Security and Trust Services APIs for Java 2 Platform, Micro Edition

---

Version 1.0
Reference Implementation Installation Guide

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, California 95054
U.S.A. 650-960-1300

July 2004

Please
Recycle

Contents

Preface

This document describes how to install the Security and Trust Services APIs for the Java 2™ Platform, Micro Edition 1.0 Reference Implementation.

Who Should Read This Guide

This Installation Guide should be read by J2ME developers working with the Security and Trust Services APIs 1.0 Reference Implementation.

Before You Read This Guide

In order to fully use the information in this document, you must have thorough knowledge of the topics discussed in these guides:

- *Java Card Platform, Version 2.2.1 Development Kit, User's Guide*
- *MIDP Reference Implementation, Version 2.0, Using MIDP*
- *MIDP Reference Implementation, Version 2.0, Creating MIDlet Suites*

How This Guide Is Organized

Chapter 1 describes the supporting software needed to install the SATSA 1.0 Reference Implementation, how to test if the SATSA 1.0 installation is successful, and information about SATSA 1.0 configuration.

Typographic Conventions

Typeface	Meaning	Examples
`AaBbCc123`	The names of commands, files, and directories; on-screen computer output	Edit your `.login` file. Use `ls -a` to list all files. `% You have mail`.
`AaBbCc123`	What you type, when contrasted with on-screen computer output	`% su` `Password:`
AaBbCc123	Book titles, new words or terms, words to be emphasized	Read Chapter 6 in the *User's Guide*. These are called *class* options. You *must* be superuser to do this.
	Command-line variable; replace with a real name or value	To delete a file, type `rm` *filename*.

Related Documentation

Application	Title
Release Notes	*Security and Trust Services APIs Reference Implementation: Release Notes*

Accessing Sun Documentation Online

The Source for Java Developers web site enables you to access Java™ platform technical documentation on the Web:

http://java.sun.com/reference/docs/index.html

Sun Welcomes Your Comments

We are interested in improving our documentation and welcome your comments and suggestions. Send us your comments at:

http://java.sun.com/docs/forms/sendusmail.html

CHAPTER 1

Installing the SATSA Reference Implementation

The Reference Implementation for the *S*ecurity and Trust Services APIs (SATSA) for Java 2 Platform, Micro Edition Specification provides an implementation of the four optional packages defined in that specification. These are:

- SATSA-APDU - Defines an API to support communication with smart card applications using the Application Protocol Data Unit (APDU) protocol.
- SATSA-JCRMI - Defines a Java Card Remote Method Invocation (JCRMI) client API that allows a Java™ 2 Platform, Micro Edition (J2ME™) application to invoke a method of a remote Java Card object.
- SATSA-PKI - Defines an API to support application-level digital signature signing and basic user credential management, using the Public Key Infrastructure (PKI) protocol.
- SATSA-CRYPTO - Defines a subset of the Java™ 2 Platform, Standard Edition (J2SE™) cryptography API. It provides basic cryptographic operations to support message digest, signature verification, encryption, and decryption.

The SATSA 1.0 Reference Implementation is designed to run on top of a J2ME platform. It also runs with the Java Card™ Platform Development Kit 2.2.1 Reference Implementation, as this is used to simulate the functionality of a security element.

This chapter provides information about the software you must install prior to downloading the SATSA 1.0 RI distribution, as well as detailed instructions for installing the SATSA RI. This chapter also describes how to test your SATSA 1.0 installation and provides additional information about configuration.

Locating and Downloading Supporting Software

The default platform for the SATSA 1.0 Reference Implementation is the Windows 2000/x86 platform. The SATSA 1.0 Reference Implementation is based on the Mobile Information Device Profile (MIDP) 2.0 code base.

