

No applications

Preferred Organization

Choose which organization you are taken to when logging into the site.

shivangmangal01



ORGANIZATION

shivangmangal01

Dashboard

Projects

Integrations

Members

Settings

Product updates

Help

Shivang

https://app.snyk.io/account

Account > General

Account settings

General

Authorized Snyk Apps

Notifications

Share with a friend

Auth Token

Use this token to authenticate the Snyk CLI and in CI/CD pipelines. Learn more about authenticating CLI in our docs.

KEY	CREATED	
ea9e683d-179e-4efc-8679-7978320cb23b	25 September 2025, 16:23:29	Revoke & Regenerate

Authorized Applications

List of applications you have authorized

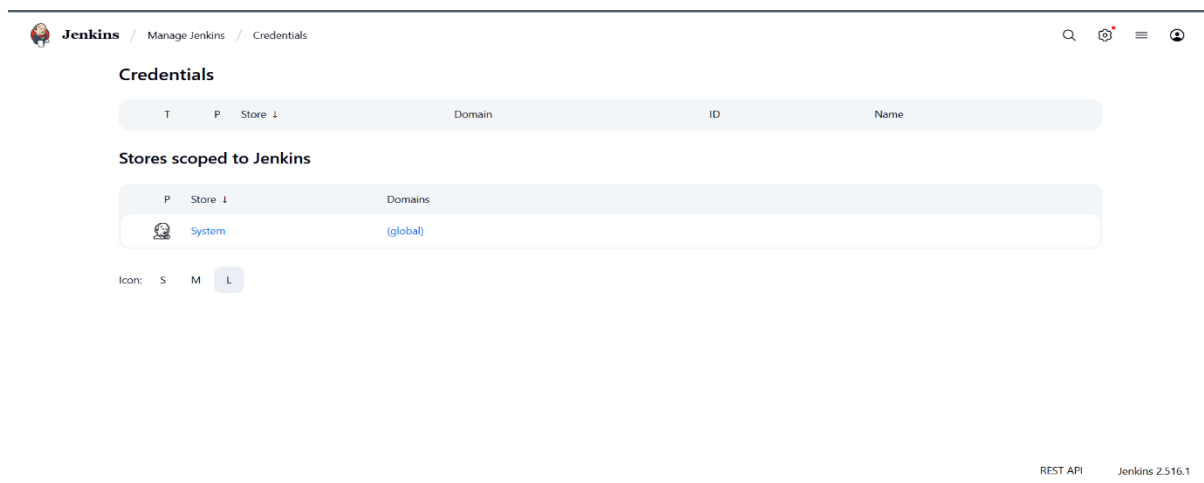
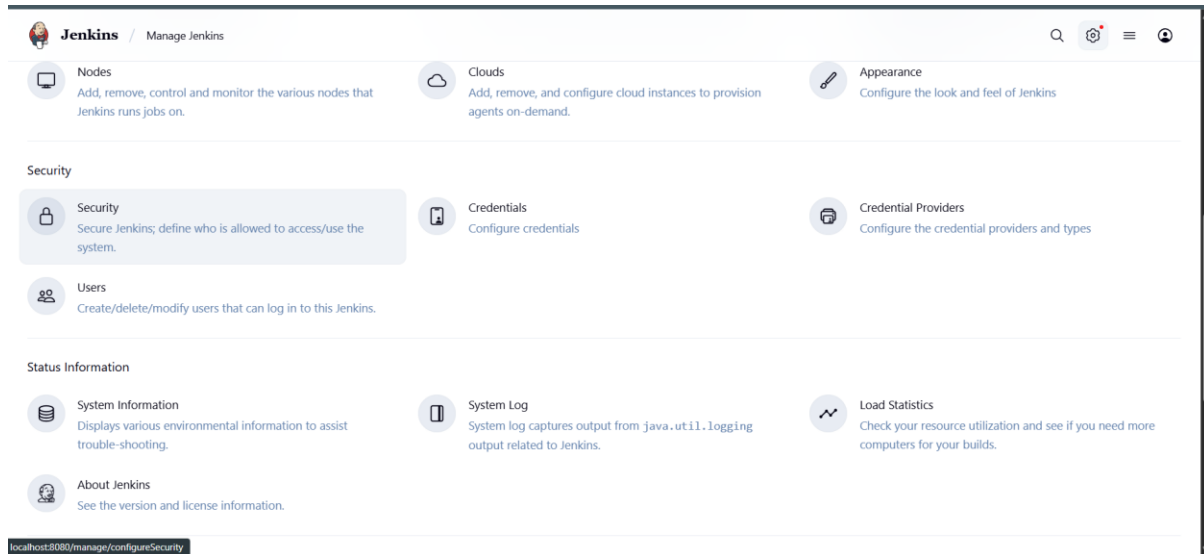
No applications

