

## Lab Exercise 19

### Setting up Snky for SAST in Jenkins

**Objective:** To demonstrate the setup of the Snky 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:** Snky

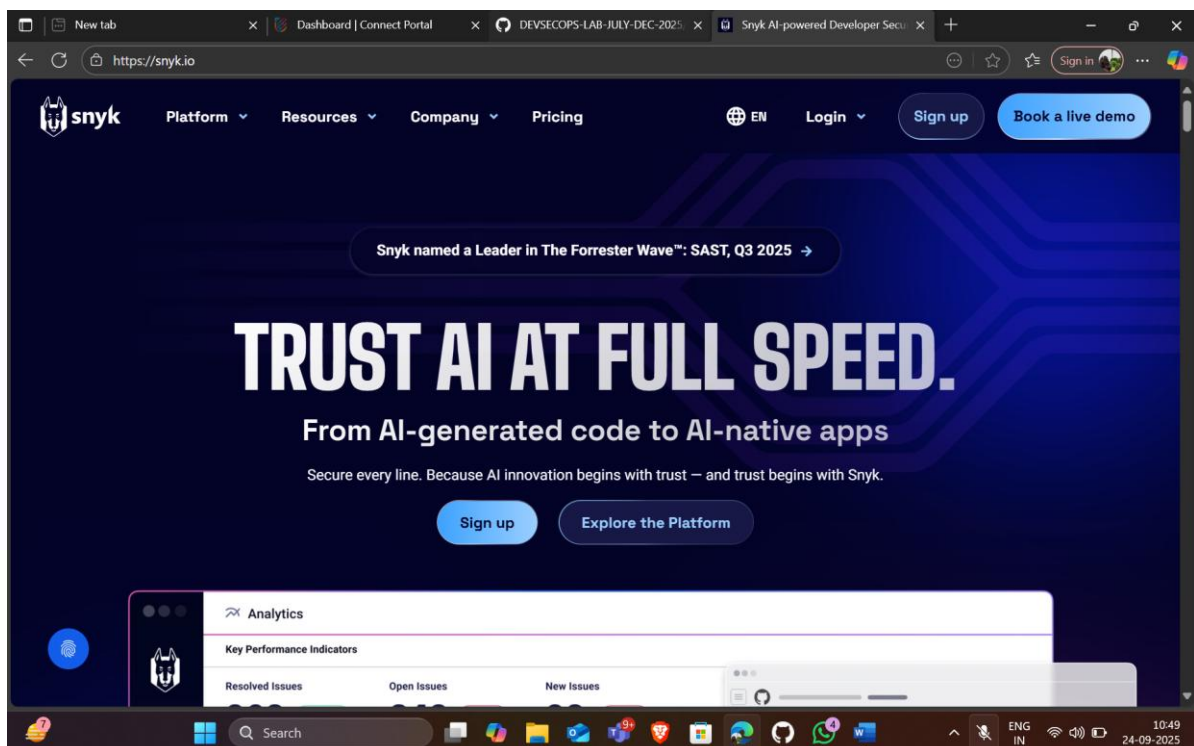
**Prerequisites:** None

Steps to be followed:

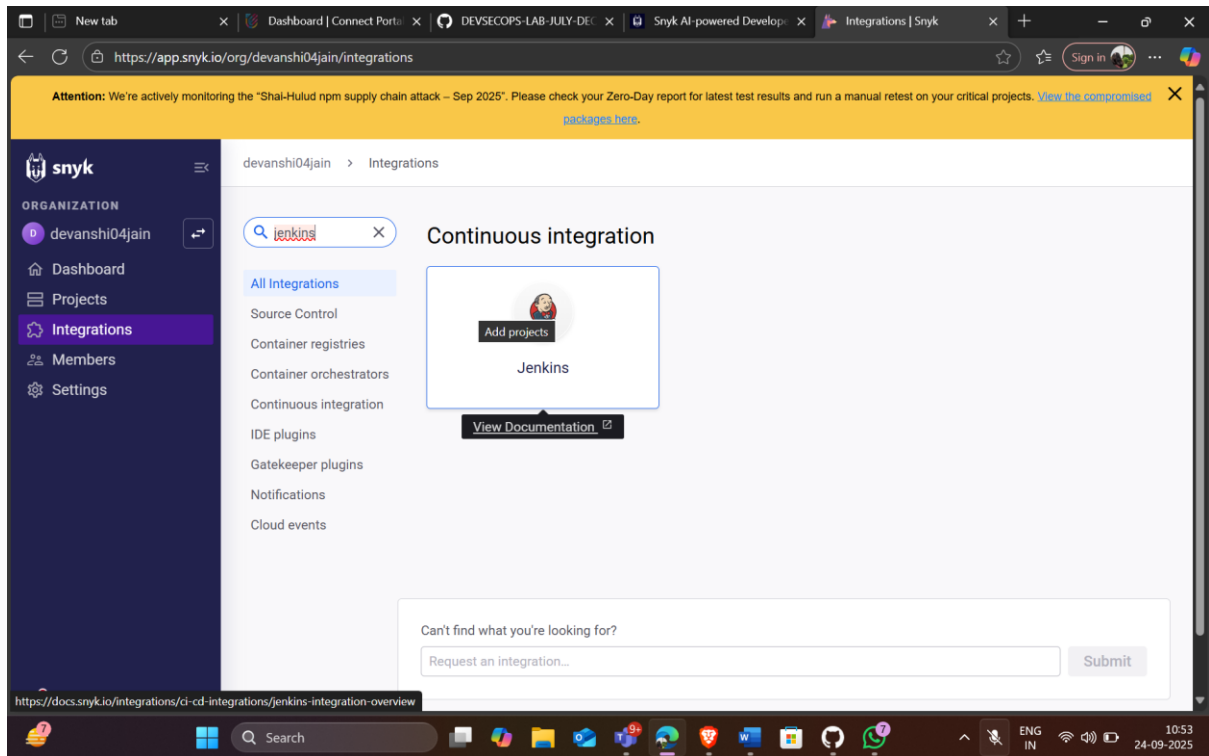
1. Configure Snky as a SAST scan tool
2. Create and configure a Jenkins job for Snky integration
3. Manage Snky API and Jenkins credentials
4. Configure the Jenkins job for scanning

#### Step 1: Configure Snky as a SAST scan tool

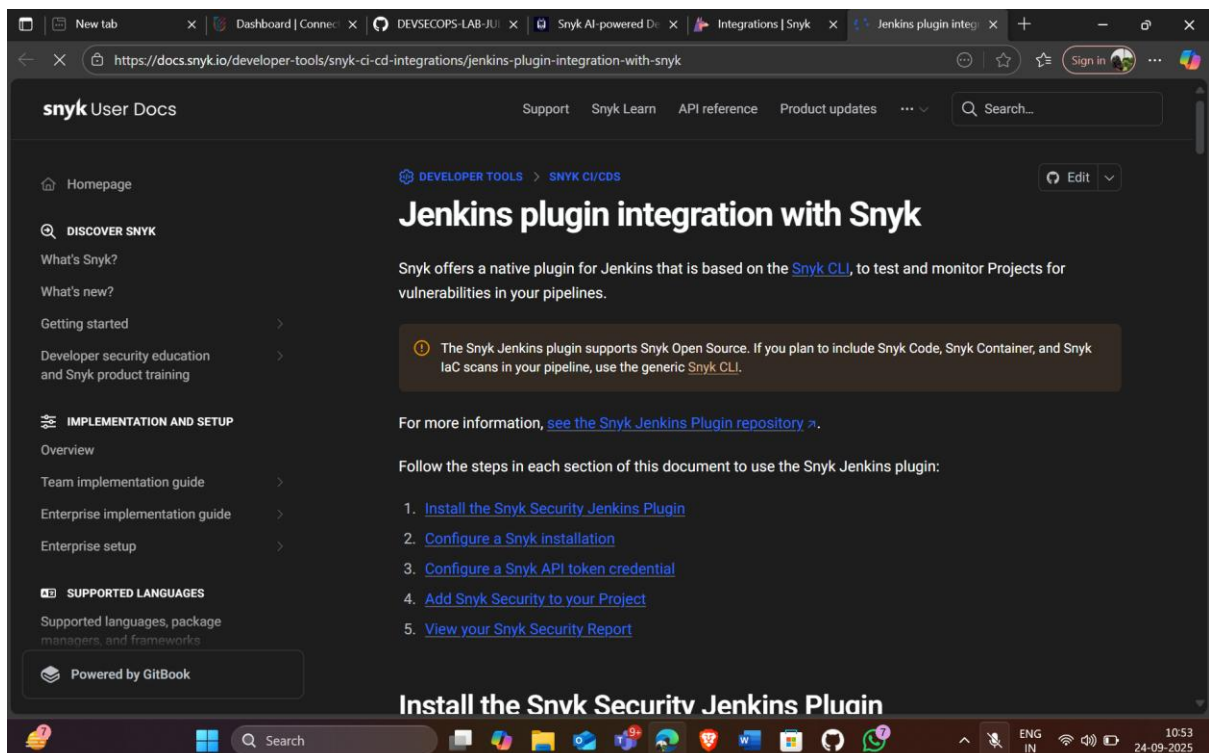
1.1 Visit <https://snky.io/>, sign up for a new Snky account, and log in



