Assignment 8.1 Case Study: Static Security Testing at Twitter (2009)

*Becca Buechle*

*October 31, 2020*

For this week’s discussion board assignment, read the case study presented on page 320 of the course textbook and compose a brief summary of the main points the author made and the lessons learned. A structured post has an introduction, a conclusion, and a developed body that flows well – generally at least a few paragraphs.

**The initial post must be in essay format (introduction, body, and conclusion) and a minimum of 250 words. Points will be deducted for not meeting the specified word count requirements.**

Twitter like many other tech companies such as Google and Facebook when through a hype-growth stage when they were new. In January 2009 to March of that same year that user base jumped from 2.5 million to 10 million. (Kim, G.) This huge jump caused many issues for Twitter. One of the bigger issues was the security for the application. Below I’ll discuss how Twitter was able to combat this issue.

For years Twitter had issues handling the rapid growth that they were getting for their applications this causes several issues for the company. Users would reach home page only to find the Fail Whale error page because user demand was just too great for them to handle. During this same time frame in 2009 Twitter was also having issues with security and had two serious breaches. (Kim, G.) The first one happened in January when Barack Obamas account was hacked and the second was in April when the administrative accounts where compromised through a brute-force dictionary attack. (Kim, G.) The FTC (Federal Trade Commission) to judge Twitter saying that it was misleading its users into believe their accounts where secure when they weren’t and issued an FTC consent order. (Kim, G.) Twitter had to work quickly to get these issues sorted out per the FTC consent order. They were required to designate employees what would be responsible for information security plan. (Kim, G.) Along with identifying any risks both internally and externally that could lead to anymore intrusion incidents and to create a plan on how to solve these issues. Lastly, they had to come up with a plan to maintain privacy of any user information both from inside and outside intrusions with some sort of verification and testing of the implemented security. The teams of engineers that got assigned to this had to not only figure out how to close the holes that were already there but also had to integrate security into the daily work. Collins, Smolen, and Matatall came up with a list of things the Twitter team need to address: The needed to find a way to prevent the same mistakes from being reported over and over because each time they were fixing the same issue multiple times. So, they needed to modify the way the system automation tools to keep this from happening. (Kim, G.) They needed to find a way to find the errors early on and be able to relay those to the developers, so they knew exactly what to fix but they also needed to know when there was a false positive. (Kim, G.) Twitter had a huge breakthrough for its Infosec team when it hosted a hack week the goal was to out in security scanning into the earliest stages of the development process. (Kim, G.) The results where incredible.

To sum it up when you first start building your application security should be one of the main things on your mind as you are developing not just after you finish. By keeping external and internal users safe is so important. By writing these security features it really helps the developers write more secure code. I believe this is something that should be talked about at the start of each project so that its in everyone’s head. With any type of breach your risking your company and users trust.

Bibliography

Kim, G., Humble, J., Debois, P., & Willis, J. (2016). DevOps Handbook. Portland, OR: IT Revolution Press.