Preferred Organization

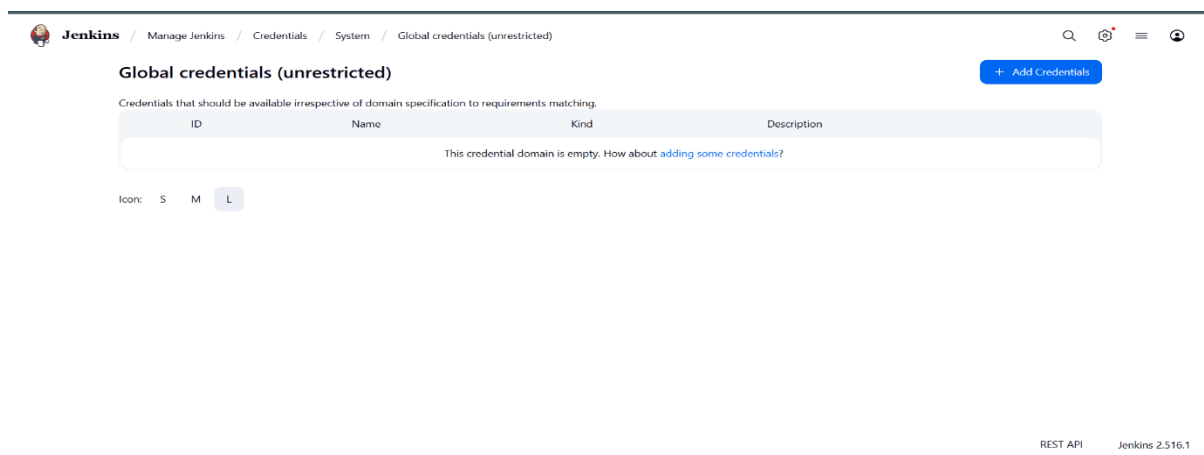
Choose which organization you are taken to when logging into the site.