## 1.2 Navigate to **Integrations** and select **Jenkins**

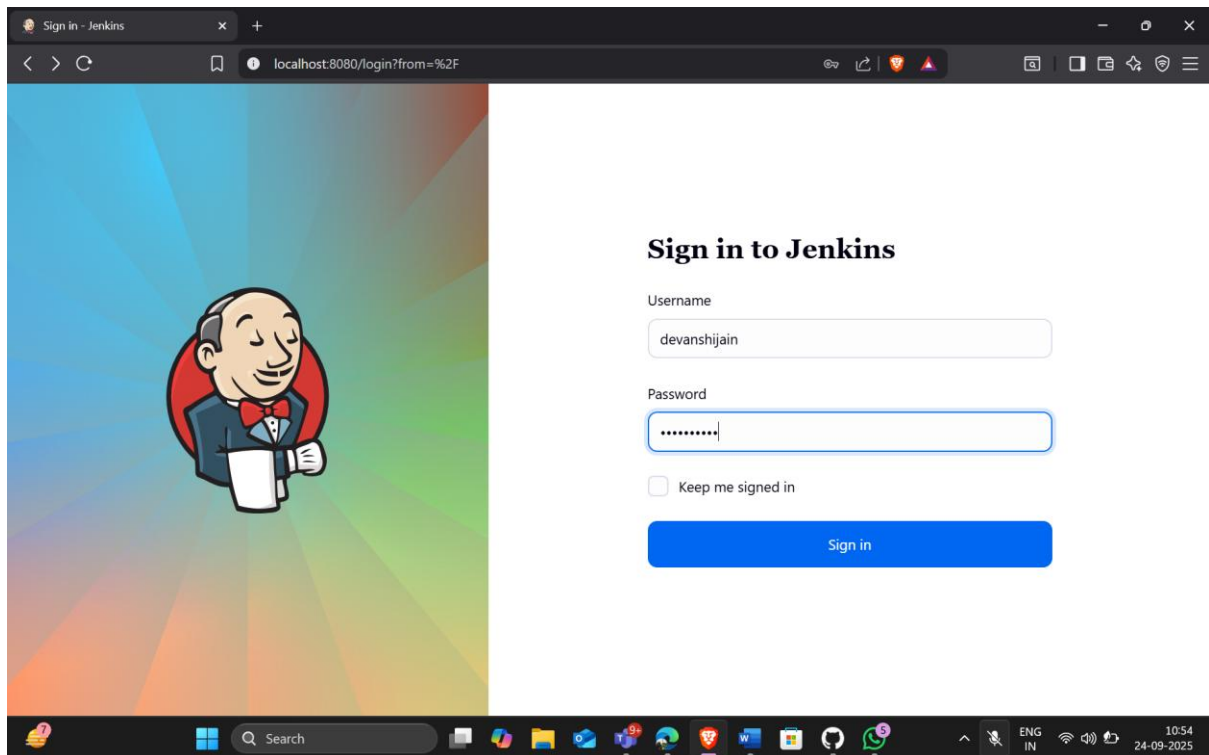


This will direct you to the documentation for integrating Snyk with Jenkins.



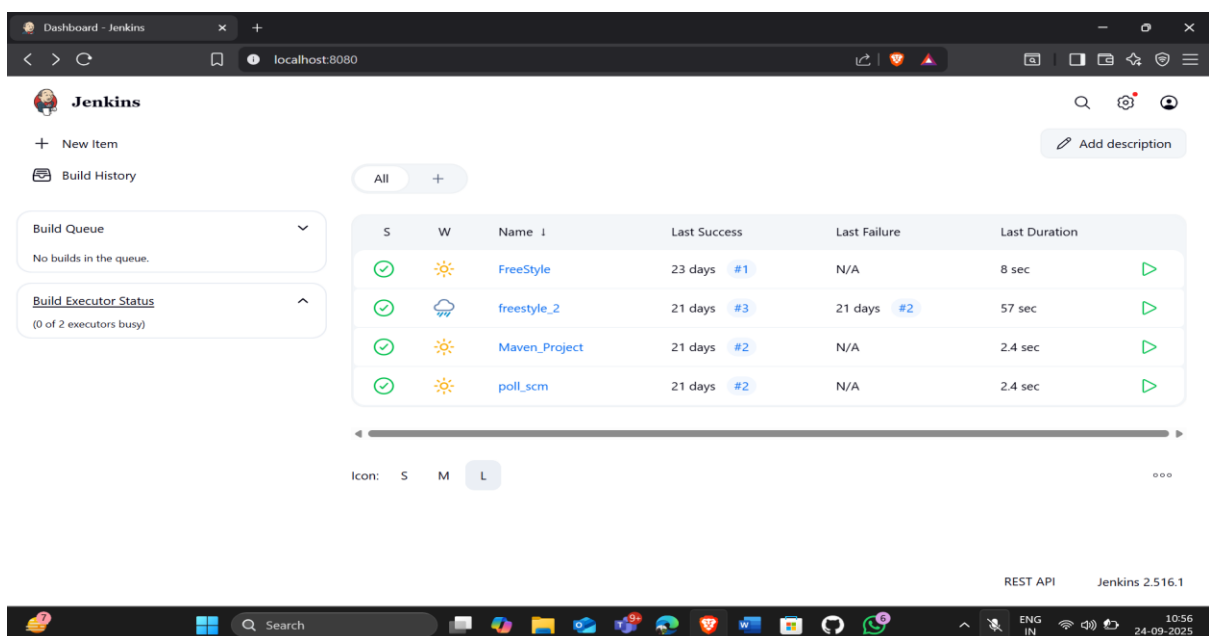
## Step 2: Create and configure a Jenkins job for Snky integration

### 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 screenshot shows the Jenkins 'Available plugins' page. The browser address bar is 'localhost:8080/manage/pluginManager/available'. The Jenkins header shows 'Manage Jenkins' and 'Plugins'. On the left, the 'Available plugins' tab is selected. A search bar contains 'snyk'. A table lists available plugins:

Install	Name ↓	Released	Health
<input checked="" type="checkbox"/>	Snyk Security 5.0.1 <a href="#">DevSecOps</a> Add the ability to test your code dependencies for vulnerabilities against Snyk database	3 mo 13 days ago	93

At the bottom right, it says 'REST API' and 'Jenkins 2.516.1'. The Windows taskbar at the bottom shows the time as 10:56 on 24-09-2025.

