CS 405 Project Two Script

Andre Burton

18 April 2024

Project Two: Security Policy Presentation

Video Part 1: https://youtu.be/R9PBz4EQly8

Video Part 2: https://youtu.be/tEaHyJF4D2Q

| **Slide Number** | **Narrative** |
| --- | --- |
| **1** | “Hello everyone, my name is Andre and I’ll be presenting about green pace security policy.”  “Ill being going into detail about topics such as Defense in depth, coding standards, encryption policies, triple-A policies, unit-testing, importance of automation, usable tools, risks of waiting vs acting, and finally giving recommendations and a conclusion. “ |
| **2** | “Knowledge is the first part of securing software and one of those crucial bits of information we all should know is the importance of defense in depth.”  “Defense in depth is the practice of proving many layers of security at different parts and step for the software.”  “Its important to know that one level of security may be perfect for to protect one part of the software but lacks protecting other parts which is why this multi layered security is needed.”  “Lastly, its important to understand the more layers you add the more power will be needed to keep up with all of the resources so it’s important to have a good middle ground of security and efficiency” |
| **3** | “The 10 coding standards play an important role in helping programmers understand areas why they should pay attention to when building software with DevSecOps in mind.”  “Under each coding standard there are example coding risks that apply to those coding standards and proper use of the standards will reduce vulnerabilities.” |
| **4** | “If we take these coding vulnerabilities, we can put them on a threat matric chart to get a quick and easy understanding of how severe the different risks are.”  “This means for a company who wants to make security changes they can worry about the one that will occur mor likely and have a more severe reaction and then work there way down going from red needing an immediate response, all the way own to green where it should be scheduled at some point.” |
| **5** | “Again, we can see in a list form of the most important to fix and should be done right away.”  “There are some vulnerabilities that have the same number which just means they are equally dangerous as the others and the company should look at which parts are affected by those vulnerabilities and motives or attackers to choose which one of the matching numbers should be fixed first, but all should be fixed as soon as possible due to the severity of the issue. |
| **6** | “When looking at other information of developers to know encryption policies and encryption use in different steps is very important.”  “Encryption in flight is just information that is currently being sent or received and such as a message being sent to another person over the internet.”  “Encryption at rest is physical devices such as usb sticks which means as long as they are disconnected from a device, they are essentially safe as long as someone doesn’t take it unknown to the owner but need their own security when connected to a device.”  “Encryption in use is the most vulnerable one of the three and thus must have more protection that the other ones and affects information that is currently being used.” |
| **7** | “Triple-A policy is more information that software developers must understand and implement in there day to day coding.”  “Authentication covers tasks of identifying users through different forms of identification and is the first step.”  “Authorization on the other hand gives the authenticated users the access they need; this information just like the authentication process is stored in the database; and access is based on default deny and given out by as admin account most of the time.”  “Accounting is the last part, and this is why account information is tracked and audits are performed, this can give us a lot of details about different users and show potential for threats and issues.” |
| **8** | “Unit testing is a crucial part of the security of the software we create and can make testing quicker and more efficient; in the next four slides are examples of positive and negative tests and also failed tests that help give examples of how developers should be implementing these into their own work.”  “In this example the test was to check if the collection stays at zero if the number tries to go more negative than zero, we can see by the testing results that the test failed and gives a good example to developers of what happens when there is an issue with the test.” |
| **9** | “The next test checks to see if a collection is empty when it initially created; We can see that the test ran and finished successfully.”  “This is an important test for a lot of applications because if a collection isn’t empty, it can cause information spillage or errors”. |
| **10** | “This next test checks for max size that is equal or greater and test the given parameters.”  “Again, this is useful in a lot of different aspects and should be given a consideration when developing unit tests.” |
| **11** | “This last test example checks that an exception is thrown if calling for out of bound information.”  “This test is an example of a negative test which checks that the system throws an exception when the parameter is met which is important to check that exception is thrown instead of having vulnerabilities in the code.” |
| **12** | “Automation is an important part of testing code which should make testing quicker and more efficient but relies heavily on proper usage of testing.”  “From the chart we can see in different areas where automation can be implemented.” |
| **13** | “DevSecOps is the center ground of software security, and all products should be implementing it.”  “A way to make security easier is using tools, I have shown an example tool which has run some code and then outputs some issues with the code; there are many types of tools available to use and should be picked based on experience and product usage and which suits the software the best.” |
| **14** | “There will always be risks and benefits to acting now vs waiting and its important to understand these in order for a company to implement the needed security fixes.”  “For the most part acting now will have initial additional costs but will greatly reduce the chance of vulnerabilities: waiting prevents initial costs but with the high risk of vulnerabilities and high costs to fix stuff after attacks.”  “There are some examples that show what happens when companies fail to implement software security and how much of an impact it had on their finances because they decided to wait.” |
| **15/16** | “Based on all of the given information I would recommend immediate implementation of the coding standards and applicable information to fix and prevent all software vulnerabilities.”  “Although here is the higher initial cost it can greatly prevent long term catastrophic consequences.”  “In conclusion developers should first be educated on the new security practices, shown examples, and then guided on how to implement it in current and future work; This will save the company lots of time and money in the long run, and overall great happy user which is the overall end goal.” |