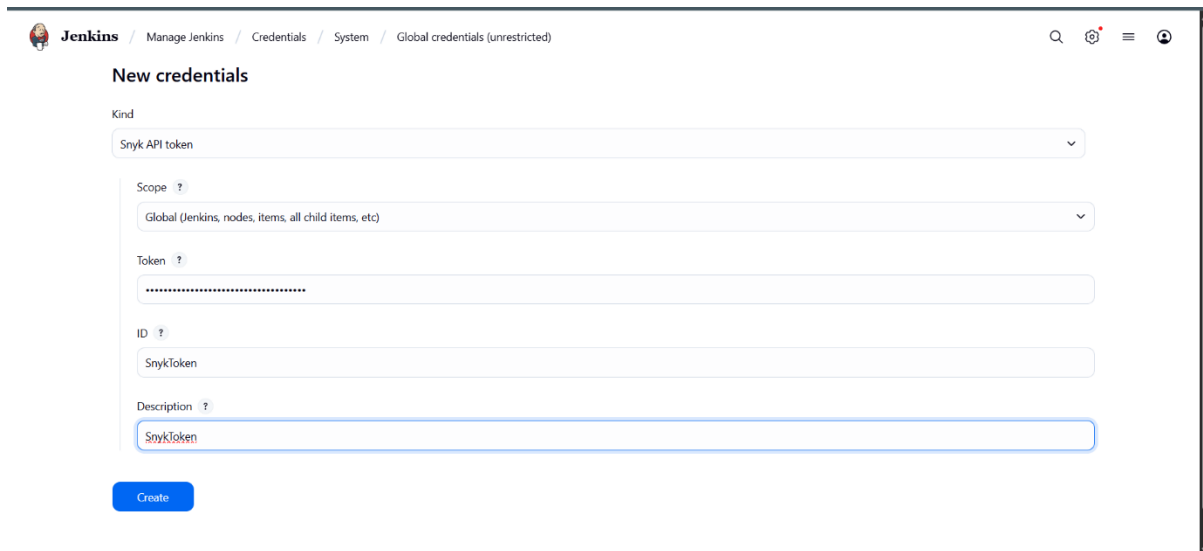
shivangmangal01

3.2 In the Jenkins interface, go to Manage Jenkins, select Security, then choose Credentials and select global to add global credentials



3.3 Click on Add Credentials, select the Snyk API token from the Kind field, paste the copied token from step 3.1 into the Token field, and then click the Create button

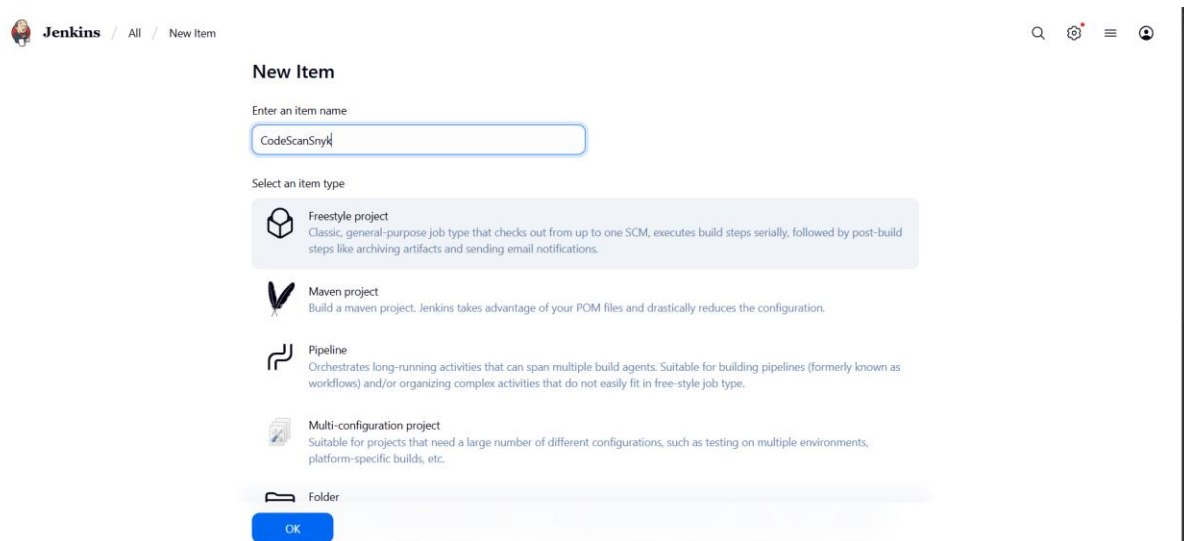




The screenshot shows the Jenkins 'New credentials' page. The breadcrumb trail is 'Jenkins / Manage Jenkins / Credentials / System / Global credentials (unrestricted)'. The page title is 'New credentials'. The 'Kind' dropdown is set to 'Snyk API token'. The 'Scope' dropdown is set to 'Global (Jenkins, nodes, items, all child items, etc)'. The 'Token' field is masked with dots. The 'ID' field is set to 'SnykToken'. The 'Description' field is set to 'SnykToken'. A blue 'Create' button is at the bottom left.