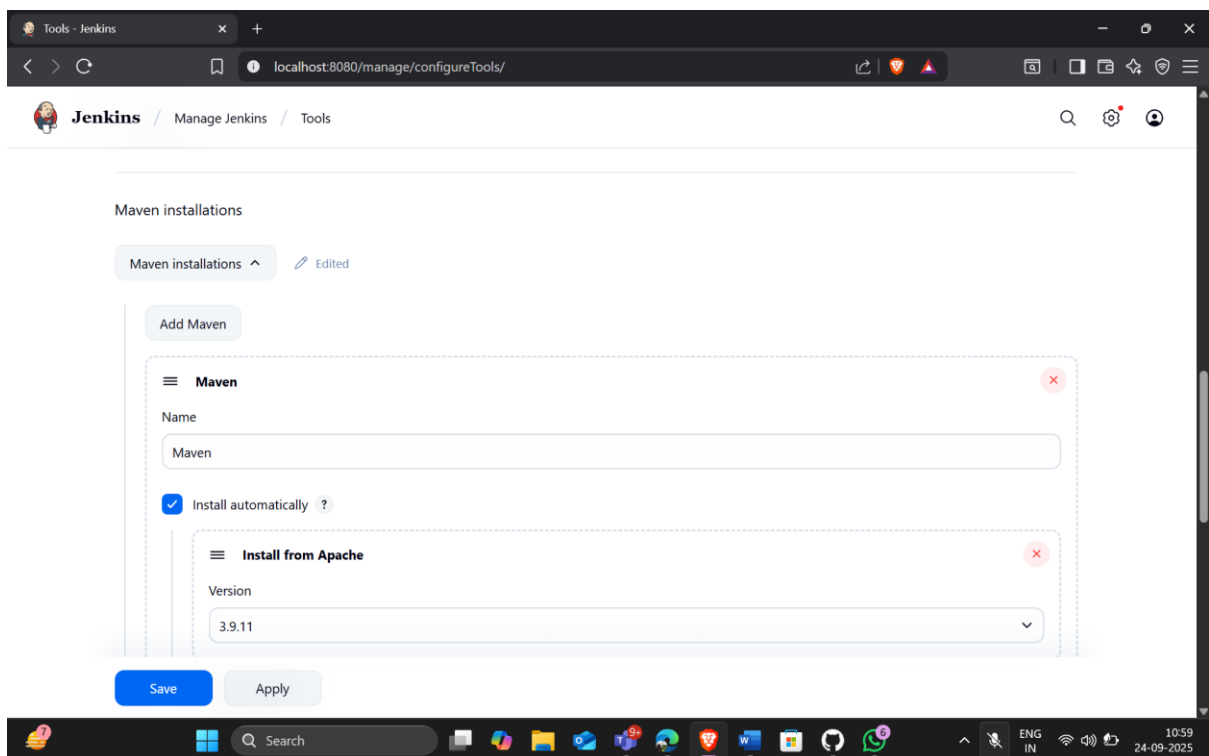
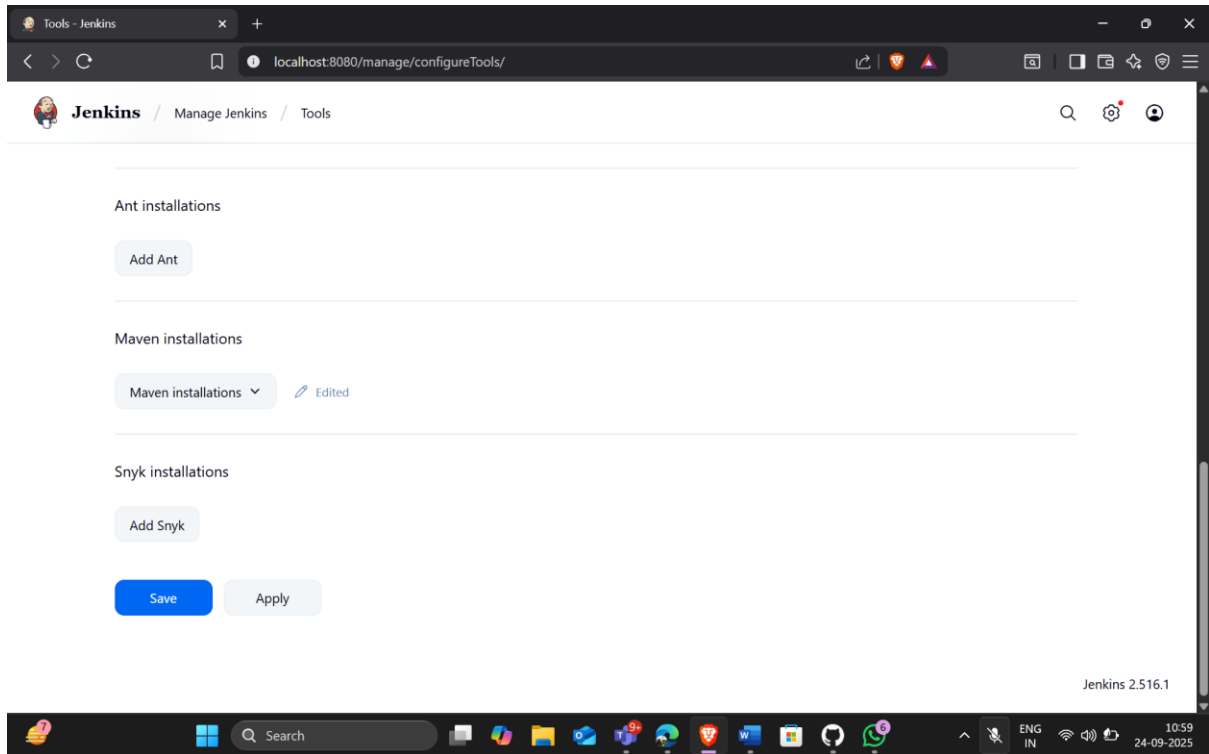
## 2.3 To configure Maven and Snyk in the **Global Tool Configuration**, click on **Tools** inside **Manage Jenkins**

The screenshot shows the Jenkins 'Manage Jenkins' page. The browser address bar is 'localhost:8080/manage/'. The Jenkins header shows 'Manage Jenkins'. On the left, the 'Tools' option is highlighted under 'System Configuration'. The main content area shows various configuration options:

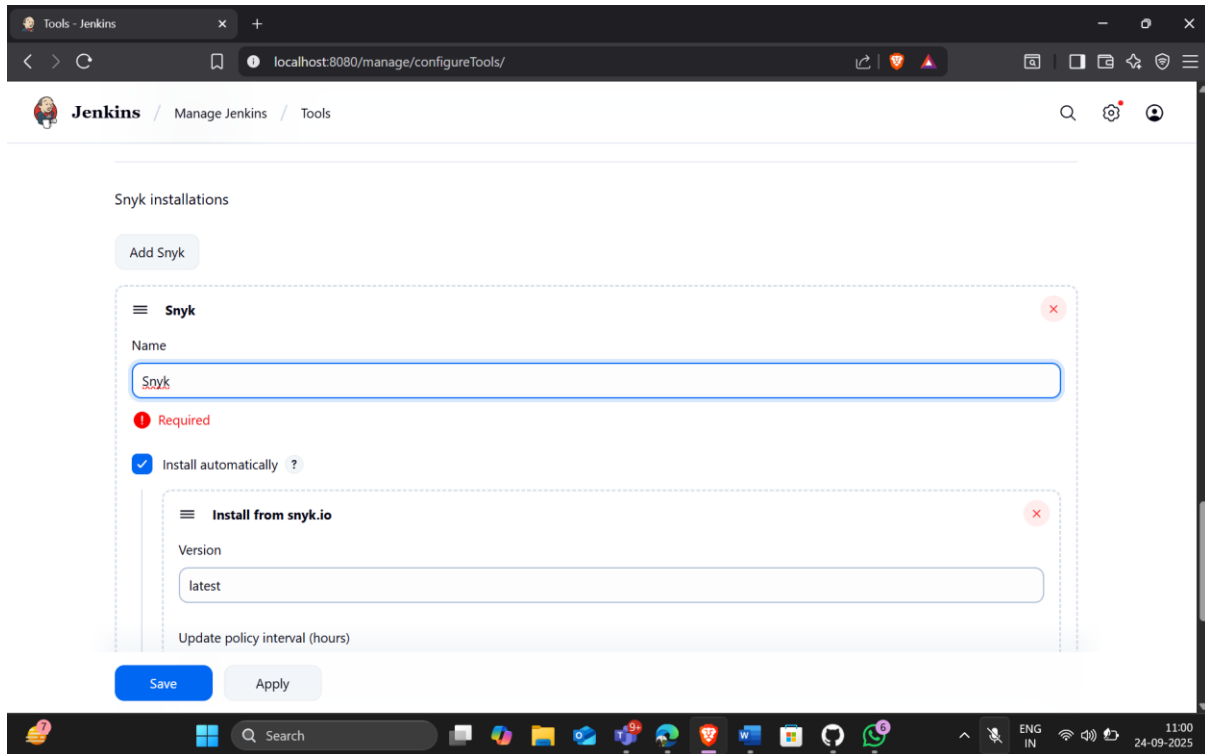
- System Configuration**
  - System: Configure global settings and paths.
  - Plugins: Add, remove, disable or enable plugins that can extend the functionality of Jenkins.
  - Clouds: Add, remove, and configure cloud instances to provision agents on-demand.
  - Tools: Configure tools, their locations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

The Windows taskbar at the bottom shows the time as 10:58 on 24-09-2025.

