COSC 4765
Computer Security

Name: Jacob Tuttle

Homework 08

November 15, 2020

Embedding Security in Waterfall Design

The waterfall design pattern is perhaps one where embedding security into the design process is most obvious. With some iterative process, I think it's easy to say, "oh, we'll design security into the system in our next sprint." However, with waterfall, it's clear where designing security will take place: at the beginning! Stepping through the five phases of the process, we can see that not only will it be relatively seamless to integrate secure designs, but also that it will be done in a meaningful way that doesn't modify the core structure of waterfall. Thus, if your job is one in which the waterfall method is an acceptable development style, putting in security to that system will also be acceptable and efficient.

The first step in waterfall design is to define the requirements of the system. At this level, we can simply define the requirements for security. User data shouldn't be accessible to anyone but them, no sensitive information can be leaked out through other avenues, and the system should be resilient to basic attacks to name a few possible requirments. Stating what will be expected out of the security of the system now ensures that when we continue to the next step and design the system, we will design it to fit these security requirements.

After defining requirements, our next step is to design the system and its architecture. Since security requirements have already been well defined in our previous step, we know that security must be designed into the system. Working from the beginning with security in mind is the easiest way to ensure that its implementation will be cost effective since we won't be trying to retrofit an insecure system with security on top of it all of the sudden when security becomes a concern.

From this point on, code that is responsible for the security of the design will follow the same lifecycle as the "non-security" related code. Testing may be expanded to ensure that security features are actually secure, but this is no different than how testing would be expanded to encompass a new feature. Likewise, the maintenance of the software would have to extend to encompass the security code in the same way it would be added to for a new feature.

One of the biggest advantages of pursuing security in this way is that the business can self-define what level of security they need. The security requirements of a nuclear guidance system would be far different than the security requirements of a todo list app, but the other requirements of these systems would likewise diverge. By modifying the waterfall method to only change by asking for us to define security requirments before the system is designed, we leverage all the power of the waterfall method to ensure we hit the requirements of the project, including the security ones.