Step 4: Configure the Jenkins job for scanning

4.1 To create a new Jenkins job, click on New Item, enter the item name as CodeScanSnyk, select Freestyle project, and then click OK



The screenshot shows the Jenkins 'New Item' page. The breadcrumb trail is 'Jenkins / All / New Item'. The page title is 'New Item'. The 'Enter an item name' field contains 'CodeScanSnyk'. The 'Select an item type' section shows four options: 'Freestyle project' (selected), 'Maven project', 'Pipeline', and 'Multi-configuration project'. Below these is a 'Folder' option. A blue 'OK' button is at the bottom left.

4.2 After creating a job, go to Source Code Management and enter the GitHub repository URL. Then, under Build Steps, add the build step Invoke Snyk Security task with the name SnykToken. Finally, click the Save button to create the build.

Use GitHub Repo: <https://github.com/hkshitesh/Secure-Coding.git>

The image displays three sequential screenshots of the Jenkins 'Configure' page for the 'CodeScanSnyk' plugin. The first screenshot shows the 'Source Code Management' tab with 'Git' selected as the provider. The repository URL is set to 'https://github.com/hkshitesh/Secure-Coding.git'. The second screenshot shows the 'Environment' tab with a dropdown menu open for 'Add build step', highlighting 'Invoke Snyk Security task'. The third screenshot shows the 'Build Steps' tab with configuration options for when to fail the build and the Snyk API token.

Screenshot 1: Source Code Management

- General
- Source Code Management
- Triggers
- Environment
- Build Steps
- Post-build Actions

Connect and manage your code repository to automatically pull the latest code for your builds.

☐ None

☒ Git

Repositories

Repository URL:

Credentials:

+ Add

Advanced

Name:

Save Apply

Screenshot 2: Environment

- General
- Source Code Management
- Triggers
- Environment
- Build Steps
- Post-build Actions

☐ Add timestamps to the console output.

☐ Inspect build log for published build scans

☐ Terminate a build if it's stuck

☐ With Ant

Build Steps

Automate your build process with ordered tasks like code compilation, testing, and deployment.

Add build step

Filter

- Execute Windows batch command
- Execute shell
- Invoke Ant
- Invoke Gradle script
- Invoke Snyk Security task
- Invoke top-level Maven targets
- Run with timeout
- Set build status to "pending" on GitHub commit

REST API Jenkins 2.516.1

Screenshot 3: Build Steps

- General
- Source Code Management
- Triggers
- Environment
- Build Steps
- Post-build Actions

