



Capstone Courseware, LLC

33 Boylston Street  
Jamaica Plain, MA 02130

877-227-2477  
capstonecourseware.com

## 107. Java Development for Secure Systems

### Version 1.4

This course exposes students to the broad range of challenges and techniques that is "Java security." As there is not one security problem for Java applications and components, but many, so there are many diverse solutions, found in various places in the Java architecture. This course starts with basic concepts of code security -- access controllers, permissions, and policies -- and good secure-coding practices. It introduces key and certificate management and code signing, and takes students through exercises in basic cryptography skills using the appropriate Java APIs; but then the heavy focus is on the Java Authentication and Authorization Service, or JAAS, and its increasing role in J2EE security.

The course emphasizes hands-on exercise, and students will spend more than half of their classroom time solving specific security problems. Most labs are organized as scenarios in which a security breach of existing software is possible - students begin by hacking the system in some way. Then the work of the lab is to tighten up the software to eliminate the threat: set a secure policy, sign a file, clean up overexposed parts of an API, require user login, etc.

This version of the course targets J2SE 1.4 and J2EE 1.4. For training within the latest Java environments, please see the latest version of this course.

### Prerequisites

- Solid Java programming experience is assumed - both structured and object-oriented techniques. Course 103 is excellent preparation for this course.
- Some knowledge of J2EE architecture and development is also required, though extensive practical experience with J2EE development is not strictly necessary. Our Course 108 offers a one-day overview of J2EE development, including architecture and working examples.



## Learning Objectives

- Understand how the Java language and standard-platform architecture solve many low-level security problems for all Java code.
- Design and implement security policies for Java applications, servers, and components.
- Manage keys and certificates for a Java application, and sign code sources as necessary.
- Practice secure design and coding, and balance usability with security in UI and API.
- Sign and verify application data and messages using the JCA, and encrypt/decrypt using the JCE.
- Incorporate JAAS authentication into an application.
- Implement a JAAS LoginModule to connect to your own application data.
- Secure J2EE applications by URL and role, and integrate JAAS authentication.

**Timeline: 3 days.**





## **Chapter 1. J2SE Security**

- Holistic Security Practices
- Threats to the User
- Protections in the Java Language
- The Class Loader and Bytecode Verifier
- System Classes and the Core API
- SecurityManager and AccessController
- Permissions
- Implication
- CodeSources
- Policies
- Configuring J2SE Security
- Dynamic Policies
- Privileged Actions

## **Chapter 2. Code Signature and Key Management**

- Encryption and Digital Signature
- Keystores
- Keys and Certificates
- Certificate Authorities
- The KeyStore API
- Signing JARs
- Signed CodeSources
- Additional Policy Semantics

## **Chapter 3. Secure Development Practices**

- Threats to the Code
- Presentation-Tier Vulnerabilities
- MVC and Security
- Validating User Input
- Injection Attacks
- Business-Tier Vulnerabilities
- Final Classes and Methods
- Inner Classes and Objects
- Singletons, Factories, and Flyweights
- Methods, Collections, and Data Hiding
- Sealing JARs
- Code Obfuscation
- Logging and Auditing
- Object Serialization





## Chapter 4. Cryptography

- Threats to Identity and Privacy
- The Java Cryptography Extensions
- The Signature Class
- SignedObjects
- The Java Cryptography Extensions
- SecretKeys and KeyGenerator
- The Cipher Class
- Dangerous Practices
- HTTP and JSSE

## Chapter 5. JAAS

- Pluggable Authentication Logic
- JAAS
- Packages and Interfaces
- Subjects and Principals
- ANDs and ORs
- Impersonation Methods
- Permissions for JAAS Use
- LoginContext and LoginModule
- Configuring JAAS
- CallbackHandler and Callbacks
- Implementing a JAAS Client
- Implementing a LoginModule

## Chapter 6. J2EE Security

- Threats for Enterprise Software
- J2EE Servers as Code Hosts
- Tomcat Security Configuration
- Securing URLs
- Declaring Roles
- Securing EJBs
- Programmatic Security
- JAAS in J2EE
- Realms and LoginModules
- JAAS in Tomcat
- Certifying a J2EE Application
- HTTPS Configuration





## Appendix A. Learning Resources

### System Requirements

**Hardware Requirements (Minimum)**

500 MHz, 256 meg RAM, 500 meg disk space.

**Hardware Requirements (Recommended)**

1.5 GHz, 512 meg RAM, 1 gig disk space.

**Operating System**

Tested on Windows XP Professional. Course software should be viable on all systems which support a J2SE 1.4 SDK.

**Network and Security**

Limited privileges required -- please see our standard security requirements at <http://capcourse.com/Guides/Security.html>.

**Software Requirements**

All free downloadable tools.