Before downloading and installing the SATSA 1.0 RI distribution, you must have the following software installed and configured:

- Java 2 Platform, Standard Edition (J2SE) SDK, version 1.4.2, or the Java 2™ runtime environment, version 1.4.2. For complete instructions on how to download and install the J2SE 1.4.2 software, see:

  `http://java.sun.com/j2se/1.4.2/download.html`

- Java Card Platform Development Kit Reference Implementation, version 2.2.1. For complete instructions on how to download and install the Java Card Platform Development Kit 2.2.1 Reference Implementation, see:

  `http://java.sun.com/products/javacard/dev_kit.html`

Setting System Variables

The installation of the required software shown above is straightforward and can be accomplished by using the documentation included with each software set. When all the required software has been installed, you should have the following system variables set:

- `JAVA_HOME` - points to the location where you have installed your J2SE platform distribution. For example, `C:\j2sdk1.4.2_05`.
- `JC_HOME` - points to the location where you have installed your Java Card Platform Development Kit 2.2.1 Reference Implementation. For example, `C:\java_card_kit-2_2_1`.

You also should have appended your `PATH` and `CLASSPATH` variables:

- `PATH` - append with `%JAVA_HOME%\bin` and `%JC_HOME%\bin.`
- `CLASSPATH` - append with `%JAVA_HOME%\lib` and `%JC_HOME%\lib.`

Installing the SATSA Packages

The Security and Trust Services 1.0 Reference Implementation is based on the Mobile Information Device Profile (MIDP) version 2.0 code base. Therefore, no separate installation of MIDP 2.0 is required; everything you need is contained in the SATSA 1.0 RI distribution.

To install the SATSA 1.0 Reference Implementation:

1. **Copy the SATSA 1.0 RI distribution file, `satsa-1_0.zip`, into some location in your file system, for example, at the same level as your Java Card platform 2.2.1 distribution, and unzip.**

   This creates the directory C:\`satsa1.0`.

2. **Set the system variable `MIDP_HOME` so it points to the location where you have installed the SATSA 1.0 RI distribution.**

   For example, if you install to the location shown in Step 1, `%MIDP_HOME%` would point to `C:\satsa1.0`.

3. **Append `%MIDP_HOME%\bin` to your `PATH` variable.**

4. **Append `%MIDP_HOME%\lib` to your `CLASSPATH` variable.**

Contents of the Reference Implementation

The SATSA 1.0 Reference Implementation contains the following subdirectories:

- `appdb` - contains information used by the MIDP 2.0 device emulator.
- `bin` - contains SATSA 1.0 RI executables, including the device emulator executable file, `midp.exe`.
- `classes` - contains the SATSA 1.0 class files, including MIDP 2.0 and Java Card platform sample programs, and configuration files.
- `docs` - includes SATSA 1.0 documentation.
- `javacard_classes` - contains class files needed for Java Card platform interaction.
- `lib` - contains default configuration files and other items used by MIDP 2.0.

Running the SATSA 1.0 Reference Implementation

Running the SATSA 1.0 Reference Implementation involves two steps:

- Starting the MIDP 2.0 emulator
- Starting the Java Card (CRef) emulator

Starting the MIDP 2.0 Emulator

The SATSA 1.0 Reference Implementation is based on the MIDP 2.0 code base. To start the MIDP 2.0 emulator that incorporates the SATSA API, do the following:

1. **Start up a Windows 2000 command shell window.**
2. **Type:**

   C:\>`%MIDP_HOME%\bin\midp`

   This displays the MIDP 2.0 device emulator screen.

Note – Successful interaction with the MIDP 2.0 device emulator requires several setup steps that are outside the scope of this document, such as writing an HTML page to point to the sample midlet suites and running a web server such as Apache or TomCat to handle the HTTP requests sent by the emulator. For more information, see *Using MIDP* and *Creating Midlet Suites*, in the MIDP 2.0 documentation set.

Starting the Java Card Emulator

The SATSA 1.0 Reference Implementation works in conjunction with the Java Card platform 2.2.1 Reference Implementation. The Java Card platform 2.2.1 Reference Implementation include a Java Card platform emulator called CRef that simulates the functionality of a security element.

