**8-2 Journal: Portfolio Reflection**

Nathaniel Madore

There are some fundamental basics when attempting to establish a secure coding policy within an organization. First a secure coding standard must be created in order to establish a baseline in which programmers will write code that is considered secure by the organization. Without it there would be code that could be vulnerable and no consistent way to reference a good model of secure coding. A lack of adoption of a secure coding standard will lead to more vulnerabilities and exploits written into the program. It is best to implement a secure coding standard early in the development process since leaving security to the end is another fatal flaw that must be avoided. Leaving security to the end is more expensive of a mistake to fix then simply implementing security from the start of development.

Careful evaluation of risk must be throughout the development. This means to test often, early and in as many ways possible. By careful evaluation an assessment on risk can be made early in the development of any application. This can be used to focus development efforts on fixing these issues to further reduce risk. Since there will be some level of risk present even with the best efforts. A zero-trust environment can be used to make sure that attackers cannot abuse the system or move around the system as they like. This provides a robust way to reduce the risk of vulnerabilities and exploitation while making sure that even if an exploitation were to be successful only a limited portion of the system would be affected. By implementing best practices early on in the development process. Developers can reduce the number of vulnerabilities and exploits that are released into the production version of an application. By continually reviewing policies to make sure the organization is implementing the most recent standards. The process of developing an application can continually be refined to provide a better secured product.