Kelsey Wyer

Course Reflection

CS – 405 Secure Coding

Southern New Hampshire University

It has now been proven hundreds of times that failing to handle a security vulnerability before it is exploited can have more than just significant monetary consequences. There is a serious demand for a more proactive approach. Development teams that take security into account from the beginning spend far less time mitigating vulnerabilities later (Google, 2021). This results in higher quality releases that are often pushed out before their deadline. Waiting until a later testing stage to check for and handle vulnerabilities can make rectifying them take longer, increasing expenses and even still, potential threats could go totally unnoticed until they become a much bigger problem (Google, 2021). Security must be considered during all stages of development.

It’s easy to become overwhelmed when looking at the collection of security rules and recommendations in the CERT Coding Standards published by Carnegie Mellon University’s Software Engineering Institute. Of course, not every standard discussed in this collection will be applicable to every company or project, but there is another important factor to take into account. For every threat listed in the CERT Coding Standards there is also an assessment of threat severity, the likeliness the threat will be exploited, their priority level, and the cost of remediation. You will see that many low severity and low priority threats are some of the most likely to occur while others that will rarely happen, will certainly be disastrous if they do.

The concept of zero-trust security fills many of the gaps in security that current policies allow for. Until recently, it wasn’t too big of an issue to assume that all the users and devices that had successfully connected to the corporate network were authorized to be there and to access data and resources. This is how so many companies protected their systems and data for so long, but it isn’t enough anymore. With more users connecting remotely and on multiple devices, there are a lot more points of attack (Kueh, 2020). Now, it is necessary to authorize not just the users and devices on your network, but also their connection type, external applications they may be using for access, and data they want to transmit over the network (Kueh, 2020). Taking these different areas into account to implement zero-trust makes for a much more well-rounded security policy that does not leave much room for exploitation.

A security policy standardizes and details the practices that each employee is required to adopt. It indicates leadership’s attitude toward risk and security and offers legal and ethical support (Dunham, 2021). A security policy promotes accountability and acts as a tool to enforce it. While the needs of each company will determine the policy that is most appropriate for them to adopt, there are some things that should be considered by all who wish to implement a security policy. It is important that the standards you choose to include can be enforced, any exceptions to a standard need to be thoroughly explained, and don’t make the policy to long or complex (Dunham, 2021).

References

Carnegie Mellon University. (2020). SEI CERT Coding Standards - CERT Secure Coding - Confluence. SEI Cert Coding Standards. Retrieved December 19, 2021, from https://wiki.sei.cmu.edu/confluence/display/seccode/SEI+CERT+Coding+Standards

Dunham, R. D. (2021, October 26). The Importance of A Company Information Security Policy. Linford & Company LLP. Retrieved December 19, 2021, from https://linfordco.com/blog/information-security-policies/

Google. (2021). DevOps tech: Shifting left on security |. Google Cloud. Retrieved December 19, 2021, from https://cloud.google.com/architecture/devops/devops-tech-shifting-left-on-security

Kueh, T. K. (2020, January 17). A Practical Guide to Zero-Trust Security. Threatpost. Retrieved December 19, 2021, from https://threatpost.com/practical-guide-zero-trust-security/151912/