**Lab Exercise 19**

**Setting up Snyk for SAST in Jenkins**

**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

**Prerequisites:** None

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

**A screenshot of a website

Description automatically generated**

1. Navigate to **Integrations** and select **Jenkins**

A screenshot of a computer

Description automatically generated

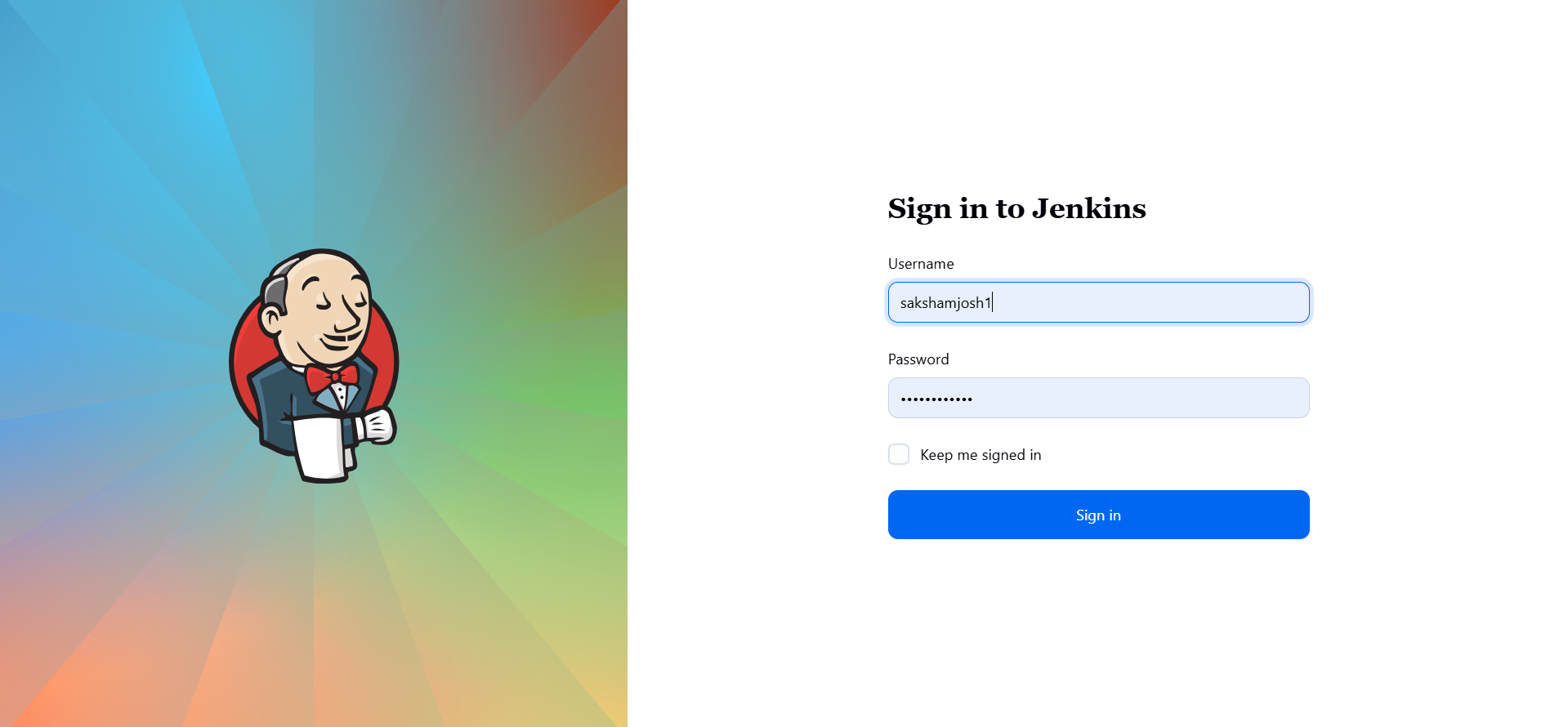
This will direct you to the documentation for integrating Snyk with Jenkins.

