CS 405 Journal: Don’t Leave Security to the End

By: Darrell Walker

The phrase “Don’t leave security to the end” means that security should be built into the software from the very beginning, not something added later as an afterthought. If you wait until the end of development to start thinking about security, it’s often too late. By then, major parts of the code might already be vulnerable, and fixing them could take a lot of time, money, or even break other features. It’s like building a house and only thinking about locks after it's done—you might realize the doors or windows don’t support proper security, and now you're stuck with a risky setup.

To prevent threats, there are several steps developers can take early on. First, I can follow secure coding standards, like validating user input to avoid injections or buffer overflows. Second, I can regularly use static code analysis tools like Cppcheck to catch potential vulnerabilities before the code even runs. Third, I can plan and build with principles like least privilege, making sure each part of the software only has access to what it really needs. Finally, keeping software and libraries updated is important so known security flaws don’t remain open.

For my Project Two presentation, I plan to include an example that shows I’m thinking about security throughout development. One way I’ll do this is by using unit testing with security-focused test cases. For example, I’ll write a test to make sure user input is properly filtered and doesn’t allow special characters that could cause SQL injection. This kind of test will run every time I update the code, helping catch problems early before they become real threats.

In the end, building software with security in mind from the start is not just a good habit, it’s necessary. Waiting until the last minute can be risky, expensive, and dangerous. Taking small, regular steps throughout the development process keeps both the app and the users safer.