The purpose of this assignment is to find security issues presented by the implementation of the open source project your group chose in the first assignment. This analysis will be conducted by performing a code review of selected files, with or without the use of a static analysis tool.

Choose three different software components of the application that you think are likely to contain security vulnerabilities. Components may be modules or files of source code. All three components should have access to assets that a threat would want to target. Conduct a code review of these components as a group and report on the security issues discovered.

1 Code Review

After selecting the software components to review as a group, one or more group members must prepare the code inspection package, consisting of the source code and static analysis results if possible. If the programming language of the project is supported by Fortify Source Code Analyzer, use that tool to do the static analysis. Otherwise, find an open source analyzer for your project. Once the project has been analyzed, create a ZIP archive containing the code to be reviewed along with static analysis results. Distribute the inspection package to each member of the group for reading and analysis before the code review meeting. Each member must bring a list of potential problems to discuss to the meeting.

Groups may have one or more code review meetings. During a code review meeting, group members should alternate roles as moderator, reader, and recorder for each software component being reviewed. There should be enough components for each group member to take on all or all but one of the roles for the review of at least one component.

2 Code Review Report

After the introduction, which summarizes the results of the code review, the sections of the report must be: Inspection Selection, Code Review Results, Code Review Checklist, and Conclusion. The three middle sections are described below:

1. Inspection Selection Describe your reasons for selecting the three software components that you chose to review. After doing your code inspection, re-evaluate your selection. What guidelines would you suggest for finding files for code review that are likely to contain problems?

2. Code Review Results After an overview of the results for all components, provide detailed results for each module in their own subsections. In each subsection, identify the component and which team member was the moderator, reader, and recorder for the review of that component. Explain what the code does, how it fits within the overall project, what assets are affected by the code, and what security impact a vulnerability in this module could produce. Describe each vulnerability in the code, providing
• Vulnerability type.
• Location (filename, line number, function/method)
• Impact of the vulnerability if exploited.
• Short description of how to mitigate the vulnerability.

3. **Code Review Checklist** Create a project specific checklist for future reviewers to use when inspecting modules of this project. The checklist should be based on the type of project (web, mobile, library, etc.), history of past vulnerabilities, assets, and functionality. Be sure to include the following types of concerns in your code inspection checklist:

• Domain-specific concerns
• Coding mistakes
• Past vulnerability mistakes
• Design concerns
• Availability concerns

The summary should recapitulate your findings and generalize upon the specific concerns you identified in the code inspections.

3 **Deliverables**

An electronic copy of the report must be sent to the instructor as an e-mail attachment in ODT or PDF format named a3-code-review.EXT where EXT is replaced with the appropriate extension suffix. The inspection package should also be attached to the e-mail as a ZIP archive named a3-code-review-package.zip. The e-mail must have a subject line of “CSC 666 Assignment 3: Code Review.”