Security Design Principles

Tyler O’Riley

CSD370

8/29/2023

THREE SETS

|  |  |  |
| --- | --- | --- |
| CSSLP Secure Software Textbook  • Good enough security  • Least privilege  • Separation of duties  • Defense in depth  • Fail safe  • Economy of mechanism  • Complete mediation  • Open design  • Least common mechanism  • Psychological acceptability  • Weakest link  • Leverage existing components  • Single point of failure | National Cyber Security Centre  • Establish the context before designing a system  • Make compromise difficult  • Make disruption difficult  • Make compromise detection easier  • Reduce the impact of compromise | Microsoft Azure  • Plan resources and how to harden them  • Automate and use least privilege  • Classify and encrypt data  • Monitor system security, plan incident response  • Identify and protect endpoints  •Protect against code-level vulnerabilities  • Model and test against potential threats |

EVALUATION

All three principal lists identify an intensive review of having multiple layers of different security facilities. Some topics were in depth while others kept a broad conceptual viewpoint if tasks at hand. Our textbook went the most in depth in terms of detailing each layer of security that would be utilized and how to evaluate all parts of the program. The National Cyber Security Centre opted for a conceptual view. Evaluate the program on a wholistic standpoint, then slowly begin to dissect the finer details of potential risk points and how to make them stronger. Finally, Microsoft Learning in regard to Azure discussed the primary building points of a program design. The way the details were discussed were more related to building a structure itself rather than maintaining an existing one.

In name, all three principal lists have varying parts that do not use the same naming convention. In concept, they are all similar in how they approach security design. Our textbook seems to go the most detailed covering topics such as psychological acceptability and economy of mechanism.

MY LIST

* Full system evaluation
* Identifying weakest assets
* Test modeling and risk assessment
* Cost of security structure
* Open design
* Flexible implementation

I chose to keep my structure simple and closure in relation to software development itself. Each principle is organized in a step-by-step process starting at the review of a program or architecture, moving through the phases of testing, evaluating, and implementing a security structure that is fluid and able to change as new methods are founded. I chose this design to better mix the ideas of the three lists above to help with flow, ease of access, and a structure that can be followed for development.

RESOURCES

*Cyber Security Design principles*. NCSC. (n.d.). https://www.ncsc.gov.uk/collection/cyber-security-design-principles/cyber-security-design-principles

PageWriter-MSFT. (n.d.). *Security design principles - microsoft azure well-architected framework*. Microsoft Azure Well-Architected Framework | Microsoft Learn. https://learn.microsoft.com/en-us/azure/well-architected/security/security-principles