Orion J. Murray

CS-405

**Module 8 Journal**

Security in development should be at the forefront of every decision implemented. A defense-in-depth approach ensures that all system components are compliant, resilient, and effectively prepared against threats. Breaches will always be attempted by bad actors, and organizations that proactively integrate security measures benefit from lower remediation costs and reduced reputational damage. A fundamental takeaway from this course is that security should be embedded throughout the software development lifecycle rather than being addressed as an afterthought before deployment. Secure coding standards, such as those recommended by OWASP, provides guidelines for writing resilient code that mitigates vulnerabilities like SQL injection, buffer overflows, and improper input validation. When integrating security from the initial design phase, developers can identify and address flaws early, reducing the cost and complexity of remediation.

The course emphasized that risk assessment is a critical component of cybersecurity strategy. Risk evaluation involves identifying potential vulnerabilities, assessing their likelihood of exploitation, and estimating the impact on the system. One key takeaway was the cost-benefit analysis of security controls in how delaying security implementation often results in higher costs and fines. The Zero Trust model assumes no implicit trust and requires every access request to be verified regardless of the user’s location or network. Traditional perimeter-based security model has been proven insufficient due to exploits and Zero trust has key principles that include least privilege access, continuous authentication, network segmentation, and assumes that breaches will occur at some point.

Security policies are essential for standardizing best practices and ensuring compliance across an organization. Throughout the course we explored various frameworks, such as ISO/IEC 27001 and NIST 800-53, which provide structured guidelines for developing robust security policies. Key policy recommendations often include encrypting data at rest and in transit, creating robust incident response plans, implementing RBAC and enforcing least privilege access, and education or security development training for developers and IT staff.

Security in development is an ongoing, multi-layered process that requires a proactive approach. By adopting secure coding standards, conducting rigorous risk assessments, implementing Zero Trust principles, and enforcing robust security policies, organizations can significantly reduce their exposure to cyber threats. In an era where cyberattacks are increasingly sophisticated, prioritizing security is not just an option, it is a necessity.