Nave Secure Development Lifecycle

15 Sep 2020

nave

Table of Contents

1. Concept and Planning

Define the implementation concept and evaluate its viability using the following practices:

- **Security Development Lifecycle Discovery (SDL Discovery)**
  Define security and compliance objectives. Select an SDL methodology and write a detailed plan of relevant SDL activities to address security issues as early as possible.

- **Security Requirements**
  Prepare a list of security requirements. Include both technical and regulatory requirements to identify and fix potentially non-compliant areas of the implementation.

- **Security Awareness Session**
  Hold a training session to address essential security concerns to achieve alignment between everyone involved. Define secure design principles.

  Purpose:
  Improve the success of the planning phase and ensure application compliance with security standards. Allocate the necessary team members with expertise in application security.

2. Architecture and Design

Design a solution that meets customers' requirements including modeling the application structure and its usage scenarios, as well as choosing third-party components that can speed up development using the following practices:

- **Threat Modeling**
  Identify probable attack scenarios and add relevant countermeasures to the application design to uncover possible threats early, reduce the associated costs and lay the basis for future incident response plans.

- **Secure Design**
  Validate the design document and subsequent updates in light of the security requirements to identify features exposed to security risks before implementation.

- **Third-Party Software Tracking**
  Monitor third-party software for vulnerabilities regularly and apply patches when necessary to spot areas threatened by compromised components and fill in the gaps.

  Purpose:
  Identify weaknesses before they make their way into the application. Mitigate security risks and minimize the chance of vulnerabilities originating from third-party components.

3. Implementation

Develop the application code, debug it, and produce stable builds suitable for testing using the following practices:

- **Secure Coding**
  Follow the established secure coding guidelines to enforce secure coding principles and eliminate trivial vulnerabilities.

- **Static Scanning**
  Scan the newly written code on a daily basis to find potential weaknesses to uncover mistakes before they can make their way into the application build.

- **Code Review**
  Perform code review and ensure compliance with the established security principles. Flag and fix potential issues before you move on to the next task.

  Purpose:
  Reduce the number of security issues. Combine automatic scanning and manual code reviews.

4. Testing and Bug Fixing

Discover and correct application errors running automatic and manual tests, identifying issues, and fixing them. Use the following practices:

- **Dynamic Scanning**
  Execute runtime scanning with monitoring of executed code and application data flow to discover regular vulnerabilities and pinpoint configuration errors that impact security.

- **Fuzzing**
  Generate random inputs based on the defined patterns and check whether the application can handle such inputs properly to improve protection from attacks that use malformed inputs, such as SQL injection.

  Purpose:
  Reduces the number of security issues to provide decent protection from a wide range of known threats.

5. Release and Maintenance

Go live. Use the following practices:

- **Environment Management**
  Security monitoring must cover the entire infrastructure to improve the overall security of your application.

- **Incident Response Plan**
  Review the procedures in the incident response plan regularly to make sure we are able to address any security breaches that might occur.

- **Ongoing Security Checks**
  Perform security checks regularly to identify new types of vulnerabilities at a steady rate and protect our application from newly discovered vulnerabilities.

  Purpose:
  Respond to emerging threats quickly and effectively.