**Lesson 01 Demo 03**

**Running SAST Scan on a Static Web Application   
Using GitHub**

**Objective:** To run a SAST scan on a static web application using GitHub workflows integrated with Semgrep for secure code analysis and validation

**Tools required:** GitHub Actions and Semgrep SAST tool

**Prerequisites:** None

Steps to be followed:

1. Fork the GitHub repository
2. Set up a workflow
3. Connect the GitHub account with Semgrep
4. Validate the SAST scan

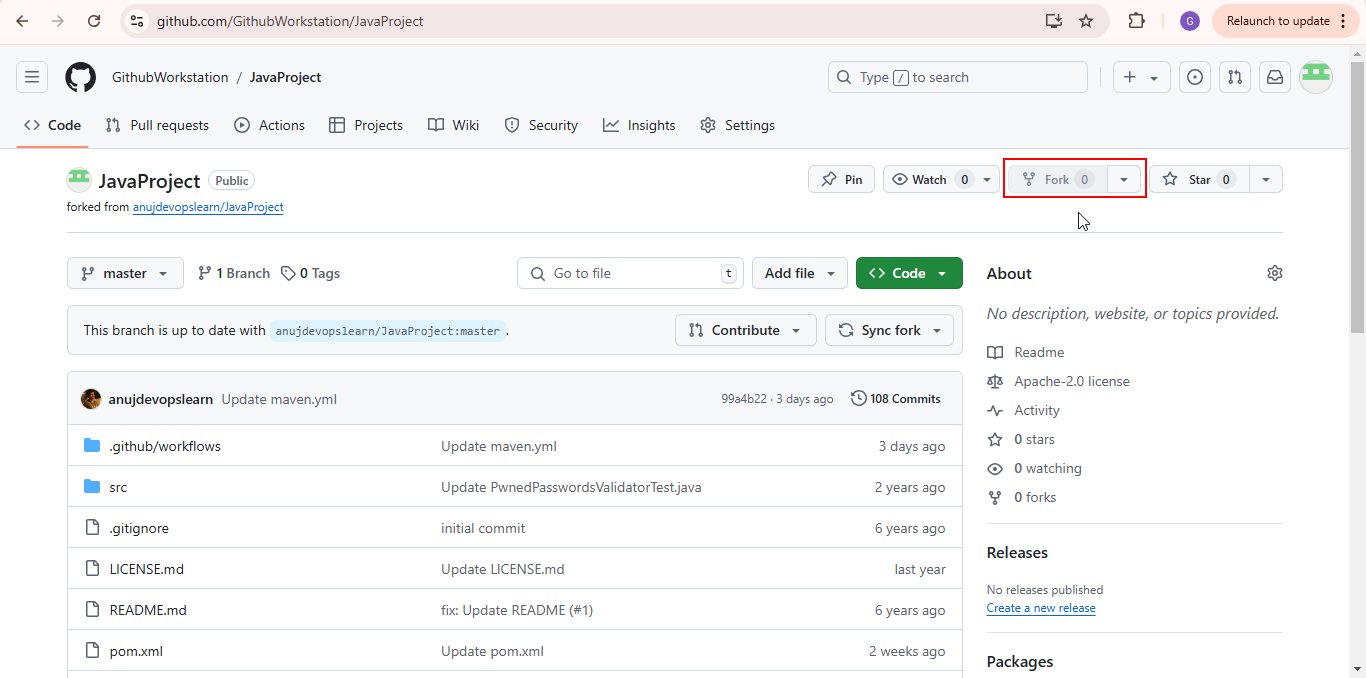
**Step 1: Fork the GitHub repository**

* 1. Sign in to your GitHub account using the following URL:  
     **https://github.com/login**

**A screenshot of a computer

Description automatically generated**

* 1. Fork the following repository:  
     **https://github.com/GithubWorkstation/JavaProject**