When issues are found

☒ Fail the build, if severity at or above Let the build continue

☒ Fail the build if errors occur

☒ Monitor project on build

Snyk API token

+ Add

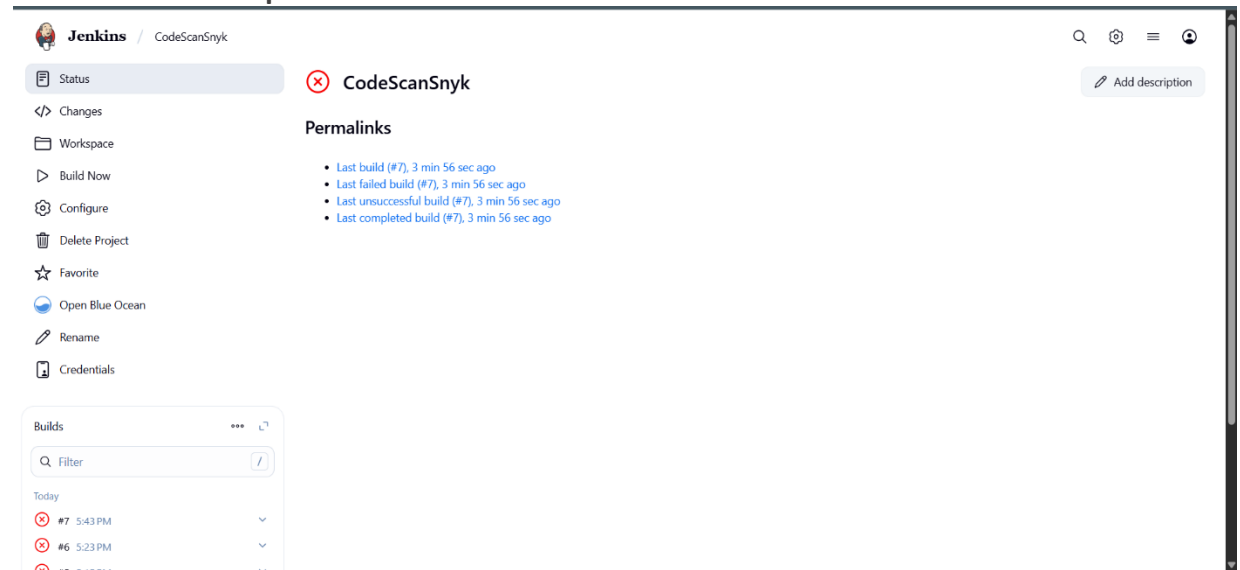
Target file

Organisation

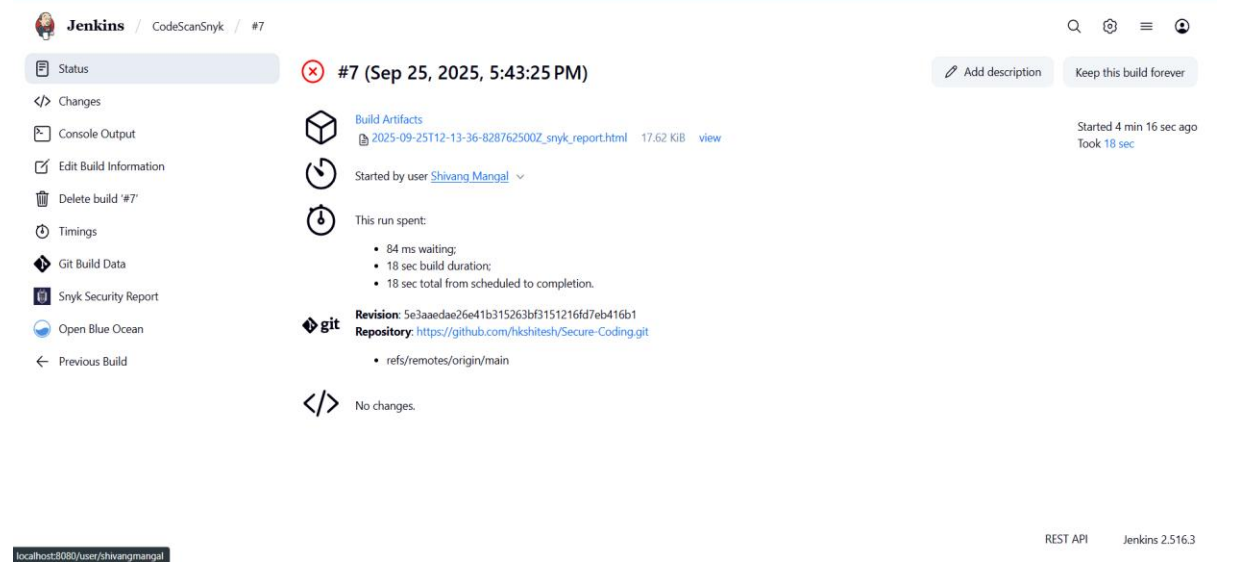
Save Apply

Note: For GitHub repository URL, use <https://github.com/hkshitesh/Secure-Coding.git>