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

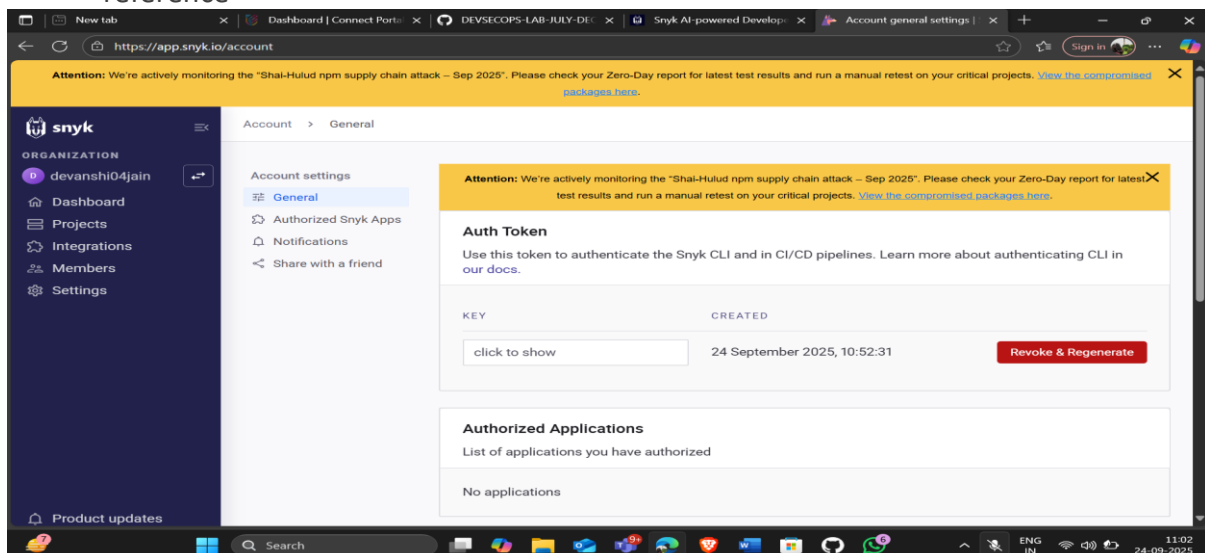


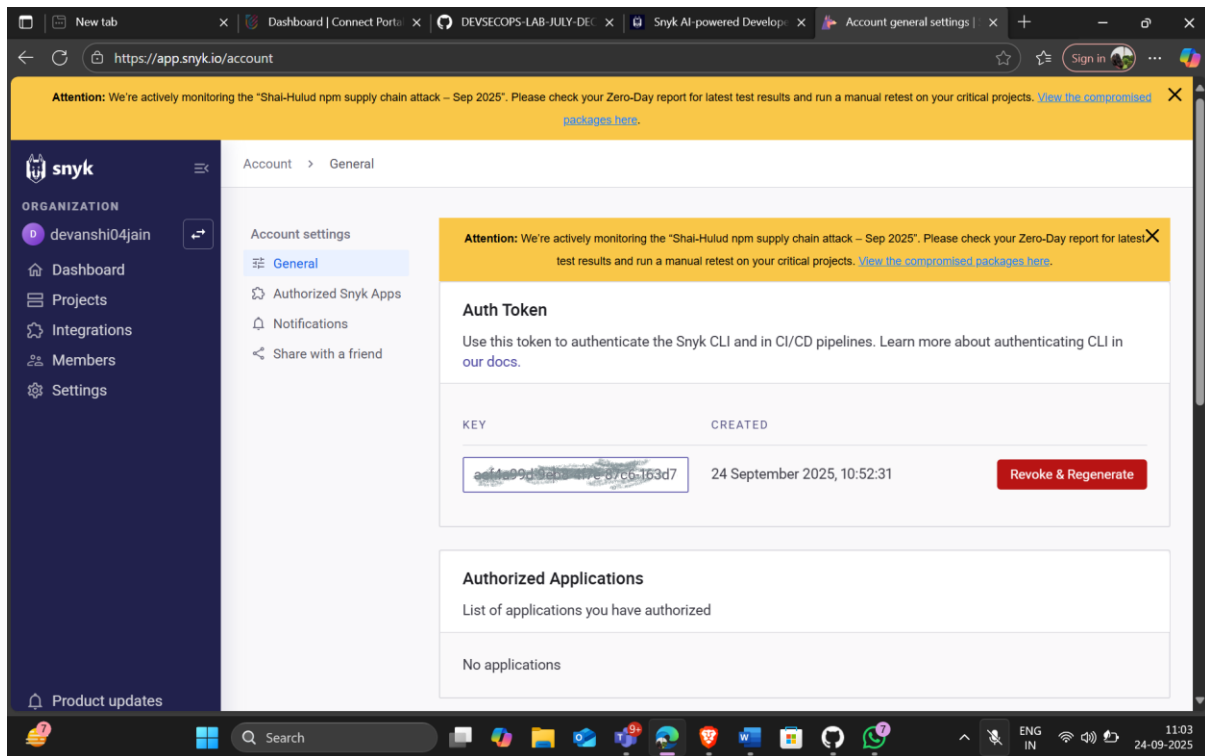
2.5 To add Snky, click on **Add Snky** under **Snyk Installations**, add **Name** as **Snyk**, and click on the **Save** button



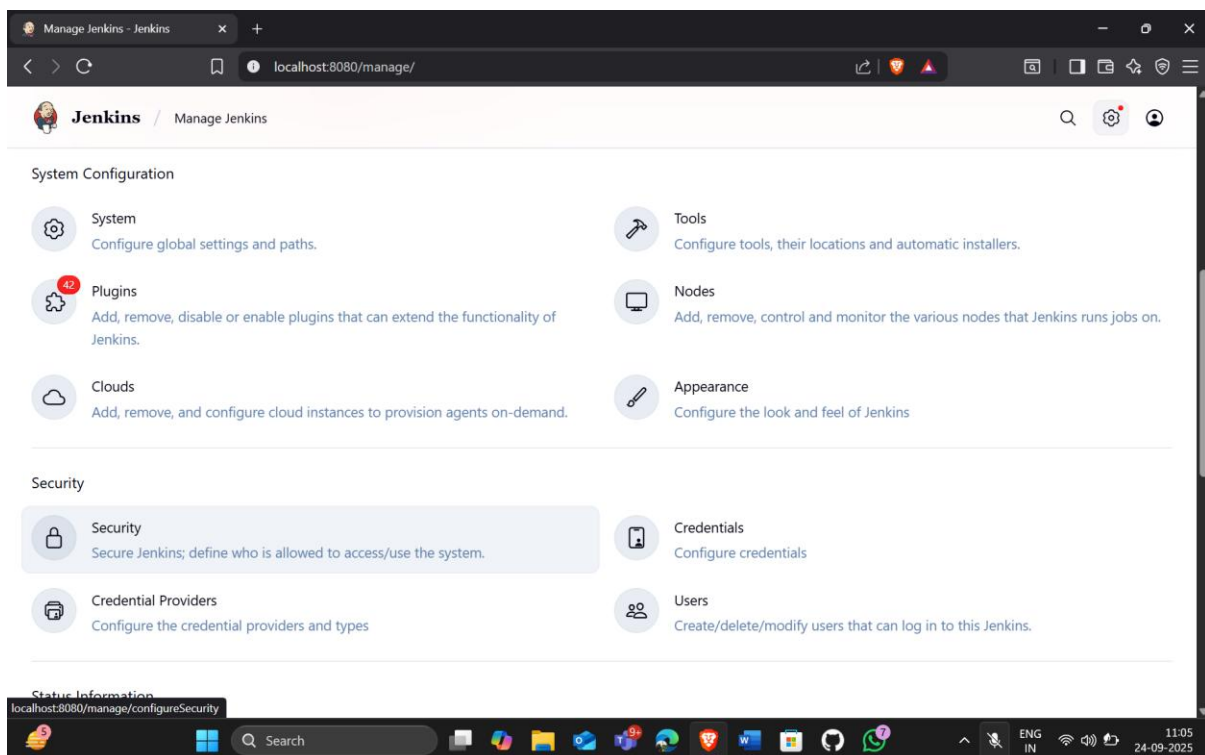
### Step 3: Manage Snyk API and Jenkins credentials