****

**Step 2: Set up a workflow**

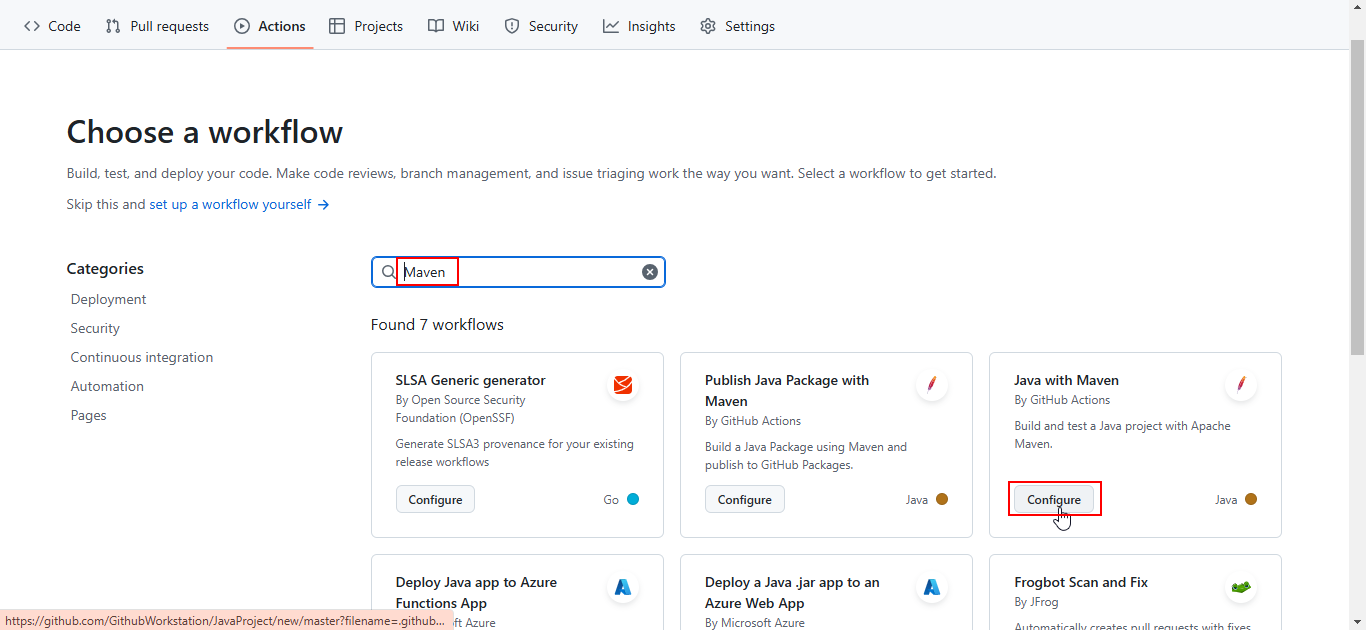
* 1. Click on the **Actions** tab

A screenshot of a computer

Description automatically generated

* 1. Search for **Maven** and click on **Configure** in the **Java with Maven** workflowas shown below:

Create a workflow



* 1. Remove the optional step from the workflow as shown below:

A screenshot of a computer

Description automatically generated

* 1. Search for and select **semgrep-action** from **Marketplace** to proceed with integration in the workflow  
       
     A screenshot of a computer

     Description automatically generated

|  |
| --- |
| **Note:** The access token will be generated in the subsequent steps. |

**Step 3: Connect the GitHub account with Semgrep**

* 1. Navigate to Semgrep website using the following link:  
     **https://semgrep.dev/**

A screenshot of a computer

Description automatically generated

* 1. Click on **Try for free**

A screenshot of a computer

Description automatically generated

* 1. Click on the **Sign in with GitHub** buttonA screenshot of a computer

     Description automatically generated

* 1. Click **Authorize semgrep-app**  
       
     A screenshot of a computer

     Description automatically generated
  2. Add the organization’s name and click **Create new organization**  
       
     A screenshot of a computer

     Description automatically generated
  3. Select the required option and click **Continue**  
       
     A screenshot of a computer

     Description automatically generated
  4. Select **GitHub** to scan the code  
       
     A screenshot of a computer

     Description automatically generated
  5. Click on **Install GitHub App**  
       
     A screenshot of a computer

     Description automatically generated
  6. Select **All repositories**A screenshot of a computer

     Description automatically generated
  7. Click **Install and Authorize**   
       
     A screenshot of a computer

     Description automatically generated
  8. Select the account and click **Continue**A screenshot of a computer

     Description automatically generated
  9. Click **Register GitHub App**   
       
     A screenshot of a computer

     Description automatically generated
  10. Click **Create GitHub App for GithubWorkstation** as shown below:  
        
