Julianne Takaya

3/22/2025

CS-305

3-2 Journal: Reflection

As a developer, my role in solving security concerns is to develop programs securely. By building quality code and staying up to date on the latest threats to online security I will protect my product and client from attacks. Solving security concerns might involve ensuring all incoming data is verified before being put to use, or it could involve using more secure dependencies or open-source software as needed.

Security is a concern at every step in the software stack and development lifecycle. From planning to deployment and maintenance to proactively identify and mitigate vulnerabilities. Integrating security concerns from the beginning saves time and money, as waiting for testing to identify security leaks could lead to code that needs to be entirely re-written or scrapped. It also results in a more secure code, and more trust in the product.

To transform a DevOps pipeline into a DevSecOps pipeline, I would start with the planning phase. By ensuring that security concerns are identified early on, a plan is set in place to mitigate them during the planning phase. That way, security is developed and tested alongside code, ensuring a secure program.

The suggested plan involves starting with high-level, rapid risk assessment for the new release. Then plan and secure the DevOps lifecycle tool, so only those authorized to work on the program has access. Next, ensure user access keys, privileged service accounts, API keys, etc. are protected properly with privileged account security tools. Define infrastructure protection controls and enforce segregation of duties (Jeganathan, 2019). I would recommend following this plan. It seems like a good way to get everyone into the mindset of securing the code by securing the coding environment.

References

Jeganathan, S. (2019). DevSecOps: A Systemic Approach for Secure Software Development. *ISSA Journal*, *17*(11), 20–27.