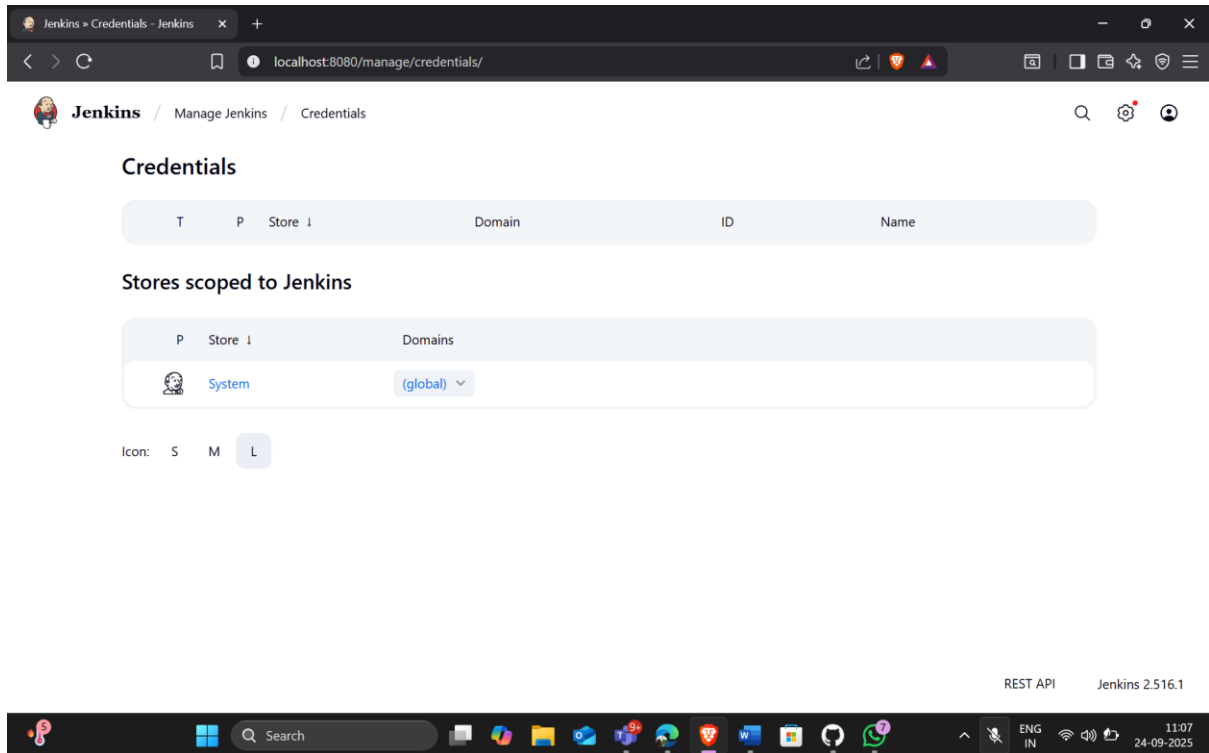
3.1 To retrieve your Snyk API token, go to **Account Settings** in your Snyk account, click on **Click to show** under the Auth Token key field, and copy the token for further reference





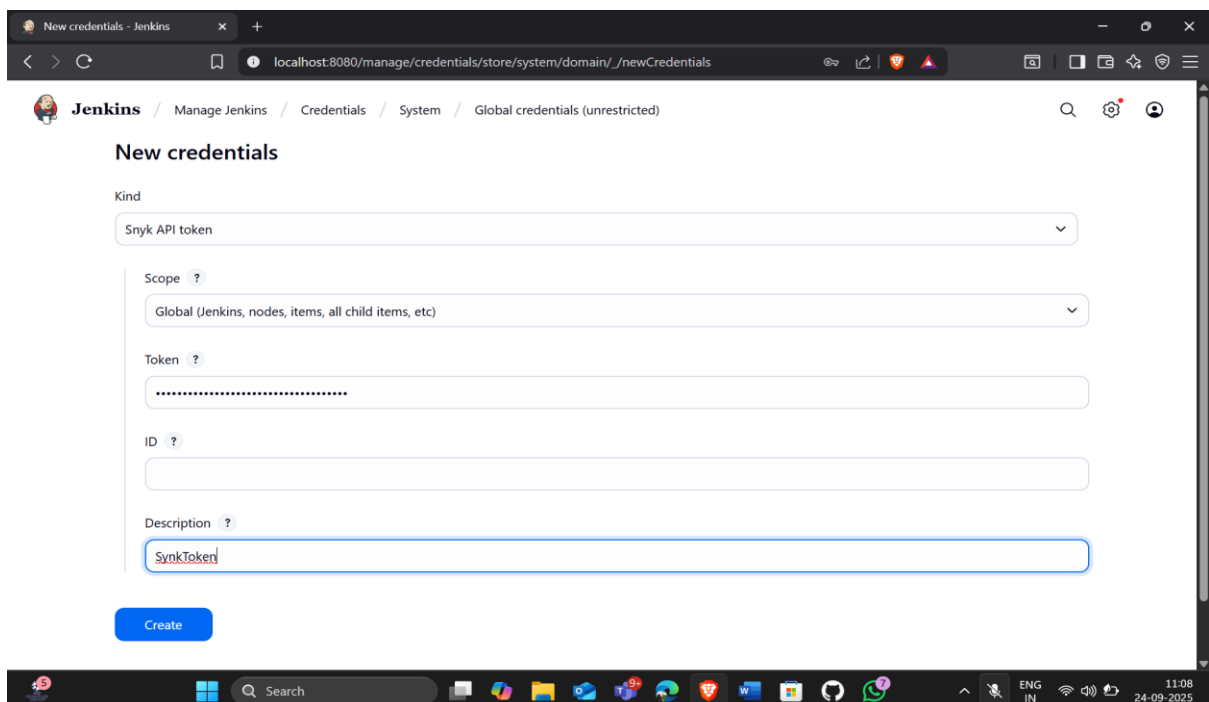
3.2 In the Jenkins interface, go to **Manage Jenkins**, select **Security**, then choose **Credentials** and select **global** to add global credentials





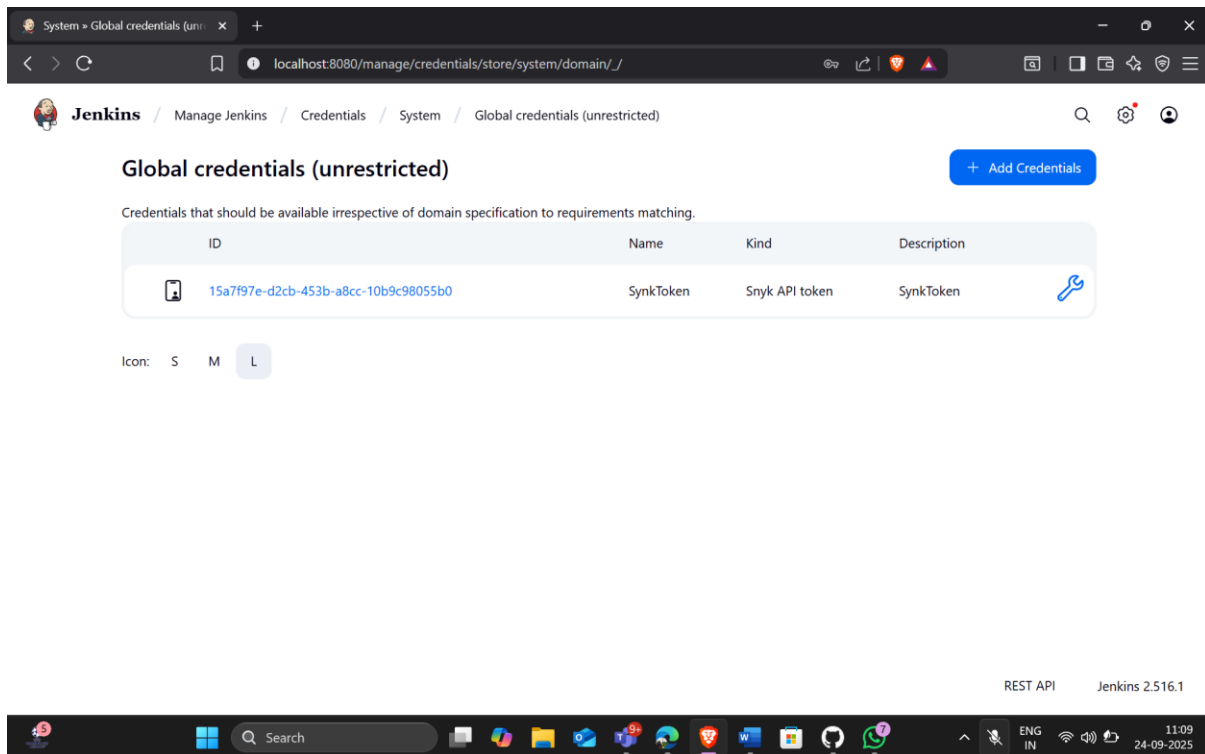
The screenshot shows the Jenkins web interface in a browser. The address bar indicates the URL is `localhost:8080/manage/credentials/`. The page title is "Jenkins" and the breadcrumb navigation shows "Manage Jenkins" > "Credentials". The main heading is "Credentials". Below it, there is a table with columns: "T", "P", "Store", "Domain", "ID", and "Name". Under the heading "Stores scoped to Jenkins", there is a table with columns: "P", "Store", and "Domains". The "Store" column shows "System" and the "Domains" column shows "(global)". At the bottom right, it says "REST API" and "Jenkins 2.516.1". The Windows taskbar is visible at the bottom with the date "24-09-2025" and time "11:07".