      A screenshot of a computer

      Description automatically generated
  11. Click on **Install**   
        
      A screenshot of a computer

      Description automatically generated
  12. Click on **Setup projects**

A screenshot of a computer

Description automatically generated

**Step 4: Validate the SAST scan**

* 1. Click **Settings** and then click on **Tokens** as shown below:

A screenshot of a computer

Description automatically generated

* 1. Click on **Create new token**

**A screenshot of a computer

Description automatically generated**

You will see the following interface:  
  
A screenshot of a computer

Description automatically generated

|  |
| --- |
| * 1. **Note:** Copy the **Secrets value** |

* 1. Copy the following Semgrep configuration and paste it into the **maven.yml** file, and then click on **Commit** **changes…**   
       
     **# This workflow will build a Java project with Maven, and cache/restore any dependencies to improve the workflow execution time**

**# For more information see: https://docs.github.com/en/actions/automating-builds-and-tests/building-and-testing-java-with-maven**

**# This workflow uses actions that are not certified by GitHub.**

**# They are provided by a third-party and are governed by**

**# separate terms of service, privacy policy, and support**

**# documentation.**

**name: Java CI with Maven**

**on:**

**push:**

**branches: [ "master" ]**

**pull\_request:**

**branches: [ "master" ]**

**jobs:**

**build:**

**runs-on: ubuntu-latest**

**steps:**

**- uses: actions/checkout@v4**

**- name: Set up JDK 17**

**uses: actions/setup-java@v4**

**with:**

**java-version: '17'**

**distribution: 'temurin'**

**cache: maven**

**- name: Build with Maven**

**run: mvn -B package --file pom.xml**

**semgrep:**

**# User definable name of this GitHub Actions job.**

**name: semgrep/ci**

**# If you are self-hosting, change the following `runs-on` value:**

**runs-on: ubuntu-latest**

**container:**

**# A Docker image with Semgrep installed. Do not change this.**

**image: semgrep/semgrep**

**steps:**

**# Fetch project source with GitHub Actions Checkout. Use either v3 or v4.**

**- uses: actions/checkout@v4**

**# Run the "semgrep ci" command on the command line of the docker image.**

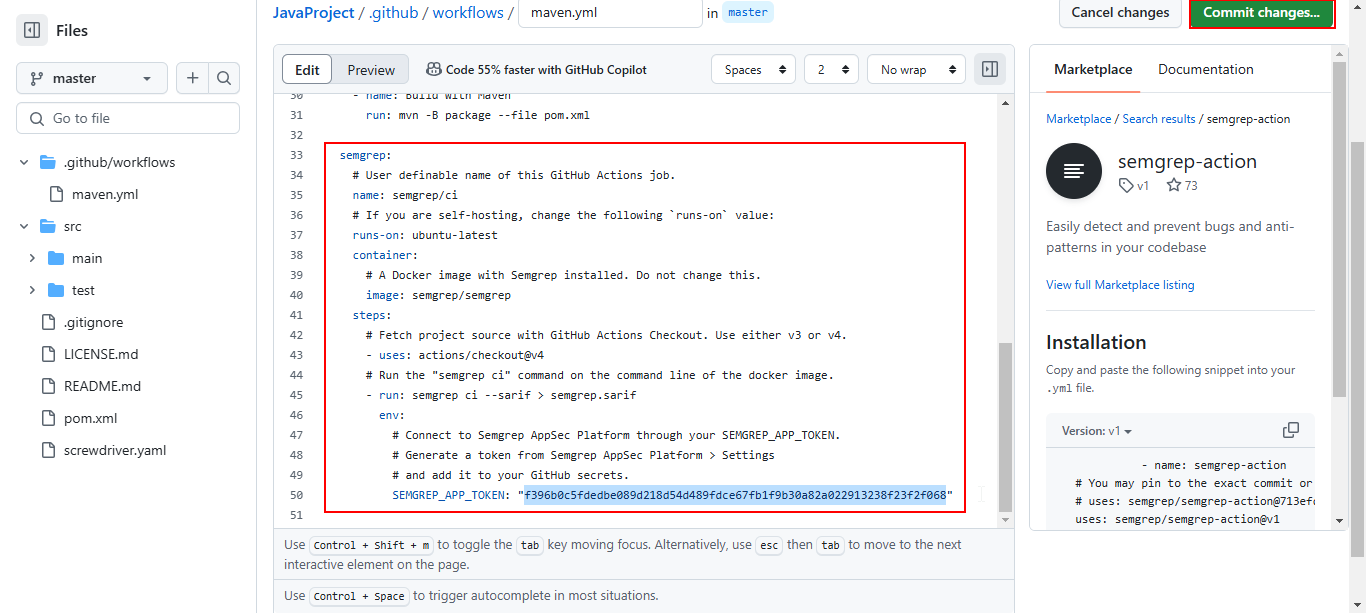
**- run: semgrep ci --sarif > semgrep.sarif**

