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

Solving security concerns as a developer is crucial to the integrity and confidentiality of the software I am working on. Solving security concerns as a developer involves: writing secure code; when planning the software, considering security and building the software around those concerns; testing for known security vulnerabilities; and ensuring dependencies are updated.

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

Simply, all of it. Security should be implemented at every level from design to release.

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

I think that the easiest way of adding security measures is to start embedding security practices into the normal DevOps pipeline. Doing this at the beginning of a project is probably the best way to ensure it is engrained into the project team.

# 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 would recommend following it. Incorporating security measures from the early stages of development helps reduce the risk of vulnerabilities.

The suggested plan from the article follows:

* “Start with a high-level rapid risk assessment for the new release and quantify the risks by evaluating the threat models.
* Plan and secure the DevOps lifecycle tool, typically web-based tools such as GitLab, Azure DevOps, etc.
* For example, secure access points based on role- or attributes-based access control models. • Protect user logon by integrating with company federation (identity provider) and web-access management tools if exist, otherwise with a compensating control to meet the requirements.
* Apply 2FA/MFA based on the criticality of the environment and systems.
* Ensure user access keys, privileged service accounts, API keys, etc. are protected properly with privileged account security tools if exist, otherwise with a compensating control to meet the requirements.
* Define infrastructure protection controls and enforce segregation of duties. For example, developers don’t need access to the live environment, only the operations team.” (Jeganathan, 2019)

# References

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