A screenshot of a web page

Description automatically generated

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

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



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

A screenshot of a computer

Description automatically generated

A screenshot of a computer

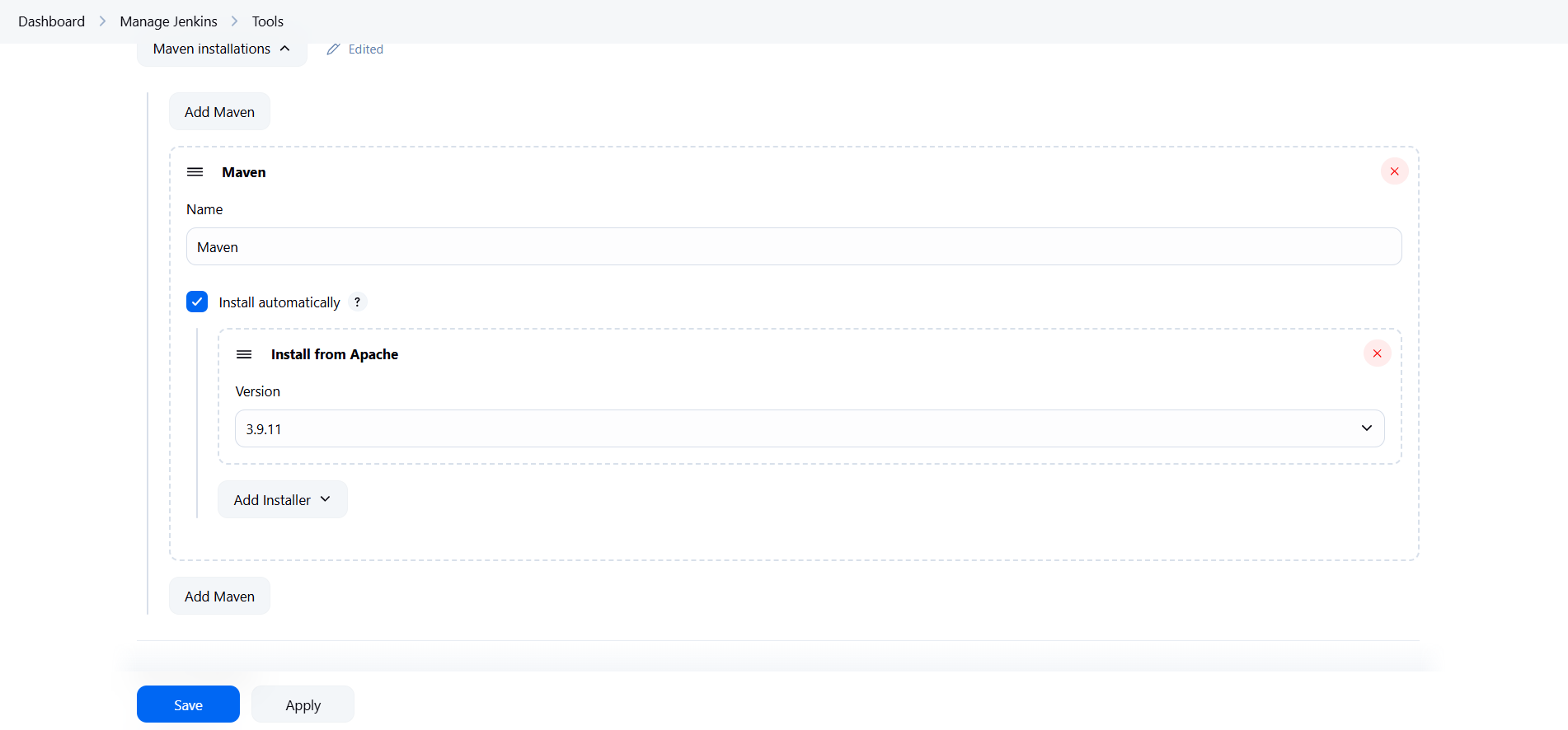
Description automatically generated

