Summary on Security Testing

Module Summary

PW SKILLS

1. Static Code Analysis Tools

- **SonarQube:** A widely used tool for continuous code quality inspection, SonarQube detects bugs, security vulnerabilities, code smells, and technical debt in multiple programming languages. It integrates with CI/CD pipelines to enforce code quality standards.
- **Fortify:** A security-focused static analysis tool that scans code for vulnerabilities, ensuring applications are secure by identifying issues like buffer overflows, SQL injection, and cross-site scripting (XSS). It supports multiple languages and integrates well into DevSecOps workflows.
- **Checkmarx:** Another popular static code analysis tool that focuses on identifying vulnerabilities early in the development lifecycle. It integrates with development environments and CI/CD pipelines.

2. Dynamic Application Security Testing (DAST)

- **DAST:** A method of testing web applications and APIs by simulating real-world attacks while the application is running. Unlike static analysis, DAST doesn't require access to the source code but interacts with the app from the outside to find security issues such as SQL injection, XSS, and authentication flaws. Tools like OWASP ZAP and Burp Suite are commonly used for DAST.

3. Penetration Testing and Ethical Hacking

- **Penetration Testing (Pen Testing):** A manual or automated process where security professionals simulate attacks on a system to identify vulnerabilities that could be exploited by malicious hackers. It typically includes network, application, and physical security testing.
- **Ethical Hacking:** A broader practice that involves using hacking techniques to test and improve the security of systems and applications in a legal and authorized manner. Ethical hackers find and fix vulnerabilities before malicious actors can exploit them.

4. Security Testing in CI/CD Pipelines

- **Integrating Security Testing in CI/CD:** Security tests, including static code analysis (SAST), dynamic analysis (DAST), and vulnerability scanning, are integrated into CI/CD pipelines to detect security issues early in the development process. Automated security tools (like SonarQube or Fortify) scan code and environments at every stage of development to ensure that vulnerabilities are identified before deployment.