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 "New credentials" form in the Jenkins web interface. The breadcrumb navigation is "Manage Jenkins" > "Credentials" > "System" > "Global credentials (unrestricted)". The form has the following fields: "Kind" (a dropdown menu with "Snyk API token" selected), "Scope" (a dropdown menu with "Global (Jenkins, nodes, items, all child items, etc)" selected), "Token" (a text input field with masked characters), "ID" (a text input field), and "Description" (a text input field with "SnykToken" entered). A blue "Create" button is at the bottom left. The Windows taskbar is visible at the bottom with the date "24-09-2025" and time "11:08".



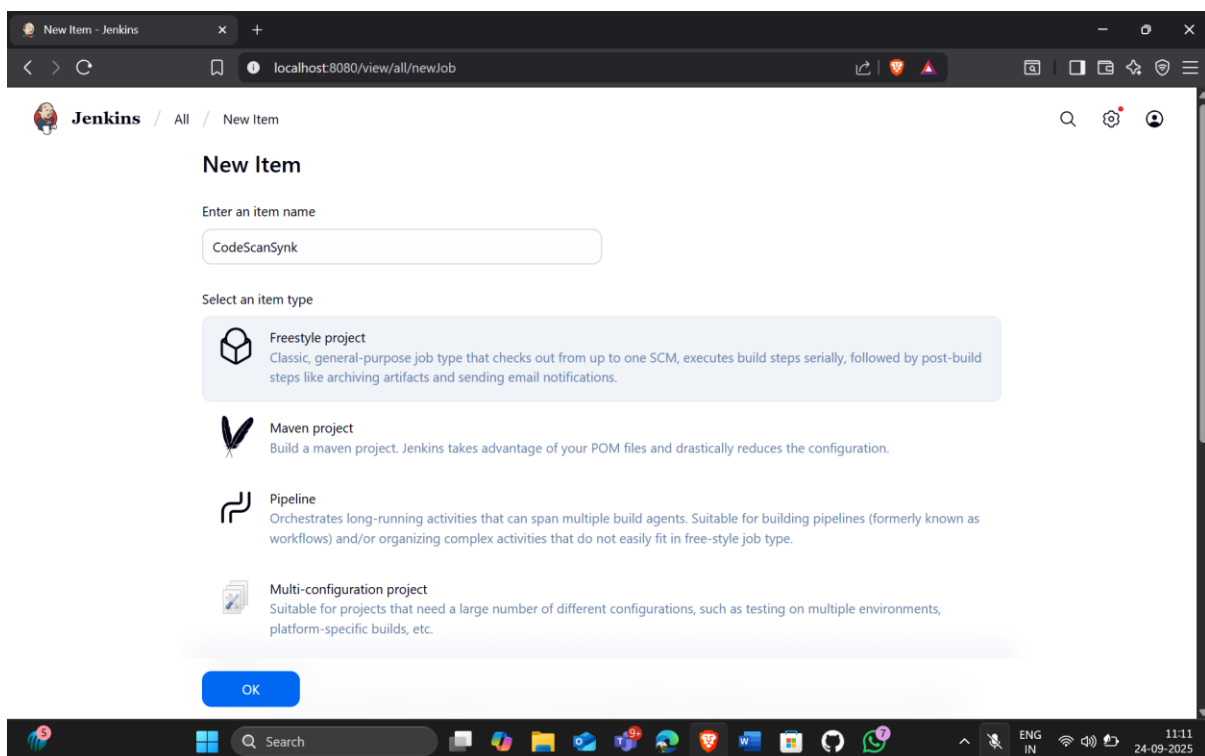


The screenshot shows the Jenkins web interface at the URL `localhost:8080/manage/credentials/store/system/domain/`. The page title is "Global credentials (unrestricted)". There is a blue button labeled "+ Add Credentials". Below this, a table lists credentials. The table has columns: ID, Name, Kind, and Description. One credential is listed with ID `15a7f97e-d2cb-453b-a8cc-10b9c98055b0`, Name `SynkToken`, Kind `Snyk API token`, and Description `SynkToken`. Below the table, there are tabs for "Icon: S M L". The bottom of the screenshot shows a Windows taskbar with various application icons and a system tray showing the time as 11:09 on 24-09-2025.

ID	Name	Kind	Description
15a7f97e-d2cb-453b-a8cc-10b9c98055b0	SynkToken	Snyk API token	SynkToken

