

# Lab Exercise 19

## Setting up Snyk for SAST in Jenkins

Name- Gourav das

SAP ID- 500122586

Batch- 2

**Objective:** To demonstrate 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

**Tools required:** Snyk

Steps to be followed:

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

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

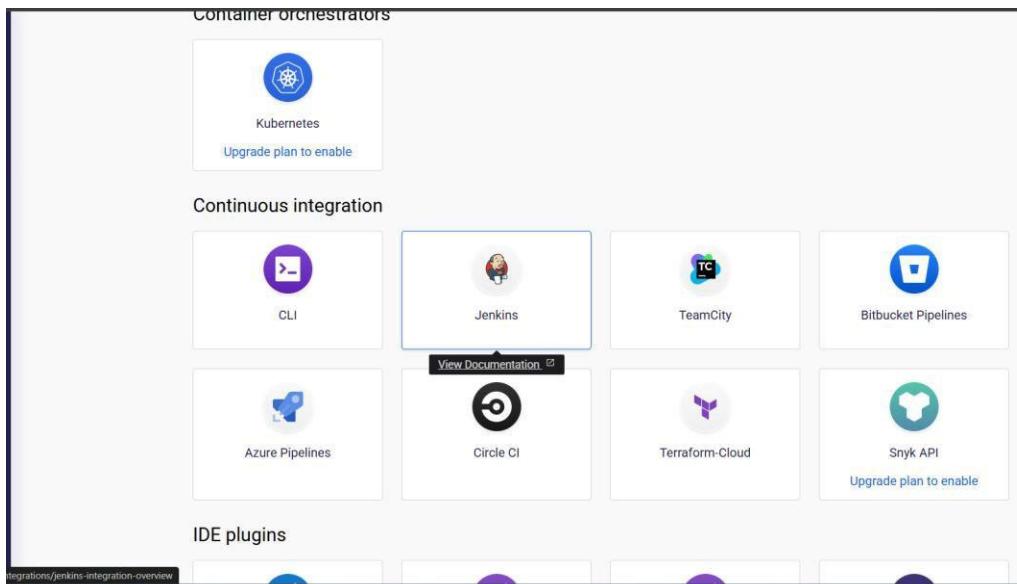
1. Visit <https://snyk.io/>, sign up for a new Snyk account, and log in

The screenshot shows the Snyk.io homepage with the following sections:

- All projects** (highlighted)
- Secure your dependencies with Snyk**: Scan your projects to get started.
- Monitor deployed apps**:
  - Test apps for vulnerable dependencies
  - Get notifications about new vulnerabilities[Browse integrations](#)
- Protect your source code**:
  - Test repos for vulnerable dependencies
  - Create pull requests with fixes and patches
  - Flag fix pull requests that add new vulnerabilities
  - Get notifications for new vulnerabilities[Add projects](#)
- Monitor local projects**:
  - Install our CLI tool to monitor local projects for known vulnerabilities:

```
npm install -g snyk
cd ~/projects/my-project/
snyk monitor
```[Full documentation for Snyk CLI](#)

## 2. Navigate to Integrations and select Jenkins



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

A screenshot of the Snyk Jenkins plugin integration documentation page. The title is 'Jenkins plugin integration with Snyk'. The page contains instructions for installing and configuring the Jenkins plugin. On the right side, there is a sidebar with links for 'Install the Snyk Security Jenkins Plugin', 'Configure a Snyk installation', 'Configure a Snyk API token credential', 'Add Snyk Security to your Project', 'View your Snyk Security Report', and 'Increase logging'. At the bottom, there is a cookie consent banner with 'Accept' and 'Reject' buttons.

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

1. Open Jenkins and log in to the Jenkins account:

- To install the Snyk plugin, navigate to **Manage Jenkins** and click **Available Plugins**, search for **Snyk Security** plugin, and then click **Install**

The screenshot shows the Jenkins Manage Jenkins interface under the Plugins section. A search bar at the top contains the text "sny". Below it, a table lists the "Snyk Security Plugin 5.0.1" with the following details:

| Name                       | Health | Enabled                |
|----------------------------|--------|------------------------|
| Snyk Security Plugin 5.0.1 | 93     | Enabled (green switch) |

The left sidebar includes links for Updates, Available plugins (which is currently selected), Installed plugins, Advanced settings, and Download progress.

