Brian Engel

Module 6 Journal

"Don't leave security to the end" emphasizes the importance of integrating security measures throughout the software development lifecycle rather than addressing them as an afterthought. By integrating security in every aspect of building software the chances that you miss a security concern or leave a vulnerability open are reduced drastically.

There are several steps that can be taken to ensure that software is secure. Starting with the planning phase, you can pick out certain areas where special attention should be given because of sensitive data or chances of error, and plan for injection attacks by realizing any commands or queries that could be harmful or out of character for the design of the system. Another thing is to conduct code reviews to examine code and point out logic errors and unintended side effects. Static analysis of the code itself can point out security vulnerabilities as well. Making sure libraries and dependencies are up to date also reduces security concerns. Unit testing is also important to ensure correct functionality. Input validation is essential for a user making a simple mistake or a bad actor actively trying to find a vulnerability. User authentication and authorization are also important for the principle of least privilege. If you were to try and add all this at the end, you would undoubtedly miss something.

In order to demonstrate in Project Two how to ensure security is addressed intrinsically, I plan on using input validation along with unit testing. Input testing should be done throughout the project, both on the client side and the server side. The client side should check for type and check size of strings and ensure values of other types fit in the bounds of the type. On the server side the same things need to be rechecked, in addition to expected values, and if it is a string input know how it is going to be used and check for XSS and SQL injection to sanitize the input. Once input validation and sanitation are done, we can incorporate unit testing to test the validation and sanitation itself, and see how it affects other unit testing in other functions and methods.

Murray, A. (2020, June 13). Secure coding: A practical guide. Mend.io. <https://www.mend.io/blog/secure-coding/>