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

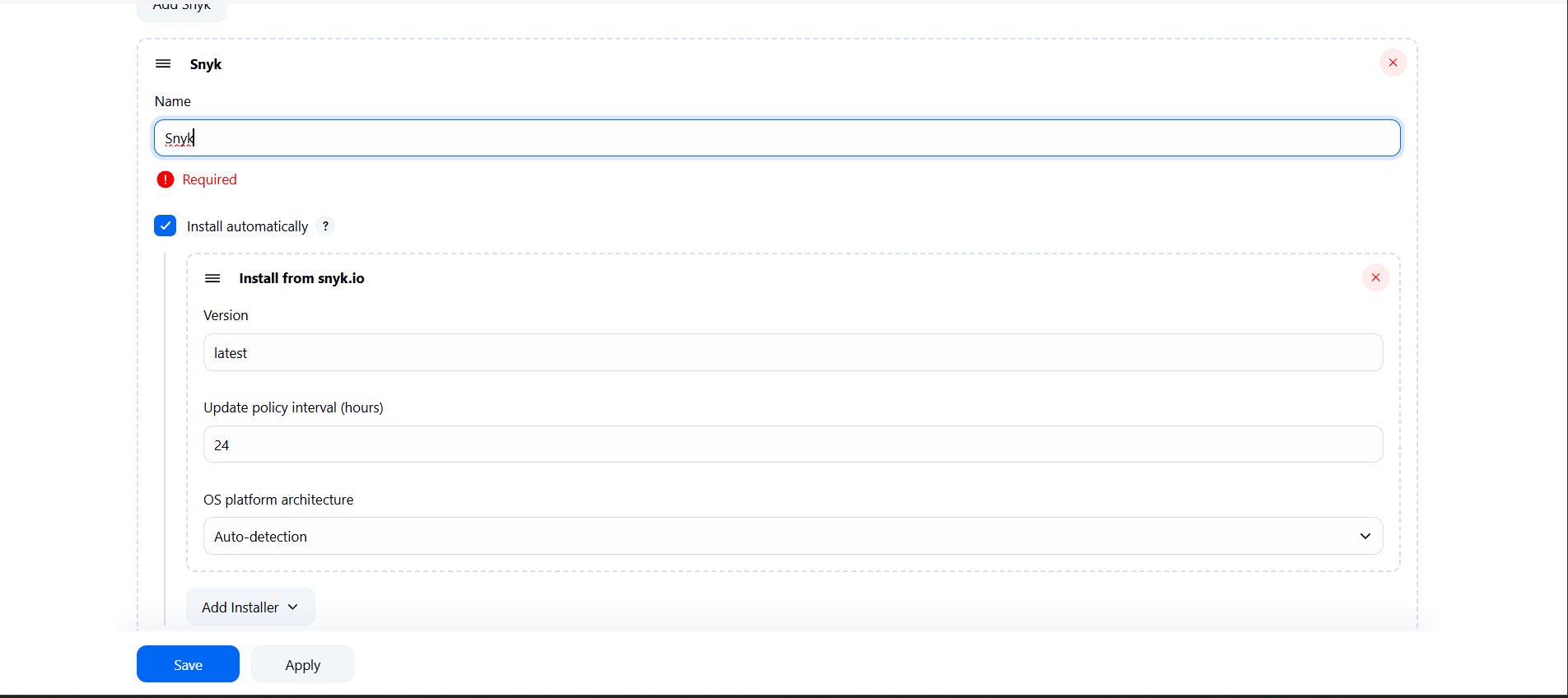
A screenshot of a computer

Description automatically generated

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

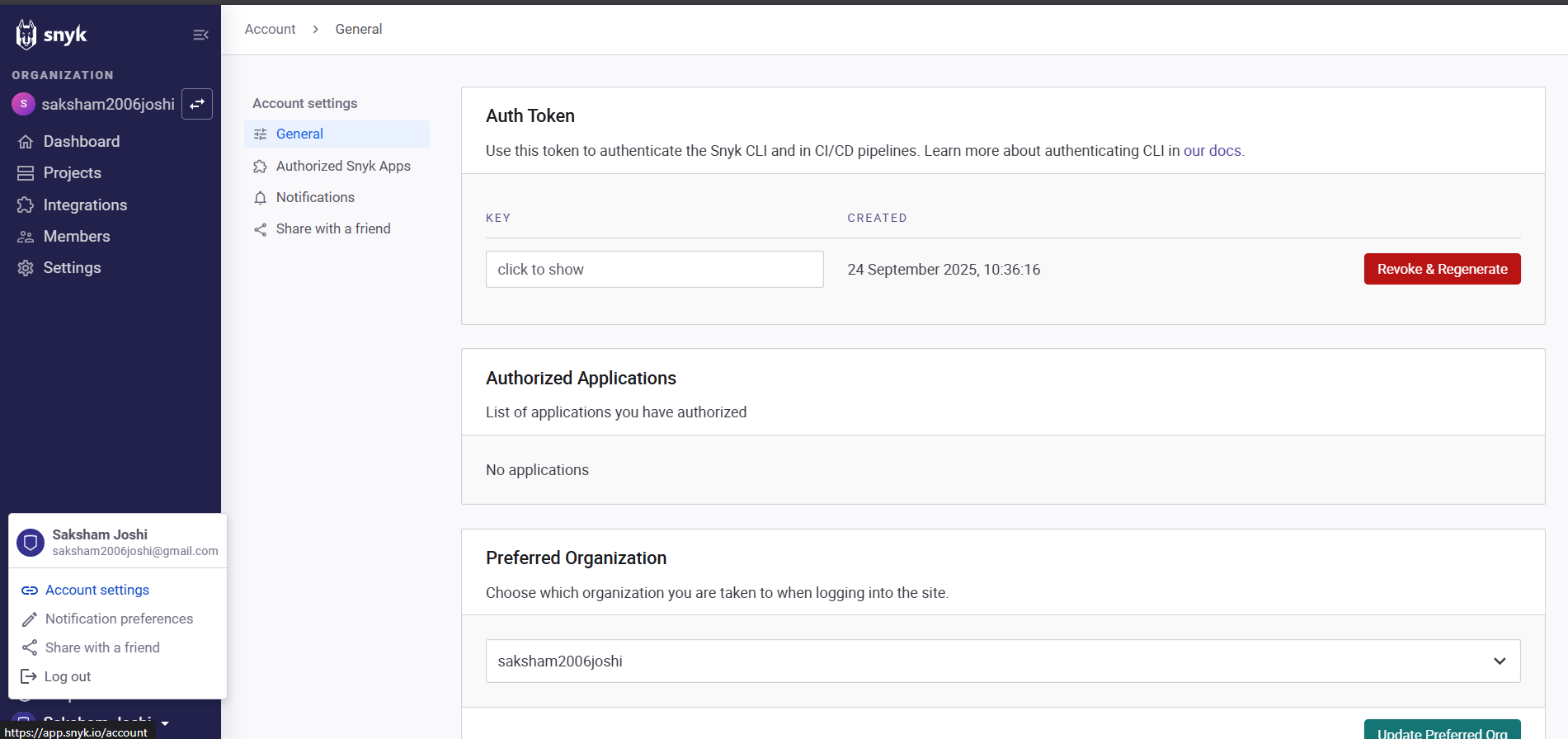


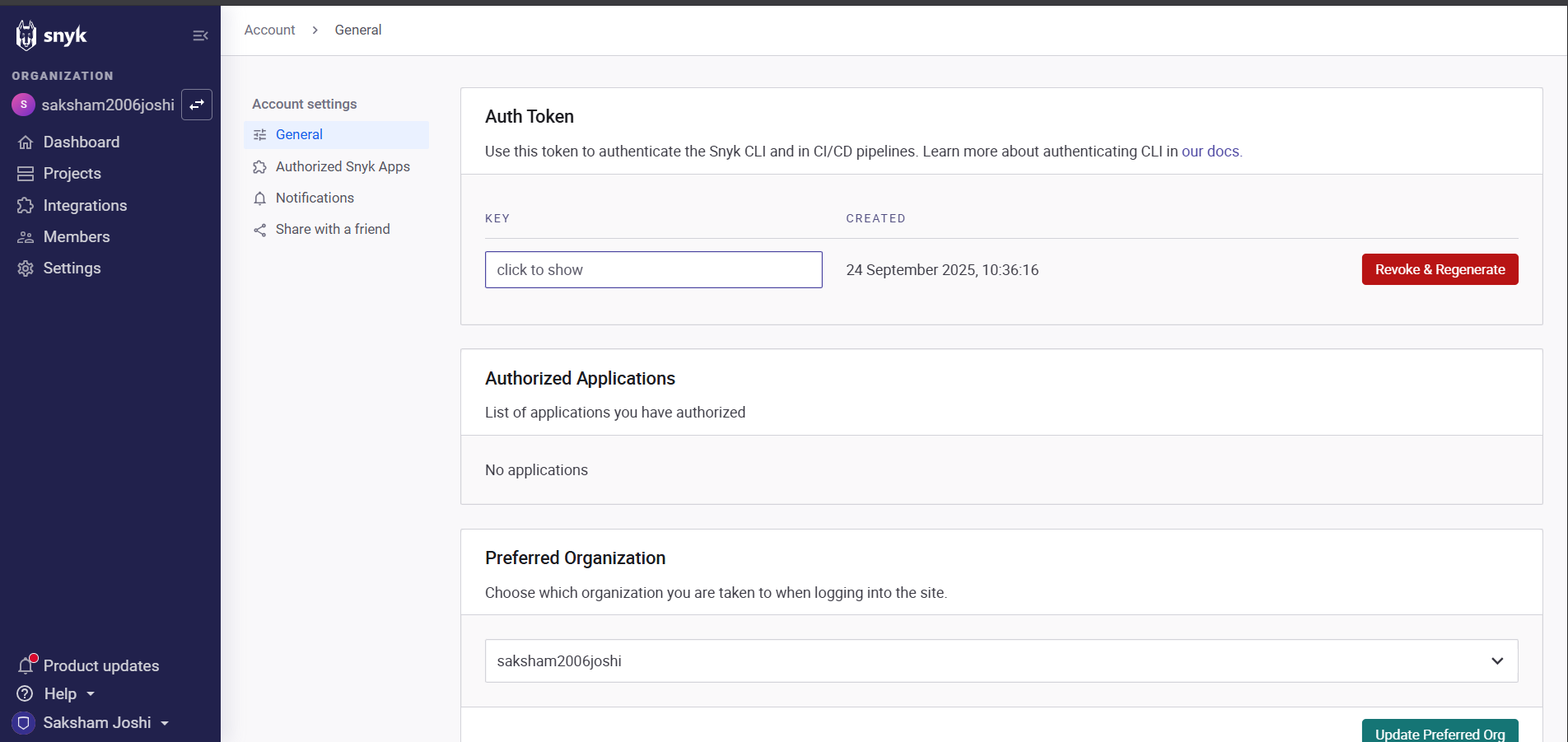
1. To add Snyk, click on **Add Snyk** under **Snyk Installations,** add **Name** as **Synk,** and clickonthe **Save** button