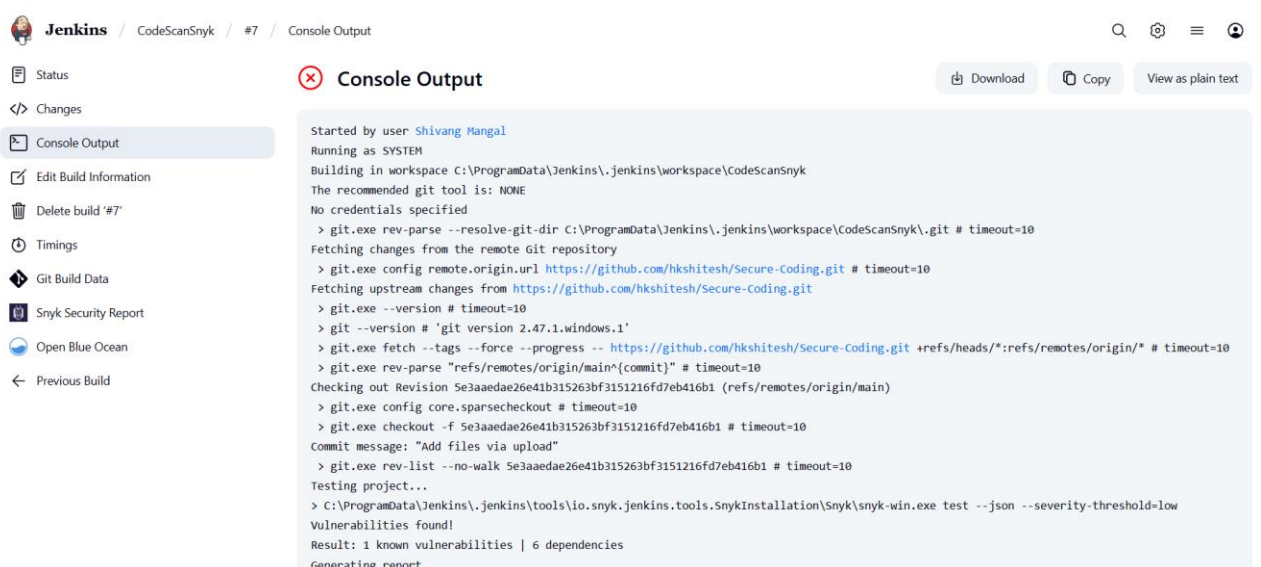
4.3 To check the build status, click on the build link under Permalinks. After that, click on Console Output



The screenshot shows the Jenkins interface for the 'CodeScanSnyk' build. The left sidebar contains a list of actions: Status, Changes, Workspace, Build Now, Configure, Delete Project, Favorite, Open Blue Ocean, Rename, and Credentials. The main area displays the build status as 'CodeScanSnyk' with a red 'X' icon. Below this, there is a 'Permalinks' section with a list of links for the last build, last failed build, last unsuccessful build, and last completed build, all dated '3 min 56 sec ago'. A 'Builds' table on the left shows two builds: #7 (5:43 PM) and #6 (5:23 PM), both with red 'X' icons.