## Step 4: Configure the Jenkins job for scanning

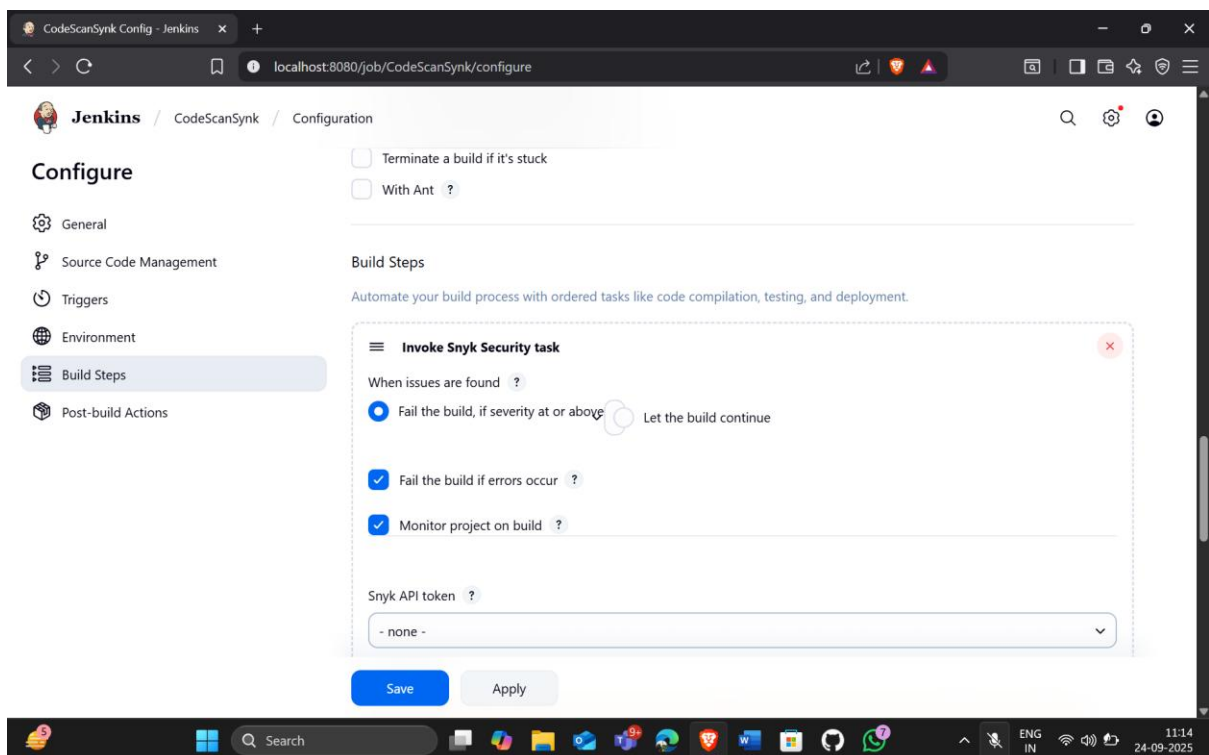
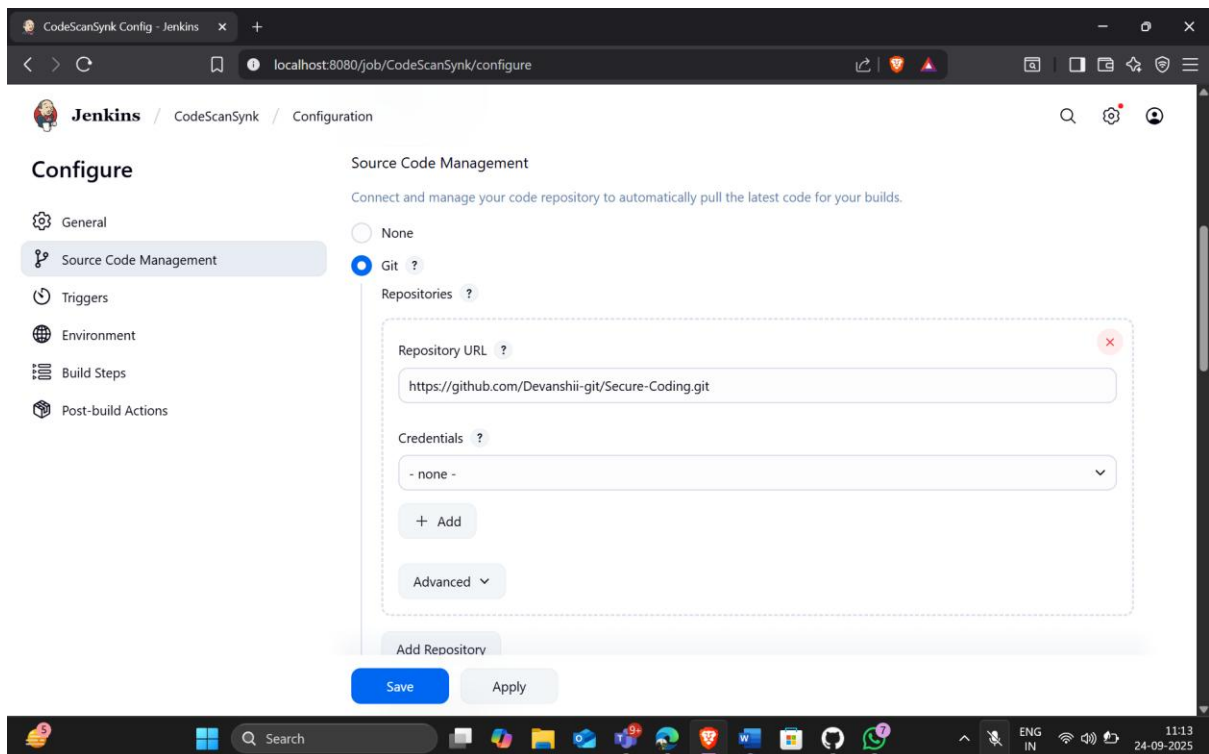
4.1 To create a new Jenkins job, click on **New Item**, enter the item name as **CodeScanSnyk**, select **Freestyle project**, and then click **OK**

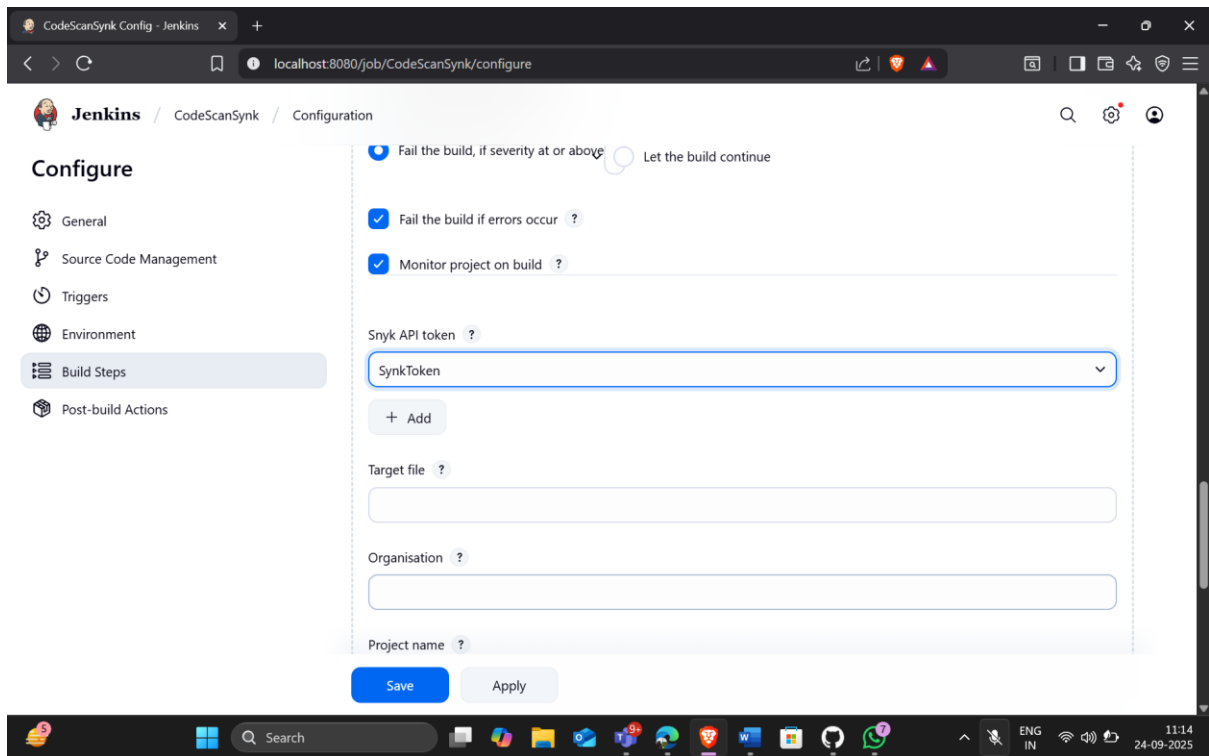


The screenshot shows the Jenkins "New Item" page at the URL `localhost:8080/view/all/newJob`. The page title is "New Item". There is a text input field for "Enter an item name" with the value `CodeScanSnyk`. Below this, there is a section "Select an item type" with four options: "Freestyle project", "Maven project", "Pipeline", and "Multi-configuration project". The "Freestyle project" option is selected and highlighted. At the bottom, there is a blue button labeled "OK". The bottom of the screenshot shows a Windows taskbar with various application icons and a system tray showing the time as 11:11 on 24-09-2025.

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 Snky Security task** with the name **SnykToken**. Finally, click the **Save** button to create the build.