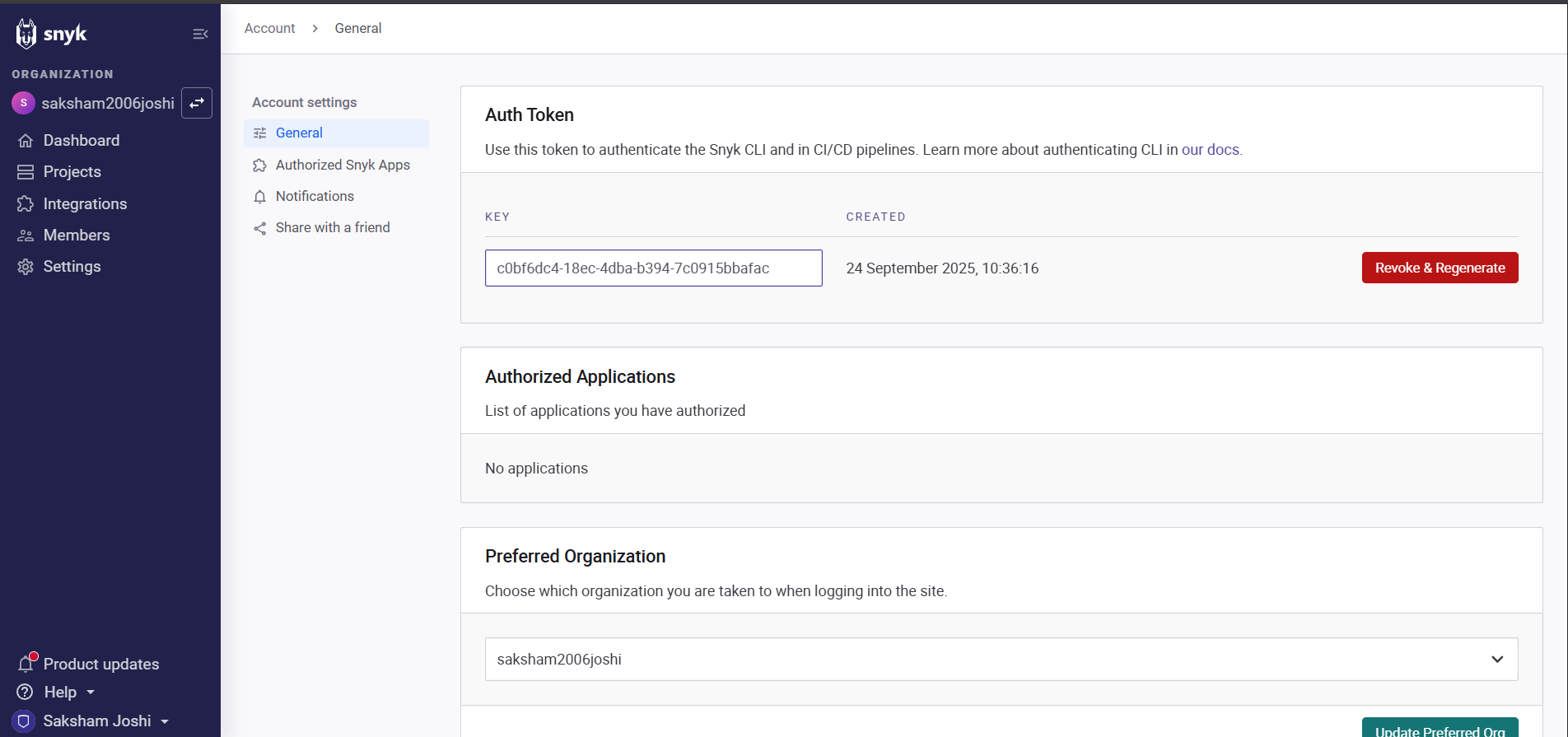


**Step 3: Manage Snyk API and Jenkins credentials**

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







1. In the Jenkins interface, go to **Manage Jenkins,** select **Security**,thenchoose **Credentials** and select **global** to add global credentials

