Customized Course related to Java Security-2

1. Introduction to Java Security

1.1 Overview of Java Security

1.2 Importance of Security in Java Applications

1.3 Common Security Threats in Java

2. Java Security Architecture

2.1 Java Security Model

2.1.1 Class Loader

2.1.2 Bytecode Verifier

2.1.3 Security Manager

2.2 Java Authentication and Authorization Service (JAAS)

2.3 Java Cryptography Architecture (JCA)

3. Securing Java Code

3.1 Input Validation

3.1.1 Handling User Input

3.1.2 Data Validation Best Practices

3.2 Secure Coding Guidelines

3.2.1 Avoiding Common Vulnerabilities (e.g., SQL Injection, XSS)

3.2.2 Using Safe APIs and Libraries

4. Authentication in Java

4.1 User Authentication

4.1.1 Password Hashing

4.1.2 Multi-factor Authentication

4.2 OAuth and OpenID Connect

4.3 Single Sign-On (SSO)

5. Authorization in Java

5.1 Role-Based Access Control (RBAC)

5.2 Permissions and Privileges

5.3 Fine-Grained Authorization

5.4 Access Control Lists (ACL)

6. Java Network Security

6.1 Secure Sockets Layer (SSL) and Transport Layer Security (TLS)

6.2 Securing Web Applications

6.2.1 Securing Servlets

6.2.2 Securing RESTful APIs

7. Data Security in Java

7.1 Encryption and Decryption

7.1.1 Symmetric Encryption

7.1.2 Asymmetric Encryption

7.2 Java KeyStore

7.3 Secure Transmission and Storage of Data

8. Java Security Best Practices

8.1 Regular Security Audits

8.2 Monitoring and Logging

8.3 Security Patching and Updates

8.4 Continuous Integration and Security Testing

9. Java Security Tools

9.1 Static Code Analysis Tools

9.2 Dynamic Application Security Testing (DAST) Tools

9.3 Security Scanning and Monitoring Tools

10. Secure Deployment of Java Applications

10.1 Application Server Security

10.2 Container Security

10.3 Cloud Security Considerations

10.4 Real-world Security Challenges

10.5 How to Apply Security Principles in Java Projects