Lab Assignment 6

Course: CS202 Software Tools and Techniques for CSE
Lab Topic: Evaluation of Vulnerability Analysis Tools using CWE-based Comparison
Date: 8th September 2025

Objective

The purpose of this lab is to explore how different vulnerability analysis tools (static application security testing tools) detect software weaknesses in real-world projects. Students will compare selected tools based on CWE (Common Weakness Enumeration) categories and analyze pairwise agreement/disagreement values between tools using metrics such as Intersection over Union (IoU).

Learning Outcomes

By the end of this lab, students will be able to:

- ✓ Apply multiple vulnerability analysis tools on real-world projects.
- ✓ Extract CWE-based findings from security tools.
- ✓ Compare tools based on CWE coverage (breadth of vulnerabilities detected).
- ✓ Compute and interpret pairwise IoU values between tools.
- ✓ Visualize and analyze CWE-based vulnerability detection results.

Pre-Lab Requirements

- Any Operating System (Windows, Linux, MacOS, etc.), Python 3.10 or later.
- **Read**: Comparison and Evaluation on Static Application Security Testing Tools (FSE 2023)

Lab Activities

- ❑ Choose **THREE large** scale open-source repositories to analyze. Make sure this is a **real-world** project and not toy projects on GitHub. You must not reuse the same repository that you may have already selected in your previous assignments.
- ❑ **Define Selection Criteria:**
  - ➢ Establish your own criteria for selection (inclusion/exclusion) of repositories and include this information in your report. Basically, you need to specify how you settled with the final set of selected repositories. Recall the hierarchical funnel diagram from the slides from Lecture 2.
  - ➢ Examples of selection criteria may include metrics such as the number of GitHub stars, forks, etc.
  - ➢ You may use the SEART GitHub Search Engine to perform this task.

Note: Please reach out to the TAs for any queries/issues.

- □ From the list of static analysis tools, select **THREE** tools that are categorized under vulnerability detection. They should explicitly support CWE-based vulnerability reporting.
- □ **Run Vulnerability Tools on Project and Collect Pairwise Findings:**
  - ➢ For each project, run all selected vulnerability analysis tools.
  - ➢ Export the results of each scan in a structured format (JSON, CSV, or XML).
  - ➢ Ensure that the exported data includes CWE IDs for each finding.
  - ➢ For each **project–tool combination (pair)**, aggregate findings (counts) for each detected CWE ID. Additionally, indicate whether this vulnerability is one of the "top 25 CWE categories" (e.g. CWE Top 25[1]). Store information in a single consolidated CSV file as follows:

Project_name	Tool_name	CWE_ID	Number of Findings	Is_In_CWE_Top_25?
…	…	…	…	…
…	…	…	…	…

- □ **Tool-level CWE Coverage Analysis:**
  - ➢ From the CSV (in prev. step), extract the **set** of CWE IDs (unique) detected by each tool.
  - ➢ Compute Top 25 CWE coverage (%) at the tool level.
  - ➢ Visualize coverage with appropriate graphs and plots as necessary.
- □ **Pairwise Agreement (IoU) Analysis:**
  - ➢ Compute IoU for each tool pair using the formula for Jaccard Index (IoU):
    IoU (T_1, T_2) = |{CWE IDs found by **both T_1 and T_2**}|/ |{CWE IDs found by **T_1 or T_2**}|
  - ➢ Create a Tool × Tool IoU Matrix,
    - **Build a square matrix** with tool names as rows and columns as shown below. Cells contain the IoU value for a pair of tools.

	Tool_A	Tool_B	Tool_C
Tool_A	…	…	…
Tool_B	…	…	…
Tool_C	…	…	…

    - **Interpret the Matrix**: along the lines of Jaccard Index (IoU), what insights can you obtain about similarity/diversity of tools based on the vulnerability analysis results?

[1] You can get the list here: [https://cwe.mitre.org/top25/archive/2024/2024_cwe_top25.html]

Note: Please reach out to the TAs for any queries/issues.

- **Explain the Matrix**: which tool combination maximizes CWE coverage? What are the takeaways of your analyses?

Resources

1. Lecture 5
2. https://cwe.mitre.org
3. https://cwe.mitre.org/top25/archive/2024/2024_cwe_top25.html
4. https://en.wikipedia.org/wiki/Jaccard_index

Note: Please reach out to the TAs for any queries/issues.