A screenshot of a computer

Description automatically generated

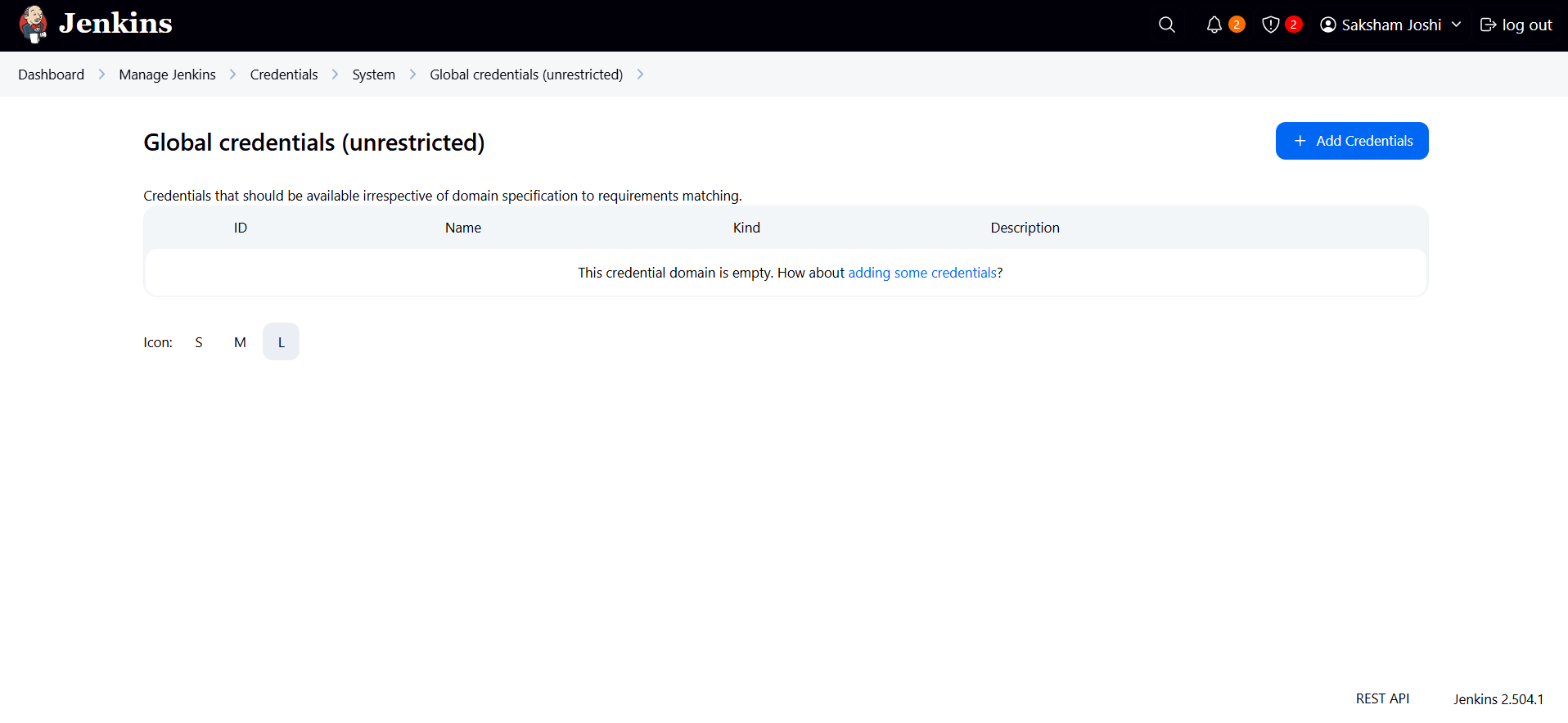
A screenshot of a computer

Description automatically generated

A screenshot of a computer

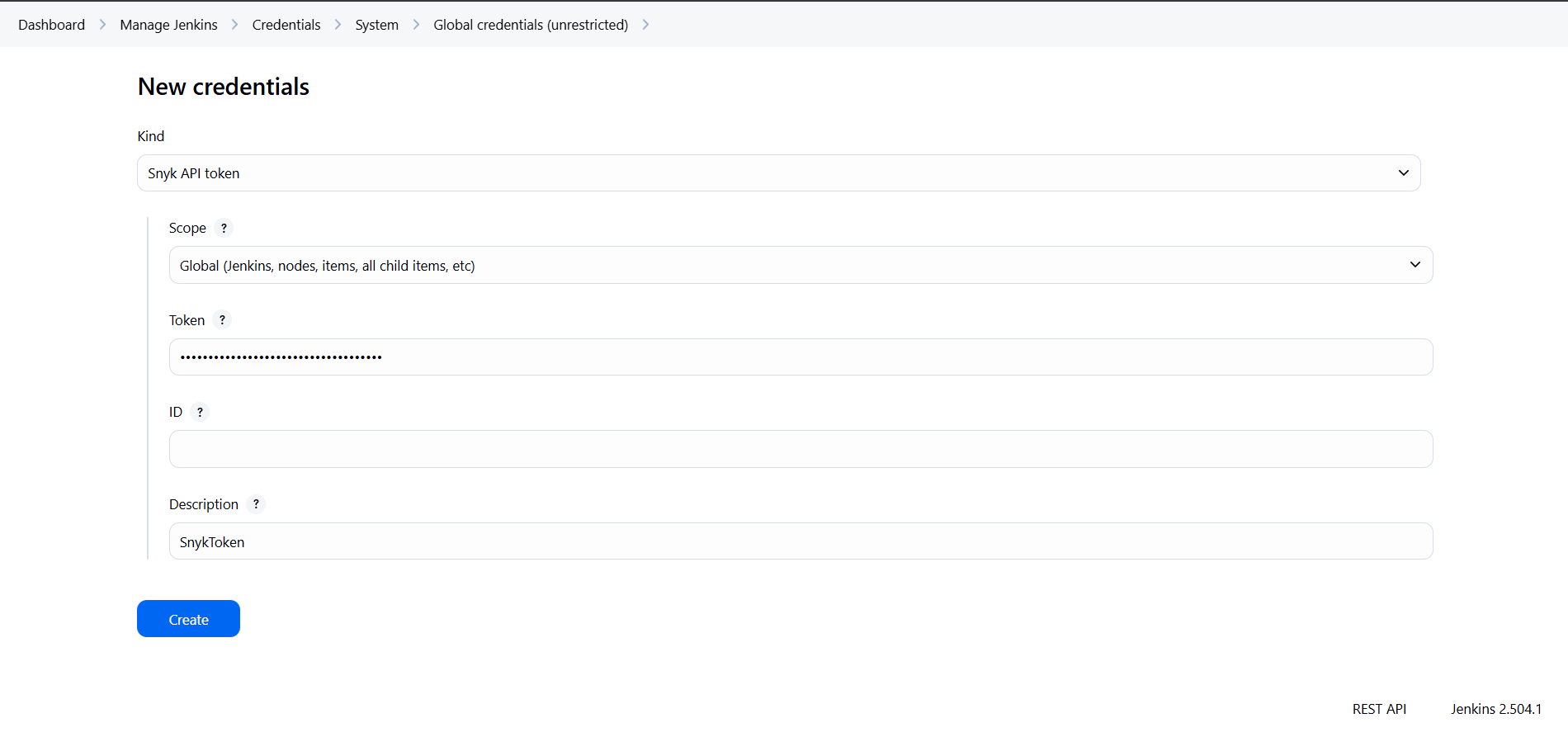
Description automatically generated

1. 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



A screenshot of a computer

