Course Name: Secure Software Engineering
Course Code: COMP SCI 4412/7412
Assessment Component: Assignment 1 (15%) – Individual Assessment
Release Date: 2/08/2023
Due Date: 16/08/2023 by 23:55.
Submission: MyUni

The list of tasks for assignment 1 is detailed below.

Part 1 (8 marks)

Please visit the link for Assignment 1 Part 1 in MyUni to input your identified vulnerable source code files **as soon as possible** after you find them. **The student who submits earlier will claim the authorship of the source code file and the later ones must choose a different file to work on.**

1. Study about **Cross-site scripting (XSS)** and **Cross-site Request Forgery (CSRF)** vulnerabilities on Common Weakness Enumeration and related websites. You DO NOT have to submit this part.
2. Identify 3 source code files in open-source GitHub repositories. Each type of vulnerability must have at least one source code file. The projects must satisfy the following conditions:
    - The programming languages must be either Java, JavaScript, or PHP
    - The repository has more 100 stars and 10 contributors on GitHub
3. Include the following artifacts about each file you have found in the report:
    - Link to the file
    - Link to the commit that fixes the vulnerable file
    - Name of the file
    - The programming language used in the file
    - Name of the repository
    - Number of repository stars
    - Number of contributors in the repositories
    - Type of vulnerability (CWE)
4. Pinpoint the code lines within the source code files you have identified that contain the vulnerabilities you found.
5. Also enter the information you have found in tasks 3 and 4 into the Assignment 1 Part 1 link along with your name and student ID to avoid duplicate submission
6. Explain how the vulnerable lines correlate to the definition or causes of the vulnerability you have studied
7. Show how to fix the vulnerability and explain in detail. It is not mandatory that the fix has to be executable, but the explanation must be reasonable. If there is already a fix available, explain how this fix complies with the standard mitigation techniques for the vulnerability.
8. Write your findings in the report.

9. Please visit the Assignment 1 Part 1 link to input your identified vulnerable source code files **as soon as possible** after you find them. You can do the analyses and put your findings in the report later (but still before the deadline). The student who submits earlier will claim the authorship of the source code file and the later ones must choose a different file to work on. In case you accidentally select the same source code file, there will be a red flag to notify you.

Part 2 (7 marks)

1. Visit the website and study about Common Vulnerabilities and Exposures (CVE), National Vulnerability Database (NVD), Common Weakness Enumeration (CWE), Common Vulnerability Scoring System (CVSS). This part helps you to gain background about the security vulnerabilities. You do not have to submit this part.
2. Install and learn how to use Git commands (e.g., git log, git show, git diff). You **DO NOT** have to submit this part.
3. Register a GitHub account or reuse your existing one.
4. Visit the link of repos assigned to you on MyUni to get your assigned security vulnerability along with the fixing commit in software systems.
5. Determine the CWE (type) of each vulnerability. **Include the CWE** you have found **in the report**.
6. Study about the process of how to go from CVE-ID to the corresponding GitHub repository for assigned vulnerability. **Describe step-by-step in the report** how you have found to go from vulnerability to its software repository.
7. Identify and describe the bug report in the issue tracking system (e.g., Jira, BugZilla, GitHub repository itself) that reports about the fix of the vulnerability. **Include the screenshot, link, the fixing commit and your comments about the status** of the vulnerability **in the report**
8. Compare the fixing commits you have identified for the vulnerability with the ones provided for you in the link above. If they are not matched, you **have to explain in detail in the report**. **Also put the results in the template table** we have given to you besides this file. **Please copy the table into your report.**
9. Imagine you are a developer responsible for a vulnerable project and you have found your vulnerability. And you are going to report the vulnerability to NVD to include in their database, you may need to suggest to them an assessment of each vulnerability based on CVSS 2.0 (commonly used version) and CVSS 3.0 (new version). **Include your CVSS (versions 2 and 3.0) metrics for each vulnerability with detailed explanation. Then, compute the base scores of CVSS versions 2 and 3.0 and compare them with the ones provided on NVD and provide your reflections. Make sure you compute the scores using your reasoning first without looking at NVD. If there is no available score on NVD, then you can skip the comparison step. Also provide your reflections on the change of the CVSS base scores from version 2 to version 3.**
10. Summarize your findings for the above tasks in your report.

The report of this activity should be in an A4-size page with Times New Roman or similar font size 12. The first page of the report should include your full name and student ID along with the vulnerabilities you have chosen and the list of three vulnerabilities we have assigned to you.

Tips about how I would go about doing this assignment:

Part 1

I will first study the vulnerabilities mentioned in the task on the Common Weakness Enumeration website. Google is also always worth a try if I want to explore more. Then, I will try to use the name of the vulnerability and search it on GitHub. After I find the repositories, I will filter them using the above criteria. Then, I will focus on the vulnerable files and analyze them line-by-line or use existing tools. If there is already a fix for that vulnerability, I will include it in my report. Otherwise, I will try to see how I can fix it using the mitigation techniques I have learned for the vulnerability. I will explain how my findings match with the materials I have learned for that vulnerability. Most importantly, I will try to input my identified source code files at Google Sheets as soon as I find them to avoid duplication and having to search again.

Part 2

I would first study about CVE, NVD, CWE and GitHub to see how they link with each other. After I understand their connection, I can identify CWE (task 2.5) and describe the process for the vulnerability (task 2.6). Then, I can start searching for bug reports in the suggested locations above. If you cannot find it there, Google is always worth a try. After I found the bug report, I would try to find whether the developers/testers mention the link they fixed that vulnerability. That would likely be my fixing commit. Finally, I would compare the one I have found with the provided one. If it does not match, then I try to investigate the provided commit to see how it is related to the vulnerability I am working on and also to my identified commit. To do this investigation step, cloning the GitHub repositories locally is a good way to go. For task 2.9, I will recall how to assess a vulnerability using CVSS in the second Seminar Session given by Jason. Then, I will use my experience and reasoning to fill in the value for each metric and then compute the score using the CVSS calculator. I will then compare my scores with the ones on NVD if they are available and give my detailed reflections. Then, I will describe and explain in detail each task for each vulnerability in the report. I will also fill in the Excel file the information I found. Finally, I will submit the 2 files to Canvas.

Please note that answers without explanation would not receive any point.

How to Submit: *The assignment will be submitted via MyUni as there is an upload facility created for this assignment on MyUni.*

This assignment is designed to help you to achieve the learning outcomes # 5 and 6 in the course outline.