ISEC 400 Homework 3 Name: Megan Leonard

Answer the following questions based on your reading of the textbooks, any supplemental material, and the instructor’s presentation this week. If you use an external source (i.e. a web-page, the required textbook, or an additional book) to help you answer the questions then be sure to cite that source. Hint: you should probably always be citing a source.

## Questions

1. **[5 points]** Consider a small business employer who writes web application software for other small businesses. Their approach to security is to perform penetration testing before product release using tools such as Vega, OWASP ZAP, Burp Suite, and w3af. When the tools find no more vulnerabilities, the owner ships the software and claim a high level of security for his products. Do you agree? Why or why not? What are the benefits and drawbacks to his approach? If you were running the business, what (if any) additional steps would you take and why?

I do not agree with the employer as while there may have not been anything with the penetration testing, that does not mean you can claim a high level of security as there are other factors that are needed to make that claim. The benefit of his approach is that you do not have to spend time at each section running the same tests. It speeds up the time it takes to reach the end product. The downsides are that your tests could miss security flaws that could have been shown with prior testing, and it can take more time should they find vulnerabilities. Additional tests I would take is testing on different platforms, since this is a web application it is best to test it out on Windows and Mac and if it connects to the web, then try different browsers. I would make sure that the testing is done to test each system it could run on as it may not have vulnerabilities shown with Windows 10 but could with Windows 11.

1. **[5 points]** Consider the standard software development lifecycle contrasted against a secure software development lifecycle of your choice (e.g. SDL, BSIMM, OpenSAMM). What are the similarities in the lifecycles? What additional activities take place in a secure development lifecycle? Which, in your opinion, are the most critical additional activities and why?

The standard software development lifecycle and the secure software development lifecycle are quite similar in the way they follow the main phases of plan, analysis, build, test, and fix. The secure lifecycle adds in security into the different phases with the addition of bug fixes, dynamic analysis, threat modeling, and incident response planning. In my opinion, the most critical additional activities are the incident response planning, threat modeling, and risk assessment. To be able to counter vulnerabilities, it is best if you know what they are and once you find them how to deal with them and anything that happens because of them. Everyone had a different opinion and for me, I think the best thing to do is to have plans for how to deal with problems that arise from threats and have an idea of how to respond to them.

1. **[5 points]** What is STRIDE and DREAD? How are the two related to one another? How are they used in a secure development context?

STRIDE is a threat model that stands for spoofing, tampering, repudiation, info disclosure, denial of service, and elevation of privilege. DREAD is damage potential, reproducibility, exploitability, affected users, and discoverability which is a threat scale. STRIDE lists several different threats while DREAD will take its threats and apply the scale to determine the threat level. They are used in secure development to help determine the areas that need the most protection and the level of importance when it comes to securing the different areas.

1. **[10 points]** Consider the scenario of user logging in to a web application for e-commerce. Using the Microsoft SDL Threat Modeling Tool (downloadable from <https://www.microsoft.com/en-us/download/details.aspx?id=49168>), do the following:
   1. Create a context diagram for the application
   2. Create a level 1 diagram for the login process
   3. Apply STRIDE to the data flows between the user and the login process *only* (or else you’ll spend too much time).
   4. Show your results from a-c above and explain what you learned through the process.

Graphical user interface, application

Description automatically generated

c. Looking at the threat model, we have a chance of spoofing, denial of service, and tampering based off the code of the login page.

d. After creating the model, I learned that there are a lot of parts to think about. I did not go very detailed with my graph, but I can see how it can get complicated as a person works through the model and applying STRIDE and DREAD to each part and each connection.

1. **[5 points]** In approximately 300 to 400 of prose (i.e. sentences, not bullet lists) using APA style citations if needed, summarize and interact with the content that was covered this week in class. In your summary, you should highlight the major topics, theories, practices, and knowledge that were covered. Your summary should also interact with the material through personal observations, reflections, and applications to the field of study. In particular, highlight what surprised, enlightened, or otherwise engaged you. Make sure to include at least one thing that you’re still confused about. In other words, you should think and write critically not just about what was presented but also what you have learned through the session. Ask at least one question that your instructor can answer in the returned assignment or class discussion.

This week we learned more about threats and secure development lifecycles. The lifecycles are always interesting to learn about as they tend to have the same phases but differ with the importance of different tasks like how the secure lifecycle has a greater focus on security. The threats models were interesting to work with and I found it cool how we can use STRIDE to identify known threat types, and DREAD to determine the severity of them. I do find it funny to think about how many different abbreviations like DREAD and STRIDE there are within security as you stride through the threat types and dread the threat severity. My question this week is what is your favorite abbreviations within the field?

Citation:

*5 reasons why penetration testing isn't enough*. SecurityScorecard. (2023, February 2). Retrieved February 12, 2023, from <https://securityscorecard.com/blog/why-penetration-testing-isnt-enough/#:~:text=If%20you%20pen%20test%20medical,that%20afflict%20the%20old%20ones>.

*Bolt-on vs baked-in cybersecurity*. Lawfare. (2022, June 23). Retrieved February 12, 2023, from <https://www.lawfareblog.com/bolt-vs-baked-cybersecurity>

Maayan, G. D. (2019, July 13). *How you should approach the secure development lifecycle*. DATAVERSITY. Retrieved February 12, 2023, from <https://www.dataversity.net/how-you-should-approach-the-secure-development-lifecycle/#:~:text=Secure%20Development%20Lifecycle%20(SDL)%20is,application%20from%20inception%20until%20decommission>.

TutorialsPoint. (2021, July 28). *SDLC - Overview*. TutorialsPoint. Retrieved February 12, 2023, from <https://www.tutorialspoint.com/sdlc/sdlc_overview.htm>

Whittaker, T., &amp; Whitney, T. (2022). Week03 Power Point. Retrieved February 12, 2023, from <https://franklin.instructure.com/courses/11070/files/1516716?module_item_id=623550>