Description automatically generated

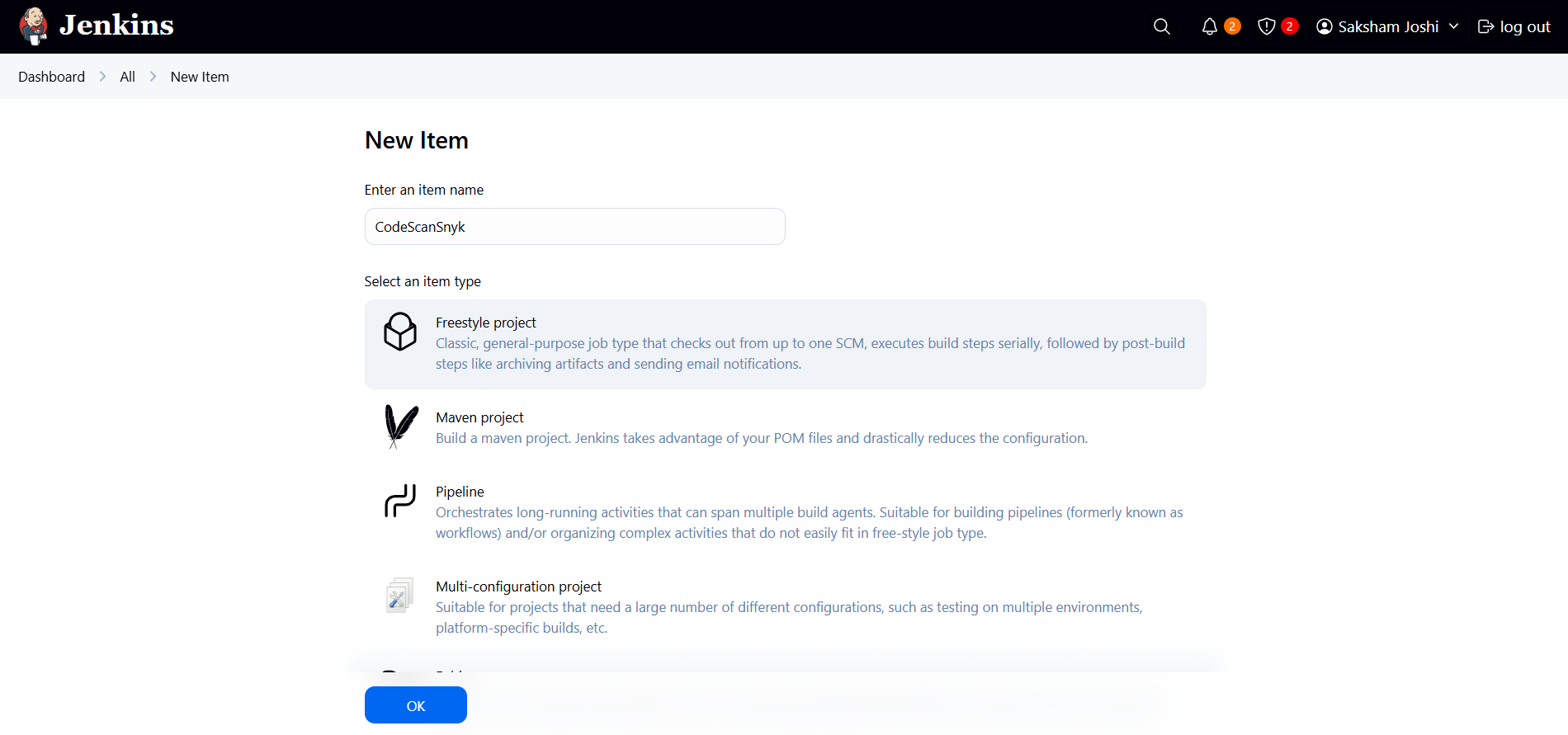


**Step 4: Configure the Jenkins job for scanning**

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

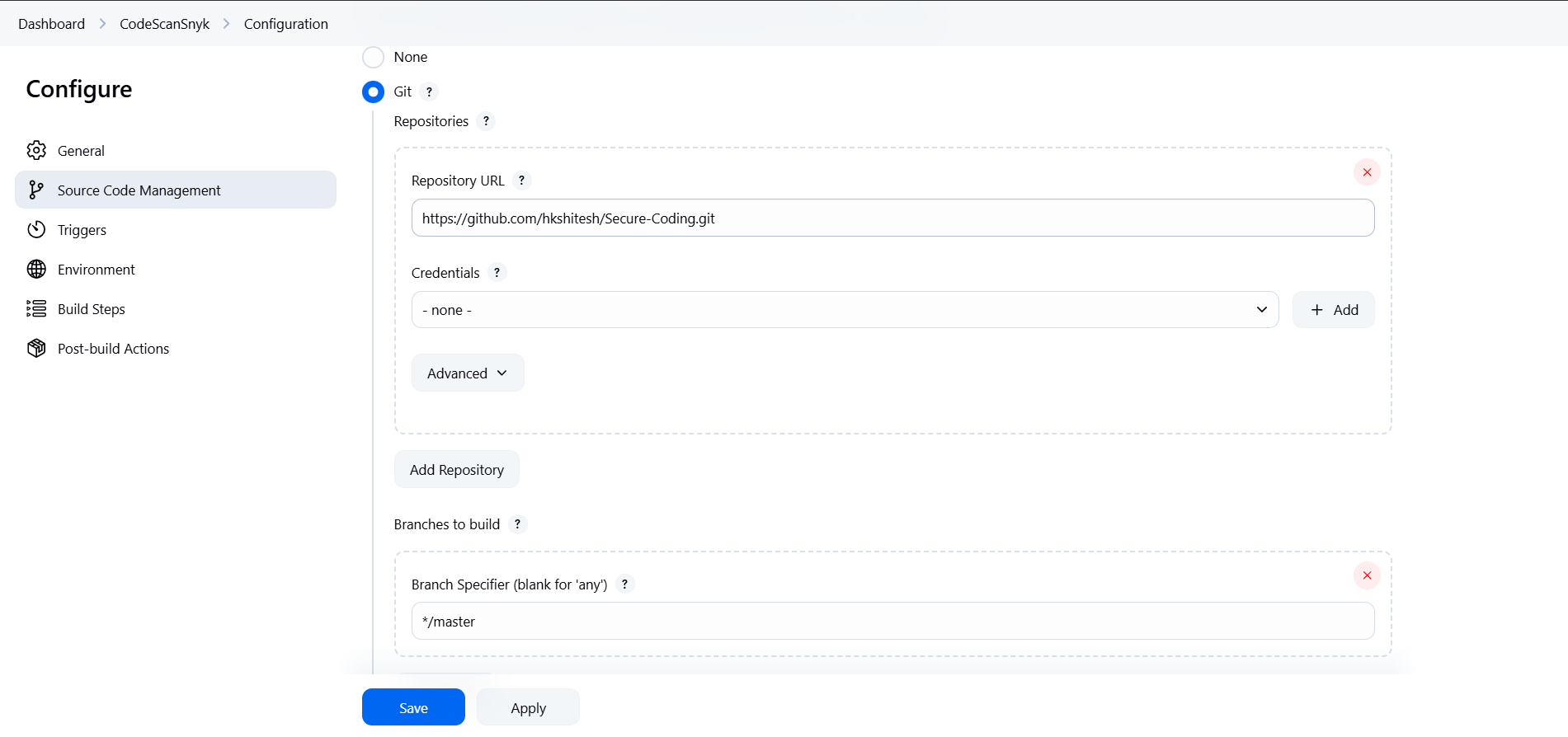
**A screenshot of a computer

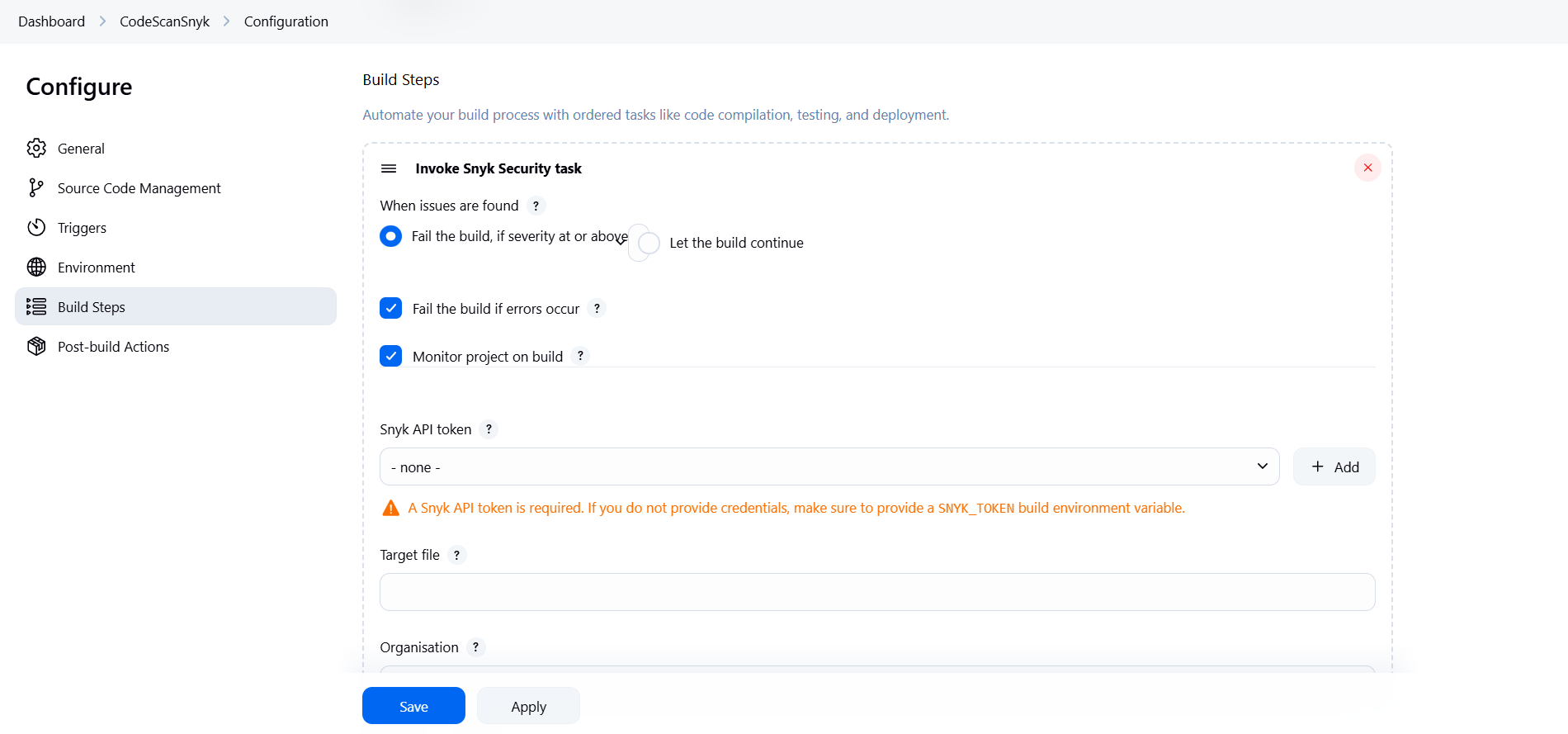
Description automatically generated**

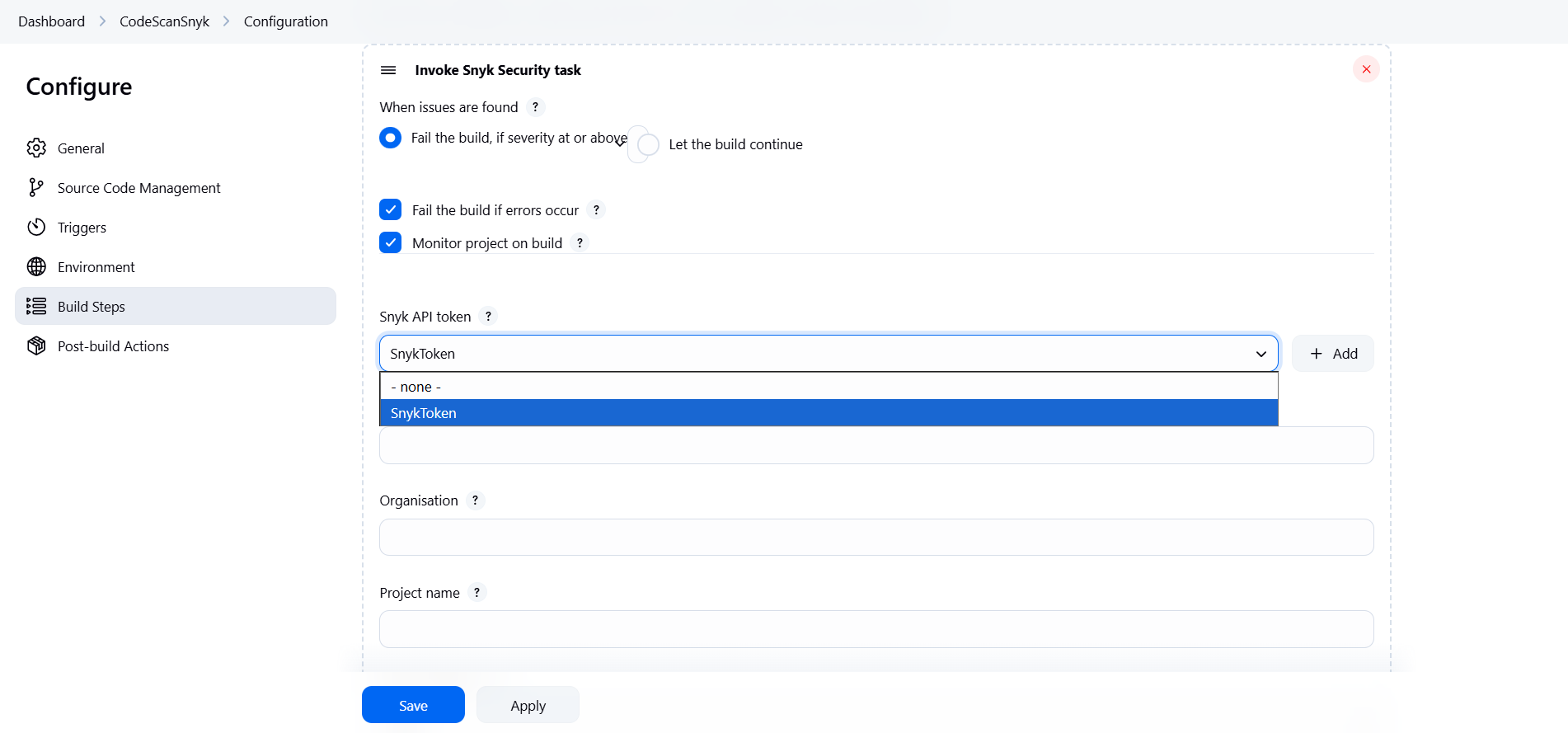