**env:**

**# Connect to Semgrep AppSec Platform through your SEMGREP\_APP\_TOKEN.**

**# Generate a token from Semgrep AppSec Platform > Settings**

**# and add it to your GitHub secrets.**

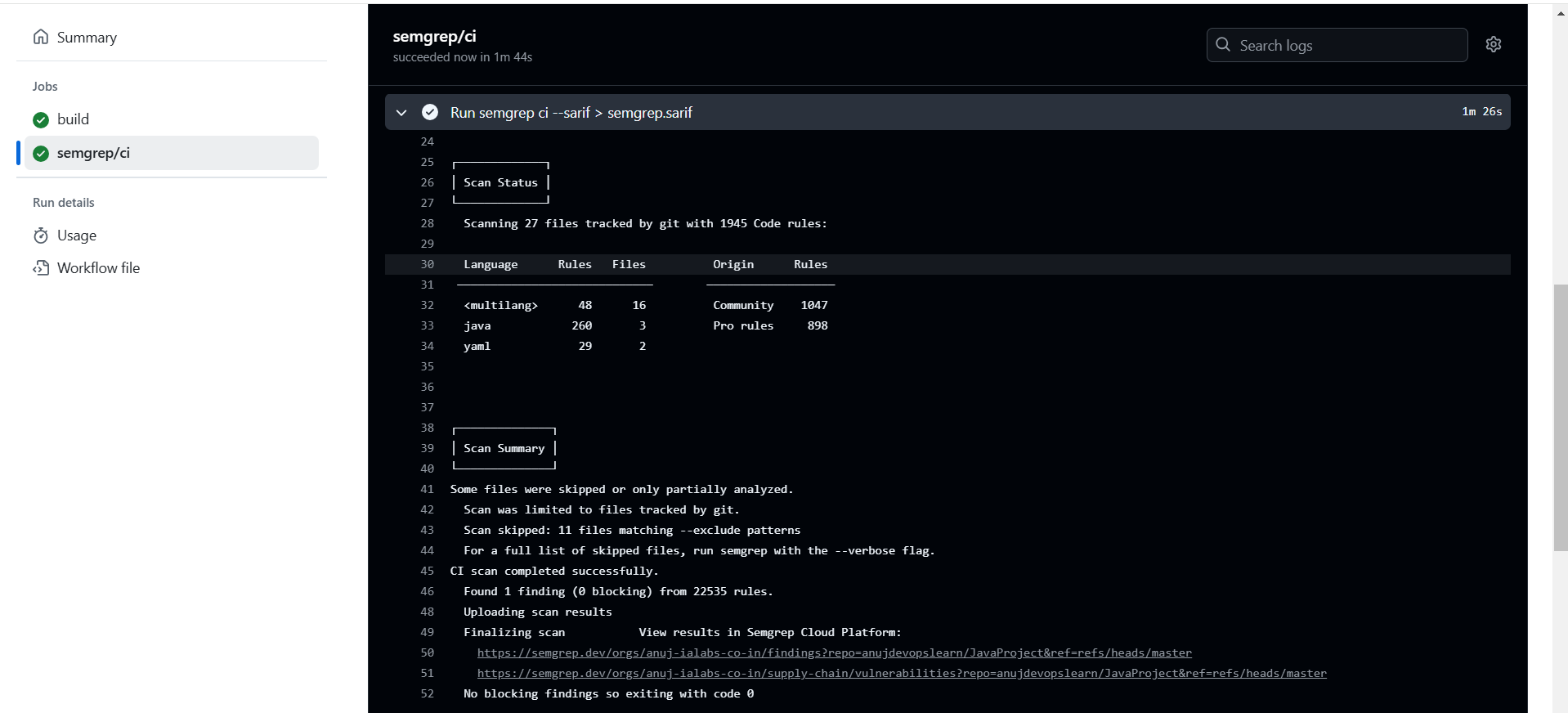
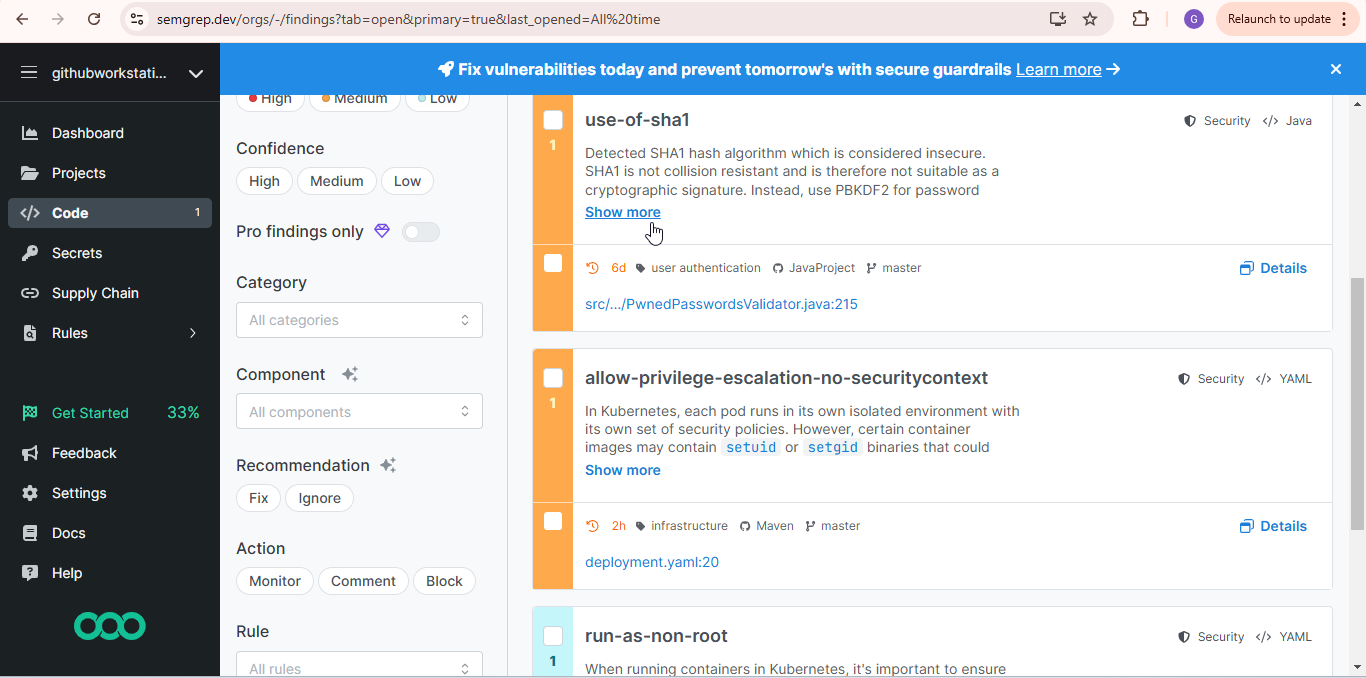
**SEMGREP\_APP\_TOKEN: "YOUR\_TOKEN"**  
  


|  |
| --- |
| **Note**: Replace **YOUR\_TOKEN** with the token copied in the previous step |

You will see that the pipeline has been invoked successfully.  
  
A screenshot of a computer

Description automatically generated

* 1. Click **Update maven.yml**, then select **semgrep/ci**, and choose the run configurations as shown below:  
       
     A screenshot of a computer

     Description automatically generated
  2. Verify the scan status and summary  
       
     
  3. Navigate to Semgrep website to view scan results  
       
       
       
     A screenshot of a computer

     Description automatically generated

By following these steps, you have successfully executed a SAST scan on a static web application using GitHub workflows and integrated Semgrep for secure code analysis.