3-2 Journal: Reflection

SNHU

CS-305

*What is your role in solving security concerns as a developer? What might solve security concerns as a developer involve?*

My role as a developer in solving security concerns is to constantly be aware of possible security risks. This includes all projects that I am currently working on as well as projects I have worked on in the past. Security should also be on the forefront of my mind as I am developing a product or application. I should not push security aside to just focus on functionality of the project. Security should be incorporated into all elements of the development process and as the developer this responsibility falls on me. Solving security concerns as a developer involves having a security-based mindset in all facets of the development process from planning to operation. The includes security design and engineering, security testing, security monitoring, and security risk management (Jeganathan, 2019). Following these steps ensures that security is planned and implemented within every step of the development life cycle allowing for more secure and complete products and applications to be delivered to the consumer.

*Where does security fall within the software stack and development life cycle?*

When using the DevSecOps method of software development security falls within all stages of the Software Development Life Cycle (SDLC). Security is at the forefront of all the stages of DevSecOps in the SDLC. This is where DevSecOps differs from other types of SDLC’s. This method ensures that security is planned and orchestrated throughout the entire development process instead of just the testing phase. This saves time, energy, and resources and if done correctly helps produce a more secure final product for the customer.

*How might you add security measures to transform a DevOps pipeline into a DevSecOps pipeline?*

There are many methods to transform DevOps to DevSecOps. The first step is in the planning phase. Considerations are taken to add security into the entire planning phase of DevOps. This includes high level risk assessment, threat models, securing of DevOps lifecycle tool (GitLab, Azure DevOps, Etc.), user access keys, and segregation of duties (Jeganathan, 2019). The second step that needs to be addressed is the coding phase. This is done by applying secure coding practices and integrating SAST tools such as OWASP, Fortify, Veracode, etc. (Jeganathan, 2019). Teams will need to decide on the SAST tool that works best for their specific project. This is not a one size fits all solution. This stage also includes peer code review, development of test scripts, and elimination of vulnerable components. The third stage is the build stage. During this stage “software modules are checked into the source code repository and made available to package and bundle for deployments into next environments such as QA, user acceptance testing, pre-production, and production” (Jeganathan, 2019). Segregation of duty controls are established as well as automated tools for packaging software models. The fourth stage is known as the test stage. A major difference between DevOps and DevSecOps in this stage. “Even though unit-level security testing is done by developers, extensive system and

integration testing occurs at this stage to prevent various security flaws in the software modules. Security teams at this stage are referred to as blue team and red team, where red team focuses on offensive testing and blue team focuses on preventing attacks from the red team” (Jeganathan, 2019). The fifth stage is the release and deploy stage. During this stage the software modules are released to lower and live environments incrementally while being supported by a variety of tools such as Puppet, Chief, Etc. (Jeganathan, 2019). The final stage is the Operate stage. This stage is run mostly by the operations team and not the development team. The security team is still heavily involved in this stage. This is when logging, intrusion detection, and more is done. If a problem occurs, it should be fixed immediately during this stage. If all the stages mentioned above are implemented with the stated security concepts included DevOps can successfully and fluidly transform into DevSecOps.

*The article suggests creating and following a plan to secure the entire DevOps life cycle. What is included in the suggested plan, and would you recommend following it?*

I outlined the plan in the above section on how to secure the DevOps lifecycle and transform it into the DevSecOps lifecycle. The steps include planning, building, testing, release and deploying, and operating. A more detailed description of each step is available in the above section. I would recommend following this plan because security is implemented in each step preventing possible security lapses. This also will provide a better, more secure product in the same amount of time as the standard DevOps life cycle without having to go back and address security issues at the end of development.

**Citations**

Jeganathan, S. (2019, November). *DevSecOps A Systemic Approach for Secure Software Development*. Information Systems Security Assosiation. Retrieved January 20, 2023, from https://mydigitalpublication.com/publication/?i=632044&article\_id=3524379&view =articleBrowser