1. After creating a job, go to **Source Code Management** and enter the GitHub repository URL. Then, under **Build Steps**, add the build step **Invoke Snyk Security task** with the name **SnykToken**. Finally, click the **Save** button to create the build.

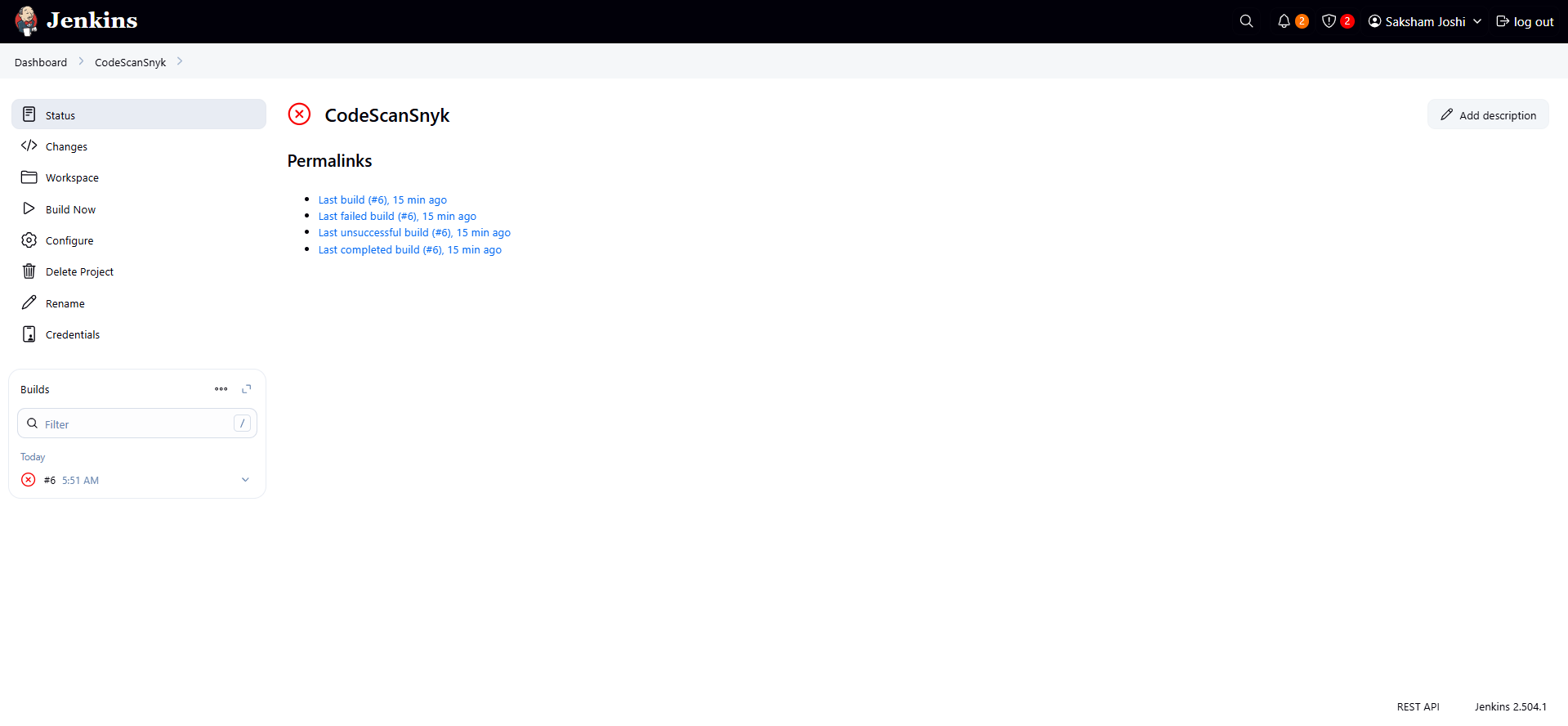
Use GitHub Repo: **https://github.com/hkshitesh/Secure-Coding.git**

