Fix Lab 11: Secure Deployment

Date Due: Tuesday, November 30, 2021 by 11:59 PM

Introduction

In this lab assignment, you will find and fix issues related to Debug Code and Secure Deployment. This assumes you have completed both Lab 0: Lab Setup Guide. For this lab you will need the ability to run the WebGoat.NET application, as well as debug it using Visual Studio 2019 Community Edition. You will also need to have completed the Git setup outlined in Lab 0 and the ability to push code changes to the remote repository on https://code.umd.edu.

For each of the following questions on the WebGoat.NET application, you will need to find and fix the weaknesses found in the source code. Along with this, you will submit a writeup containing the following:

Question Number.

a) Provide the URL of the WebGoat.NET application page where you are exercising the question. **Do not** include any query parameters or any other special characters in the answer.
b) Provide the CWE-ID, Filename, Line Number of the weakness identified based on the question.
c) For the given identified weakness in the previous step, describe how you intend to fix the weakness. This can be as detailed as possible to explain how it will address the weakness.
d) Describe why your intended fix will address the weakness (either directly or indirectly) and whether to your knowledge it will prevent future attacks. This can be as detailed as possible to explain why it will address the weakness.
e) List all the paths with filenames you are changing to implement the fix for the weakness.
f) **IMPORTANT:** Implement the fix for the question under a branch with the following naming convention:
    a. Lab<Lab #>_Phase<phase #>. For example, you can run:
        i. `git checkout main`
        ii. `git checkout -b Lab2_Phase1 # This is for Lab 2 Phase 1 fix`
    b. Now you implement your fix and commit the changes locally:
        i. `git add .`
        ii. `git commit -m "Implemented Lab 2 Phase 1." # Commit with a message`
    c. Repeat earlier step as many times to get your fix working. Commit and push the newly created branch to the remote repository using:
        i. `git push origin Lab2_Phase1`
    d. **Identify** and **submit the commit id** (the long hexadecimal alphanumeric string from `git log --pretty=oneline`) as part of line item 'e' for the given question number in your writeup.
    e. Read https://git-scm.com/book/en/v2/Git-Basics-Viewing-the-Commit-History for more information.

Please NOTE: For the questions below, the word 'discuss' below refers to question items b) and c) from the list at the top of this lab handout and the word 'implement' below refers to question item e).

Important Note: For this lab, you will be finding weaknesses and implementing fixes based off of the '**SecureDeployment**' **branch** from the instructor's WebGoat.NET repository on code.umd.edu.

Phase 1: Debug Code

In the WebGoat .NET WebGoatCoins portal part of the application, there are several instances of Debug Code being inserted into the application.

1. Identify the instances where this occurs and implement the fixes. Answer the questions at the top of this lab handout after you are done.
2. Remember to commit and push your code under the Lab11_Phase1 branch.

Phase 2: Debug Logging

In the WebGoat .NET application under Error Handling and Logging, perform the Log Injection exercise.

1. First find where the weakness exists and then try and fix it. Answer the questions at the top of this lab handout after you are done.
2. Remember to commit and push your code under the Lab11_Phase2 branch.

Phase 3: Release Build

Assuming that the WebGoat .NET WebGoatCoins portal application is in a state to be released, perform the steps in Visual Studio to build a release package and publish it to a zip archive. You don't have to commit the Zip file archive as part of your code submission. Remove scripts and files that are **unnecessary** to the WebGoatCoins portal. Make sure to also minify and combine all scripts (CSS and JS) into site.css and site.js. You can follow the guide here: https://bartwullems.blogspot.com/2012/02/javascript-and-css-bundling-and.html. Remember, the scripts are included in the Resources/Master-Pages/Site.Master page.

1. Only answer items 'e' and 'f' for the questions above after performing the release build.
2. Remember to commit and push your code under the Lab11_Phase3 branch.

Submission Criteria

Please submit a writeup with:

1. Your name, UMD email ID and Lab Number.
2. The answers provided in the format given at the top of this lab handout.

Please only submit a **Word document** or a **PDF**. This lab contains code submission with branches for each phase. Please ensure the commits submitted are accessible by the auto grader.