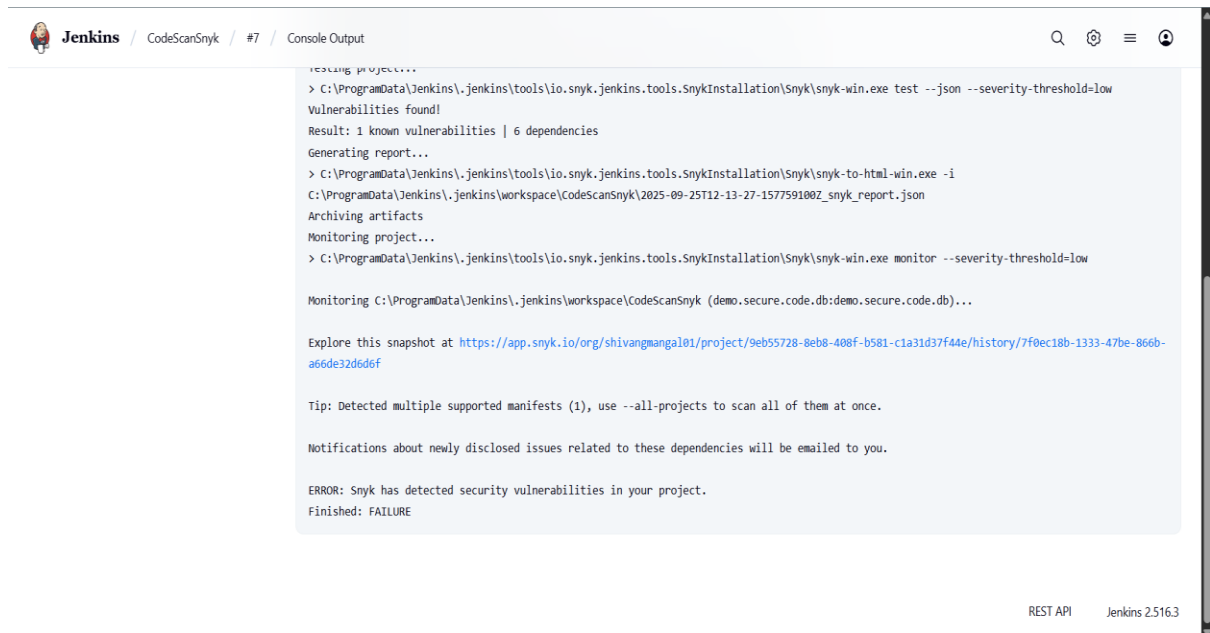


The screenshot shows the details for build #7 (Sep 25, 2025, 5:43:25 PM). The left sidebar includes actions like Status, Changes, Console Output, Edit Build Information, Delete build '#7', Timings, Git Build Data, Snyk Security Report, Open Blue Ocean, and Previous Build. The main area shows the build status as '#7 (Sep 25, 2025, 5:43:25 PM)' with a red 'X' icon. It includes a 'Build Artifacts' section with a link to '2025-09-25T12-13-36-828762500Z_snyk_report.html' (17.62 KB). The 'Started by user' is 'Shivang Mangal'. The 'This run spent' section shows: 84 ms waiting, 18 sec build duration, and 18 sec total from scheduled to completion. The 'Revision' is '5e3aaedae26e41b315263bf3151216fd7eb416b1' and the 'Repository' is 'https://github.com/hkshitesh/Secure-Coding.git'. The 'git' section shows 'refs/remotes/origin/main'. The 'No changes' section is also visible. The bottom right corner shows 'REST API' and 'Jenkins 2.516.3'.



The screenshot shows the console output for build #7. The left sidebar includes actions like Status, Changes, Console Output, Edit Build Information, Delete build '#7', Timings, Git Build Data, Snyk Security Report, Open Blue Ocean, and Previous Build. The main area shows the console output with a red 'X' icon. The output text is as follows:

```
started by user Shivang Mangal
Running as SYSTEM
Building in workspace C:\ProgramData\jenkins\jenkins\workspace\CodeScanSnyk
The recommended git tool is: NONE
No credentials specified
> git.exe rev-parse --resolve-git-dir C:\ProgramData\jenkins\jenkins\workspace\CodeScanSnyk\.git # timeout=10
Fetching changes from the remote Git repository
> git.exe config remote.origin.url https://github.com/hkshitesh/Secure-Coding.git # timeout=10
Fetching upstream changes from https://github.com/hkshitesh/Secure-Coding.git
> git.exe --version # timeout=10
> git --version # 'git version 2.47.1.windows.1'
> git.exe fetch --tags --force --progress -- https://github.com/hkshitesh/Secure-Coding.git +refs/heads/*:refs/remotes/origin/* # timeout=10
> git.exe rev-parse "refs/remotes/origin/main^{commit}" # timeout=10
Checking out Revision 5e3aaedae26e41b315263bf3151216fd7eb416b1 (refs/remotes/origin/main)
> git.exe config core.sparsecheckout # timeout=10
> git.exe checkout -f 5e3aaedae26e41b315263bf3151216fd7eb416b1 # timeout=10
Commit message: "Add files via upload"
> git.exe rev-list --no-walk 5e3aaedae26e41b315263bf3151216fd7eb416b1 # timeout=10
Testing project...
> C:\ProgramData\jenkins\jenkins\tools\io.snyk.jenkins.tools.SnykInstallation\Snyk\snyk-win.exe test --json --severity-threshold=low
Vulnerabilities found!
Result: 1 known vulnerabilities | 6 dependencies
Generating report...
```



