Southern New Hampshire University

CS-405 Secure Coding

Journal 2-1: Defense in Depth (DiD)

George Harrison Jr

There are many elements of defense in depth that can be added to secure important data from malicious users, hackers, and information bandits. How deep is too deep? That depends on how secure you want to be and how important the information is. For example, a user at a coffee shop would be using a wireless connection with many other users. For minimal protections on their laptop the would want a firewall as well as a good anti-virus software layer to protect from known virus and malware.

A larger company would want much more protection in place. In addition to a robust firewall which would prevent access to and from unauthorized networks and allow or block traffic based on a set of rules. There will also be antivirus software protecting against viruses and malware. These enterprises with also have data analytic integrity keeping track of known lists of viruses and other malicious code while behavioral analysis monitors breached in progress. This picks up the slack when the firewall doesn’t catch the intrusion.

Some considerations should be taken with the time, finances and operations of a project due to their connection to the reputation we want to hold a programmer. It’s best to take time to ensure that code is written properly by using techniques like error and exception catching. With no way to plan for all possible input we can plan for most possible exceptions. This will help to save money in the long term with a bug free functional program.

In the reading AES lock block were discussed. It was unique to me the percentage the distance techniques could achieve. The usefulness of the Hamming and Levenshtein distances seemed to be the most effective at 80% resilience to reverse engineering, brute force and other attacks.