ONE PAGE SUMMARY

IT & Software › Cyber Security › Secure SDLC

Introduction to Secure SDLC

Secure SDLC: Building Security into Software Development

DEFINITION

> Secure SDLC is a process that integrates security practices into software development. It ensures that security is considered at every stage of the development lifecycle.

KEY CONCEPTS

Requirements Gathering

Identify security requirements alongside functional requirements to address potential vulnerabilities early.

Design Phase

Incorporate security architecture and threat modeling to anticipate and mitigate risks.

Implementation

Use secure coding practices and tools to prevent common vulnerabilities during development.

Testing

Conduct security testing, including static and dynamic analysis, to identify and fix security issues.

Deployment

Ensure secure configurations and access controls are in place before releasing the software.

Maintenance

Regularly update and patch software to address new vulnerabilities and threats.

EXAMPLES

→ Using threat modeling during the design phase to identify potential attacks.

→ Implementing input validation to prevent SQL injection attacks in the implementation phase.

→ Conducting penetration testing before software deployment to uncover security flaws.

MEMORY TIPS

★ Remember 'GRIP-M' for the stages: Gather, Review, Implement, Protect, Maintain.

★ Think of Secure SDLC as a 'security umbrella' covering all development phases.

★ Use 'DIP-T' to recall: Design, Implement, Protect, Test.

COMMON MISTAKES

✗ Neglecting security requirements during the initial planning stages.

✗ Skipping security testing due to time constraints.

✗ Failing to update software after deployment, leaving it vulnerable.

QUICK RECAP

Secure SDLC integrates security into every phase of software development. By addressing security early and continuously, developers can significantly reduce vulnerabilities and risks.