Bryan Cabrera
12/21/2024
Module 12.2 Assignment

Compliance Case Study Summary

In different business environments, especially those that have more regulation such as finance, healthcare, or aerospace, it is extremely important to ensure that the software development team along with the rest of the organization follow any compliance requirements set forth by local, federal, and world governments. Compliance requirements while potentially seen as limiting are created for a reason. They are made in order to ensure integrity and confidentiality of sensitive data and/or systems. This paper will explore some real world case studies in which the main points of integrating compliance into the software development lifecycle will be discussed.

Providing Compliance in Regulated Environments:

This case study mentions how from the point of view of Bill Shinn who is a Principal Security Solutions Architect at AWS, one issue he's faced is that of having to show customers(organizations) that they can still comply with compliance laws and regulations while they move from a more traditional systems environment over to a public cloud environment. The case study mentions how auditors of the more traditional systems still follow traditional methods such as relying on screenshots and CSV files demonstrating this as an inefficient way to do things. Bill Shinn mentions the solution to this issue being to integrate any audit requirement into the control design process itself utilizing tools in order to access audit evidence. By allowing teams to work with the auditors of compliance requirements, both the companies and auditors can streamline their process and reduce any audit related errors.

Furthermore, the case study goes on to give examples of tools used for addressing compliance and audit challenges. One of those tools being AWS CloudWatch in which it's mentioned that it can be used to monitor systems with controls typically only taking a single command. Ultimately this case study shows how a development team can effectively close the gap between DevOps practices and auditor requirements essentially showing how DevOps while a newer practice can absolutely comply with any laws and regulations.

Relying on Production Telemetry for ATM Systems:

In the next case study Mary Smith(fake name) who leads a DevOps initiative for a large financial services organization mentions how she has noticed that auditors along with regulators will normally place too much reliance on code reviews when it comes to searching for potential fraud. Mary mentions that instead of only doing this, they should also count on production monitoring procedures in addition to any automated testing. An example given in the case study was about a developer purposely placing a backdoor in the code that ended up being deployed in some ATM cash machines. The backdoor had the effect of putting the ATMs into a "maintenance mode" at specific times which allowed for the developer to take cash out of the machines. The overall point of the example being that they were able to detect the fraud fairly quickly and not even through a code review. When thinking about it this makes sense because if a developer is skilled, and has the means and a motive, it's going to be very difficult or next to

impossible to locate such fraud through just a code review as the developer can likely hide this from a code review if they know what they're doing. The team was specifically able to discover the fraud from analysing data, in this case they noticed that some ATMs in a given city were being placed into maintenance mode at irregular times. Ultimately this case study shows how over reliance on code reviews and having a separation between the Dev and Ops teams can leave an organization open to vulnerabilities. Also how the practice of telemetry can help teams in detecting issues that they otherwise could very well miss, leading to potentially devastating exploits and attacks.