8-2 Journal: Portfolio Refection

Joel Meza

Professor Trevor Hodde

February 24, 2022

CS 405 – Secure Coding

Southern New Hampshire University

When we consider using adopting a security code standard, we enable a project to be persistent in security within the software DevOps lifecycle. At will no matter which developer is updating its portion of the code to be commit and pushed into a main branch, it’ll still have a great secure assurance approach at each step given. As we had learned in the latest journal called, “Do Not Leave Security Towards the End”, which specifically mentions even when planning a software project, we must require and implement security throughout each progression phase. Not having security involved since the beginning of a planned project, it’ll only result in compilation errors or simply to become a vulnerable project which can cause a major exploit at any time. Running Unit Testing is always encouraging to be built even before building new code for a software project where it will allow to fill in any vulnerability that can be projected and enforce by its security in depth measures as it goes on through its lifecycle.

Evaluating any threat within a software project is very important to always keep in mind. As threats are reviewed within categories and set to a level to define what kind of threat are going to be analyzed by software developers and what will be required to enforce our security policy and best practices on. As I learned that there are different kind of threats which entitles in a high risk to be service denial, where a hacker being involved as the middleman between a user and its access backend server. Other low prioritized level of threat are SQL injections and Phishing which both are known from a hacker to rely on a vulnerability from a system and try to obtain important data from these vulnerabilites by different portions. Lastly, the most unlikely threat is malware that is set from a hacker by influencing the user to believe first, for the user to fall into the trap to reveal important data. For software developers, developing security traits within the project can help manage time and keep a low-cost budget before it’s out of control if there was no security model and policy involved.

Including zero trust model is very important as it eliminates strong to weak hacker approaches to a software system or program. The principles of zero trust are simple which left into this saying, “Never Trust, Always Verify”. In practice, that means each user must be verified before access is granted to any resource. (Bliss, 2021) Every request from every user, inside or outside of your perimeter, must be authenticated, authorized, accounted, and encrypted in real time while it's adopting the Triple A's best security practices. This protects your organization in ways other models can’t. It stops malware from entering your network, gives remote workers more protection without affecting productivity, simplifies management of security operations centers with enhanced automation like machine learning integration and artificial intelligence, and extends visibility into potential threats to improve proactive remediation and responses.

In conclusion, consideration in using security policies as said throughout this journal is very important to have in a DevOps software project lifecycle. Implementing security to be forced to adapt in time to time is great because of the war between security models and hackers that go on after each other from time to time where both have gone through their good and bad experiences which makes both to strive and get better as they’ve learned from these battles. Defense in Depth (DiD) security policy is highly recommended because it works within layers or levels given from security measures to keep safe a confidential domain that only authorized officials are granted to their specific level of authorized access. While this is enforced, we can continue to strive in better safety for a software program or system to be by keeping the Triple A’s (authenticate, authorize, and accountability) to each phase of the security depth whenever a new or existing user will continuously be required to go through each security policy step which will allow help for our program to eliminate unwanted behaviors or intruders before they possess any confidential data.

**Citations**

1. Bliss, Phillip. (2021, April 28). *EdgeTech Podcast: Founder’s Podcast with Qnext Corp. CEO Anthony DeCristofaro.* **Retrieved From:** [**https://edgecomputingassociation.com/edgetech-podcast/2021-04-28/founders-podcast-qnext-corp-ceo-anthony-decristofaro**](https://edgecomputingassociation.com/edgetech-podcast/2021-04-28/founders-podcast-qnext-corp-ceo-anthony-decristofaro)