The screenshot shows the Jenkins console output for a job named 'CodeScanSnyk'. The output displays the execution of the Snyk CLI commands to test and monitor a project. It reports 1 known vulnerability and 6 dependencies. A report is generated and archived. The console also shows a link to explore the snapshot on the Snyk.io platform and a tip about detecting multiple supported manifests. The job ends with an error message: 'ERROR: Snyk has detected security vulnerabilities in your project. Finished: FAILURE'.

```
jenkins project...
> C:\ProgramData\jenkins\tools\io.snyk.jenkins.tools.SnykInstallation\Snyk\snyk-win.exe test --json --severity-threshold=low
Vulnerabilities found!
Result: 1 known vulnerabilities | 6 dependencies
Generating report...
> C:\ProgramData\jenkins\tools\io.snyk.jenkins.tools.SnykInstallation\Snyk\snyk-to-html-win.exe -i
C:\ProgramData\jenkins\workspace\CodeScanSnyk\2025-09-25T12-13-27-157759100Z_snyk_report.json
Archiving artifacts
Monitoring project...
> C:\ProgramData\jenkins\tools\io.snyk.jenkins.tools.SnykInstallation\Snyk\snyk-win.exe monitor --severity-threshold=low

Monitoring C:\ProgramData\jenkins\workspace\CodeScanSnyk (demo.secure.code.db:demo.secure.code.db)...

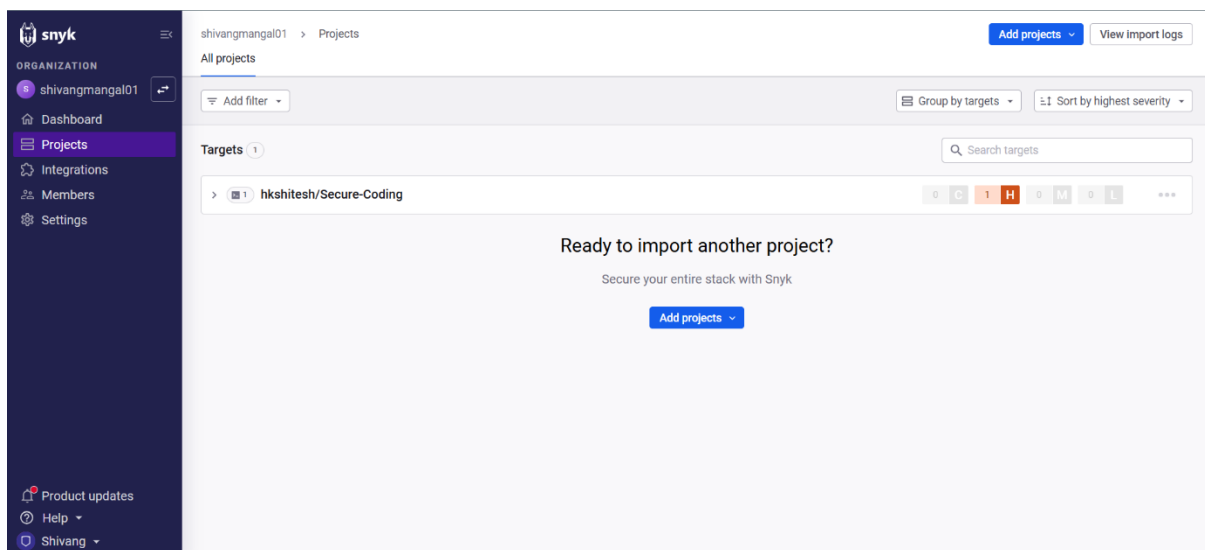
Explore this snapshot at https://app.snyk.io/org/shivangmangal01/project/9eb55728-8eb8-408f-b581-c1a31d37f44e/history/7f0ec18b-1333-47be-866b-a66de32d6d6f

Tip: Detected multiple supported manifests (1), use --all-projects to scan all of them at once.

Notifications about newly disclosed issues related to these dependencies will be emailed to you.

ERROR: Snyk has detected security vulnerabilities in your project.
Finished: FAILURE
```

4.4 To navigate to the Snyk tool to review code, scan reports under the Projects section



By following the above steps, you have successfully demonstrated the setup of the Snyk plugin in Jenkins for static application security testing (SAST), to automatically detect vulnerabilities in their codebase during development, thereby enhancing application security before deployment.