




Activity: Secure Code Review

Overview

The goal of this activity is to perform a code review on example source code. We will be looking at Java code this time. The vulnerabilities and attacks we have looked at to date are applicable in any language.

Activity

1. Form groups of 3-5 students.
 - Choose one person to be the *moderator* to focus the discussion in certain parts of the code.
 - Choose one person to be the *scribe*.
2. Examine the [Controller.java](https://uncw.instructure.com/courses/16302/files/524241/download?wrap=1) (<https://uncw.instructure.com/courses/16302/files/524241/download?wrap=1>)  (<https://uncw.instructure.com/courses/16302/files/524241/download?wrap=1>) and [DAO.java](https://uncw.instructure.com/courses/16302/files/524242/download?wrap=1) (<https://uncw.instructure.com/courses/16302/files/524242/download?wrap=1>)  (<https://uncw.instructure.com/courses/16302/files/524242/download?wrap=1>). Glance through the Java files first and read their comments. Use a simple code-editor with line numbers, like Notepad++ which is installed on all the lab machines.
3. Use the [Secure Code Review Checklist](https://arch.simplicable.com/arch/new/secure-code-review-checklist) (<https://arch.simplicable.com/arch/new/secure-code-review-checklist>) to help focus your discussion. Not everything will be relevant.
4. Now, as a group, review the code and have the scribe document the findings in the [Code Review document](https://uncw.instructure.com/courses/16302/files/524240/download?wrap=1) (<https://uncw.instructure.com/courses/16302/files/524240/download?wrap=1>)  (<https://uncw.instructure.com/courses/16302/files/524240/download?wrap=1>).
 - Line number(s) - so we can find it later on
 - Severity - based on your own judgement of how badly the system could be compromised
 - Description - a quick description of the problem
5. The class will have a discussion based on the findings of each team and will share the vulnerabilities found, and the possible implications of the vulnerabilities.

This is about finding problems, not fixing them. We might discuss fixing, but code inspection meetings are about getting an overall assessment for a developer to fix offline.

If you are not finding very many mistakes in their code, then focus on pointing out the risks of future mistakes in their code.

You are not expected to compile, build, or test on this system. This is a purely static exercise.