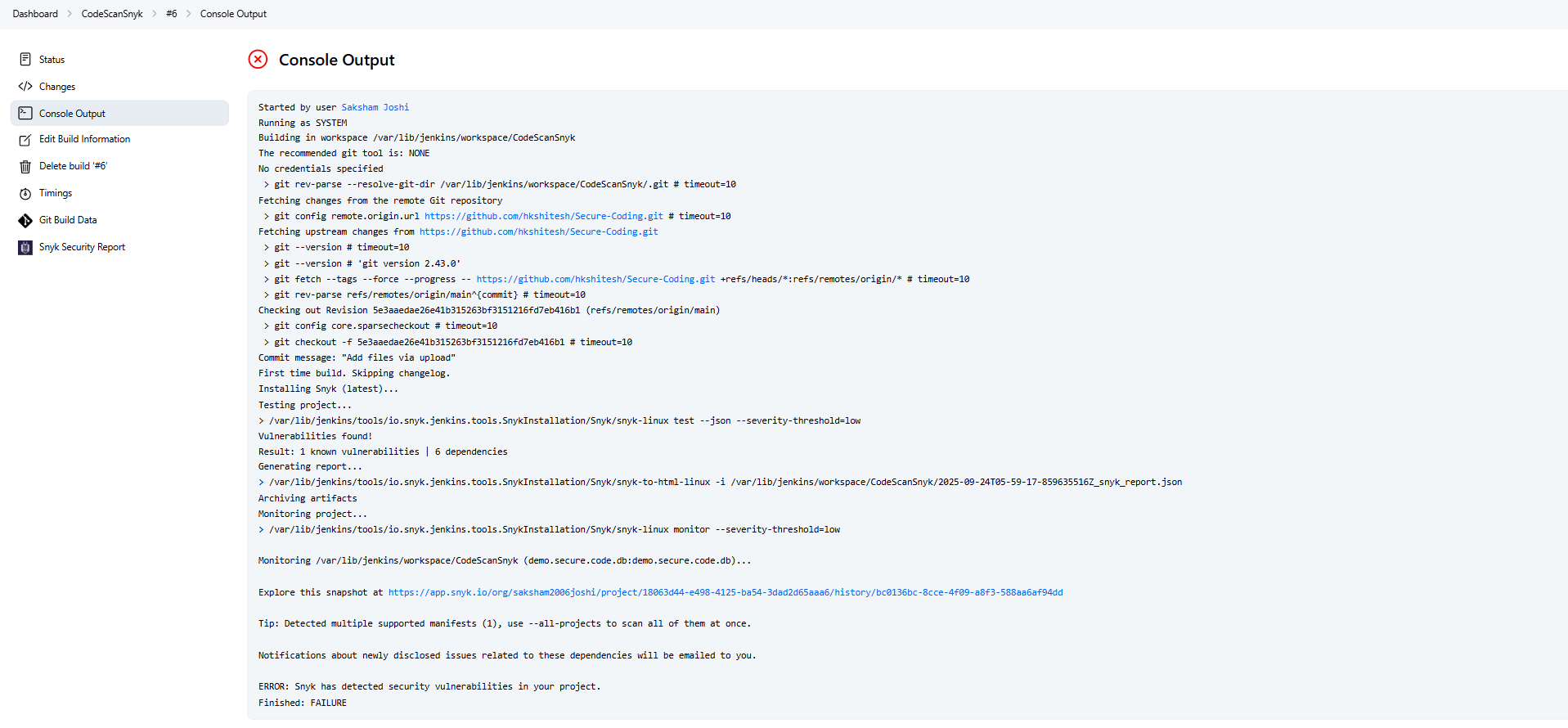






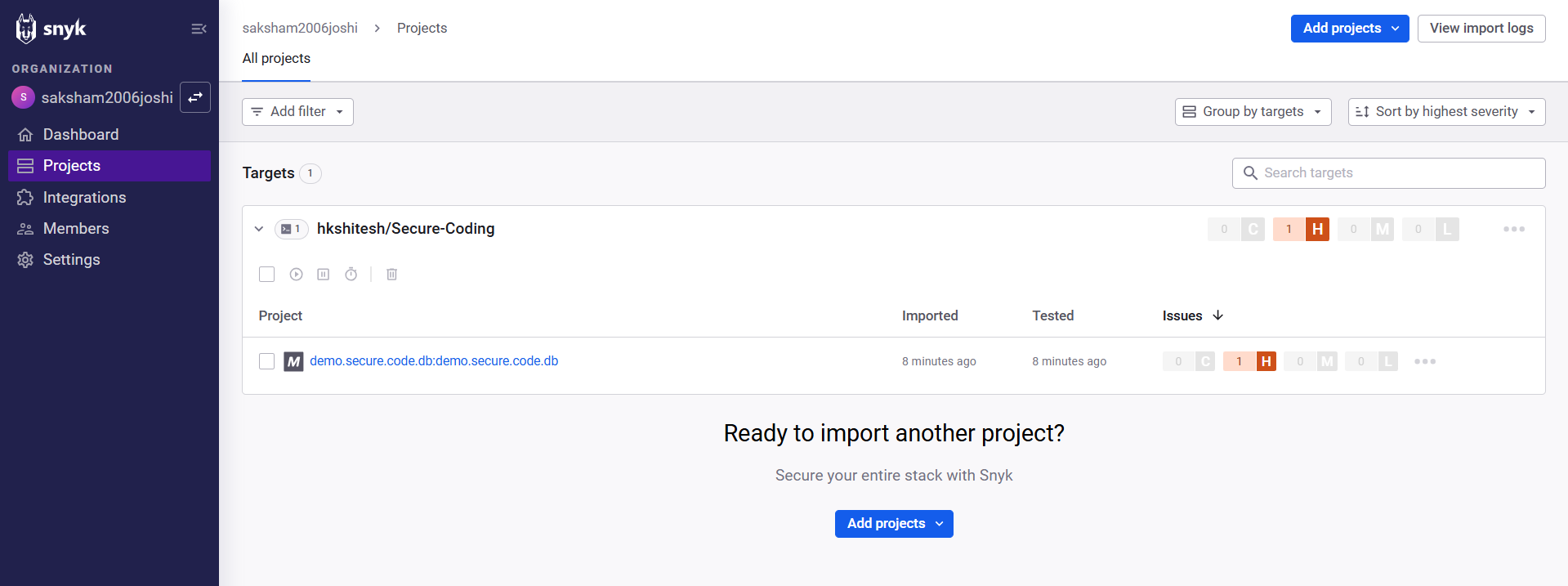
1. To check the build status, click on the build link under **Permalinks.** After that, click on **Console Output**





Build Failed! Vulnerabilities found by Snyk- **Snyk Plugin is working successfully**

1. 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.