



Trusted Labs, a department of  
Trusted Logic S.A.  
Société Anonyme  
au capital de 100000€  
RCS Versailles B 421 483 413

5 rue du Bailliage  
78000 Versailles – France  
Tel. +33 (0)1 30 97 26 20  
Fax. +33 (0)1 30 97 26 19  
www.trusted-labs.com

---

# JAVA CARD™ SECURITY TRAINING

—

## SECURE APPLICATION DEVELOPMENT

---

### **INTRODUCTION**

The aim of this training session is to raise the awareness of Java Card™ application developers about the possible security issues in their applications, and to teach them ways to address these issues.

When turning to Java Card, most smart card developers don't know how to handle security issues because they do not know how to handle the loss of control due to the addition of the Java Virtual Machine.

In this session, the various security issues faced by smart card application developers are introduced, and the respective responsibilities of the Java Card platform and application are spelled out. Some solutions are then proposed, and put into practice through exercises.

### **DETAILED PROGRAM**

The training session is organized in three parts, spread over two days.

#### *Part 1: Background and Java Card platform security*

This first part is entirely theoretical, and it consists of background information.

- **Java Card architecture and principles.**  
A brief presentation of the architecture of a Java Card Platform, introducing the various elements of the platform, followed by a brief description of Java Card's governing principles (interoperability, dynamic application management, etc.), and their impact on security.
- **Smart card security issues.**  
A recall of the main attacks that we cover in this training session, i.e., side channel attacks and perturbation attacks, including a presentation of the attacks that will be considered throughout the session.
- **Java Card platform security**
  - **Splitting the responsibility.**  
An explanation of how the security responsibility is split between the Java Card Platform and the application, with a description of the security elements within the platform.
  - **Trusting the platform.**  
Good questions to ask platform developers in order to trust them, or what could make a platform's security fail.
  - **Application management.**  
A short explanation of Java Card-specific application management issues, and the specific issues that need to be addressed.

*Part 2: Java Card application security*

This second part is the heart of the training session, and it may be mixed with the last part, which consists of practical examples.

- Java Card application security
  - Data confidentiality.  
A description of the possible uses of the Java Card API to protect the application's confidential data, and of the limits of these features.
  - Data integrity  
A description of the ways in which data integrity can be protected in Java Card, including common redundancy practices, and a discussion on the impact on performance.
  - Monitoring and auditing.  
Java Card includes many ways to program safely, which can also be used by applications in order to properly monitor potential attacks on their applications. In particular, unexpected failures can be managed in part by the application.
  - Authentication and lifecycle management.  
Authentication data and lifecycle states are usually considered as sensitive data. The tests of these values also are very sensitive; they may need to be protected by redundancy, and some techniques are discussed here.
  - The specific context of SIM Toolkit applications.  
SIM Toolkit applications are in a very specific context, which has an impact on security. First, there are additional APIs, such as the File System API, which may be useful for security, after checking that their implementation is appropriately protected. Then, there are many new communication mechanisms, which may introduce new security issues.

### *Part 3: Practical examples*

The third part consists of the study of the implementation of basic security countermeasures on common examples (PIN protection, lifecycle protection).

- Design for security
  - Command dispatching.  
Design the application interface in a way that allows the application to make use of the platform's features in an efficient way.
  - Lifecycle and security event management.  
A few things that are needed at the application level, and which are good not to forget when first designing the application.
- Writing sensitive code
  - Using the API  
How to properly use the API in order to rely on the Java Card Platform to do the difficult security work.
  - Managing confidentiality.  
How to make sure that confidential data remains confidential throughout the lifetime of a Java Card application.
  - Integrity and redundancy  
How to properly introduce redundancy in an application.
- Philosophical questions
  - Security and performance.  
Security has an impact on performance, which can be important if too many countermeasures are included at the application level. Is it worth it?
  - Security and readability.  
Secure code, in particular when redundancy has been included, is very difficult to read and maintain. Is it worth it?
  - Usefulness of a reference implementation.  
Some Java Card developers like to maintain two parallel versions: one clean, reference version, with no security and no performance optimization; and an industrial version, with all the features inside. Is it worth it?

### **PREREQUISITES**

It is assumed that the persons attending the training session have the following knowledge prior to the training session:

- Working knowledge of Java Card
- Some notions of smart card security principles.