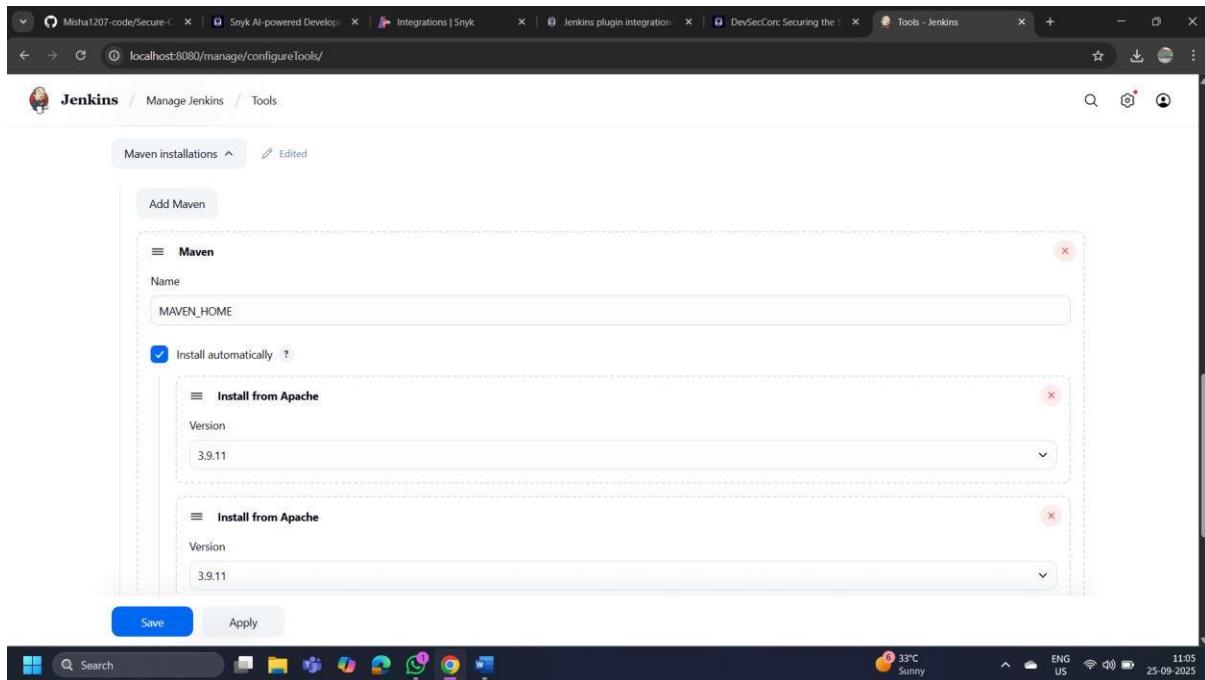
- To configure Maven and Snyk in the **Global Tool Configuration**, click on **Tools** inside **Manage Jenkins**

The screenshot shows the Jenkins Manage Jenkins interface under the Tools section. It displays various configuration options:

- System Configuration**: Includes links for System, Nodes, and Clouds.
- Security**: Includes links for Security, Credentials, and Credential Providers.
- Tools**: This section is expanded, showing the "Configure tools, their locations and automatic installers." link.
- Plugins**: Shows the "Add, remove, disable or enable plugins that can extend the functionality of Jenkins." link.
- Appearance**: Shows the "Configure the look and feel of Jenkins." link.

A red banner at the top indicates a security issue: "Jakarta Mail API 2.1.3-2: SMTP command injection vulnerability. A fix for this issue is available. Go to the [plugin manager](#) to update the plugin."