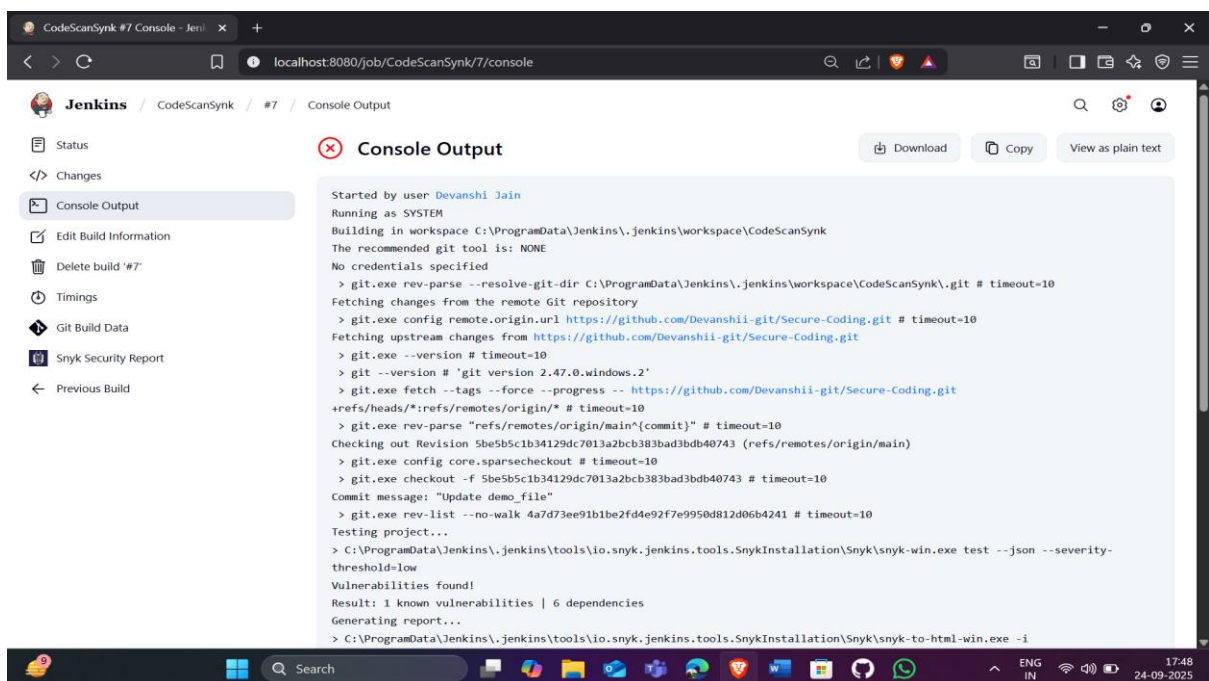
Use GitHub Repo: <https://github.com/hkshitesh/Secure-Coding.git>

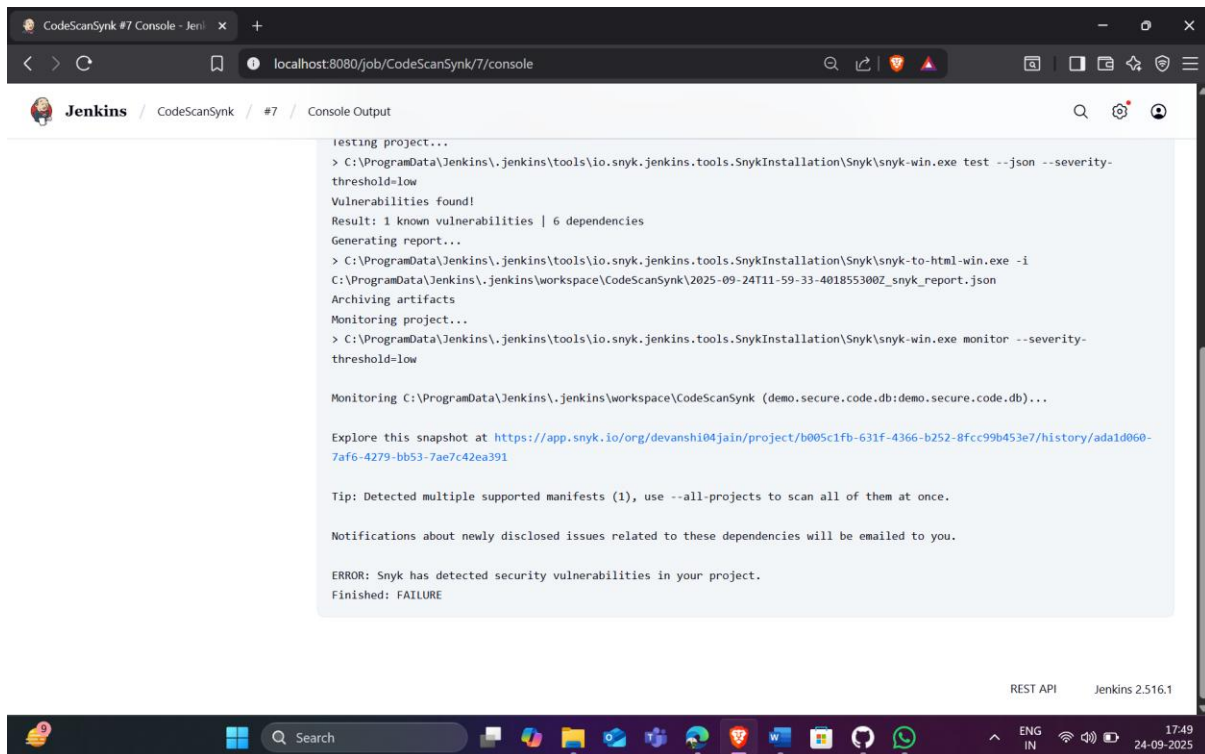




**Note:** For GitHub repository URL, use <https://github.com/hkshitesh/Secure-Coding.git>

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





The screenshot shows a Jenkins console output window for a job named 'CodeScanSynk'. The output displays the execution of Snyk 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 tip about using '--all-projects' and a notification about newly disclosed issues. The scan finished with a FAILURE status.

```
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...
> C:\ProgramData\Jenkins\jenkins\tools\io.snyk.jenkins.tools.SnykInstallation\Snyk\snyk-to-html-win.exe -i
C:\ProgramData\Jenkins\jenkins\workspace\CodeScanSynk\2025-09-24T11-59-33-401855300Z_snyk_report.json
Archiving artifacts
Monitoring project...
> C:\ProgramData\Jenkins\jenkins\tools\io.snyk.jenkins.tools.SnykInstallation\Snyk\snyk-win.exe monitor --severity-
threshold=low

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

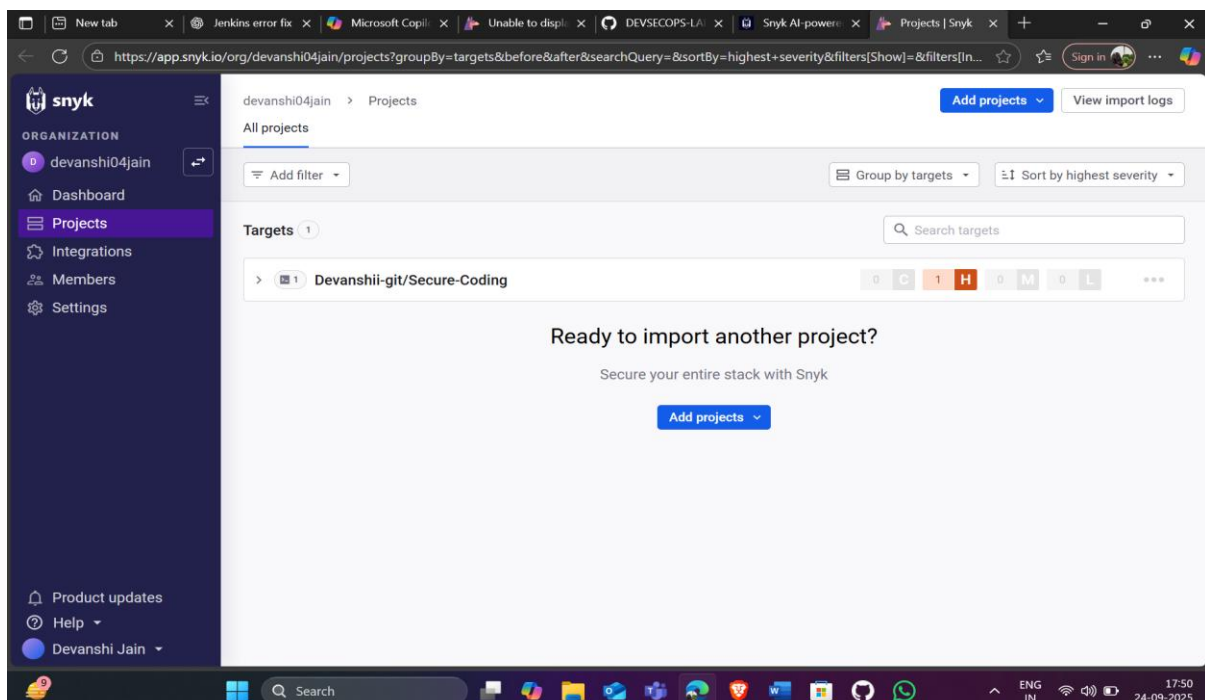
Explore this snapshot at https://app.snyk.io/org/devanshi04jain/project/b005c1fb-631f-4366-b252-8fcc99b453e7/history/ada1d060-7af6-4279-bb53-7ae7c42ea391

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.