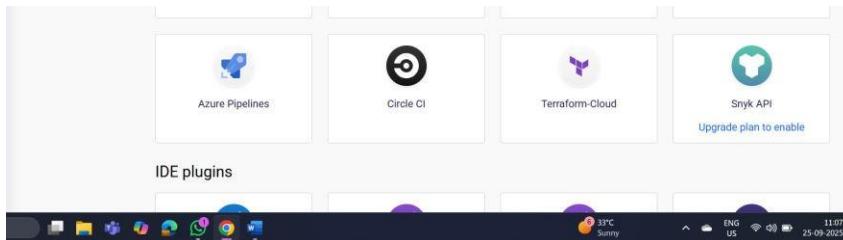
4. To add Maven, click on **Add Maven** under **Maven installations** and enter **Maven** as the **Name**



5. To add Snyk, click on **Add Snyk** under **Snyk Installations**, add Name as **Synk**, and click on the **Save** button

## Step 3: Manage Snyk API and Jenkins credentials

- 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



A screenshot of a web browser showing the "Account general settings" page. The URL is "https://app.snyk.io/account/general". On the left, a sidebar has "General" selected under "Account settings". The main content area is titled "Auth Token" and contains instructions: "Use this token to authenticate the Snyk CLI and in CI/CD pipelines. Learn more about authenticating CLI in our docs." It shows a "KEY" field containing "b1667d81-4426-475a-86bb-729570cead4b", a "CREATED" timestamp of "25 September 2025, 10:38:11", and a "Revoke &amp; Regenerate" button. Below this is a "Authorized Applications" section with the message "No applications". At the bottom is a "Preferred Organization" section with a dropdown menu showing "mishu5705".

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

A screenshot of a web browser showing the "Credentials" page in Jenkins. The URL is "http://localhost:8080/manage/credentials/". The page title is "Jenkins / Manage Jenkins / Credentials". It shows a table of credentials with one entry: "System" (Store: System, Domain: (global), ID: 46074b91-7108-4047-b9a7-7d4bb5a56cc6, Name: Misha1207-code/\*\*\*\*\*). Below the table is a section titled "Stores scoped to Jenkins" with a single entry: "System" (Store: System, Domain: (global)). At the bottom, there are buttons for "Icon: S", "M", and "L".



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

## Step 4: Configure the Jenkins job for scanning

1. To create a new Jenkins job, click on **New Item**, enter the item name as **CodeScanSnyk**, select **Freestyle project**, and then click **OK**
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
3. To check the build status, click on the build link under **Permalinks**. After that, click on **Console Output**

4. To navigate to the Snyk tool to review code, scan reports under the **Projects** section

The screenshot shows the Snyk interface for managing projects. At the top, there's a navigation bar with 'All projects' selected. Below it is a search bar with 'Add filter' and 'Targets' dropdown. The main area is titled 'Targets' and shows a single project entry: 'Misha1207-code/Secure-Coding'. This project has been 'Imported' a minute ago and 'Tested' a minute ago. It contains one critical issue (C) and no other severity levels (Info, Low, Medium). A 'Search targets' input field is also present. At the bottom, there's a message 'Ready to import another project?' and a 'Secure your entire stack with Snyk' button.

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.

5. To navigate to the Snyk tool to review code, scan reports under the **Projects** section

The screenshot shows the Snyk web interface. At the top, there's a navigation bar with 'All projects', 'Add filter', 'Group by targets', 'Sort by highest severity', and a search bar. Below this is a section titled 'Targets' with a count of 1. Under 'Targets', there's a list for 'Misha1207-code/Secure-Coding'. It shows a tree view with a single node 'demo.secure.code.db:demo.secure.code.db'. To the right of the tree, there are columns for 'Imported' (a minute ago), 'Tested' (a minute ago), and 'Issues' (with a dropdown arrow). The 'Issues' row shows counts for different severity levels: 0 Critical (C), 1 High (H), 0 Medium (M), 0 Low (L), and 4 Info (I). At the bottom of the page, there's a message 'Ready to import another project?' with a link 'Secure your entire stack with Snyk' and a blue 'Add projects' button.

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.

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

This screenshot is identical to the one above it, showing the Snyk interface with the same project details and message. It's a duplicate of the first screenshot.

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.