To simulate the SIM Application Toolkit (SAT) environment for the SATSA-APDU package and to provide the PKI functions the SATSA-PKI package, the corresponding instance of the CRef must use an EEPROM image contained in the following file:

`%MIDP_HOME%\bin\jc_eeprom_image`

In order to load this EEPROM image into CRef, do the following:

1. **Start up a Windows 2000 command shell window.**

2. **Enter the following command into the command prompt, without line breaks:**

   C:\>`%JC_HOME%\bin\cref.exe -p <`*port number*`> -i %MIDP_HOME%\bin\jc_eeprom_image`

   In the command line above, the <*port number*> should be one of the ports specified in the `com.sun.midp.io.j2me.apdu.hostsandports` property in the `internal.config` file. (For more information on the `internal.config` file, see "The `internal.config` File" in this chapter.)

   Once you have entered the `cref.exe` command above, you should see output as shown in CODE EXAMPLE 1.

CODE EXAMPLE 1 CRef Output

```
Java Card 2.2.1 C Reference Implementation Simulator (version 0.41)
32-bit Address Space implementation - no cryptography support
Copyright 2003 Sun Microsystems, Inc. All rights reserved.

Memory Configuration
    Type     Base      Size     MAX Addr
    RAM      0x0       0x500    0x4ff
    ROM      0x2000    0xa000   0xbfff
    E2P      0x10020   0xffe0   0x1ffff

    ROM Mask size =                  0x578c    22412 bytes
    Highest ROM address in mask =    0x778b    30603 bytes
    Space available in ROM =         0x4874    18548 bytes
EEPROM (0xffe0 bytes) restored from file
    "C:\satsa1.0\bin\jc_eeprom_image"
Using a pre-initialized Mask
```

At this point, CRef is ready to receive commands and respond to MIDlet requests.

For more information on working with EEPROM images and CRef, see the documentation for Java Card Platform Development Kit 2.2.1 Reference Implementation.

SATSA 1.0 Configuration Files

The SATSA 1.0 Reference Implementation contains two configuration files that, under most circumstances, can be left as-is and do not need to be altered or changed. These files are the following:

- The `internal.config` File
- The `system.config` File

Both of these configuration files are found in the directory:

`%MIDP_HOME%\lib`

The `internal.config` File

The `internal.config` file contains several MIDP 2.0 implementation-specific parameters. In a default installation of the SATSA 1.0 Reference Implementation, none of these parameters need to be changed. The parameters defined in the `internal.config` file specifically for the SATSA 1.0 RI are displayed in CODE EXAMPLE 2.

CODE EXAMPLE 2 The `internal.config` File

```
(.....)
com.sun.satsa.keygen: true
com.sun.satsa.opaquesig: true
com.sun.satsa.certsig: true
com.sun.midp.io.j2me.apdu.hostsandports: localhost:9025,
    localhost:9026
com.sun.midp.io.j2me.apdu.satselectedapdu:
    00.a4.04.07.a0.00.00.00.62.3.1.7F
com.sun.satsa.store_csr_list: true
(.....)
```

Many of the parameters shown above are optional and might not be supported on a specific platform. Setting the following parameters to false allows the SATSA 1.0 Reference Implementation to simulate platforms where those parameters are not supported (that is, to demonstrate the proper exceptions expected when the parameters are not supported by the platform).

The optional `internal.config` file parameters are:

- `keygen` - specifies that a key be generated. Setting this parameter to `false` specifies that no key be generated.
- `opaquesig` - specifies that opaque signatures be used. Setting this parameter to `false` does not allow the use of opaque signatures.
- `certsig` - specifies that a certificate signature be used. Setting this parameter ro `false` does not allow the use of certificate signatures.

The default SATSA 1.0 RI configuration defines two card slots, `0H` and `1H`. In CODE EXAMPLE 2, above, the parameter `hostsandports` provides the location of the Java Card platform emulator for each of the configured slots (that is, on the machine `localhost`, slot `0H` of the Java Card emulator listens on socket `9025` and slot `1H` listens on socket `9026`).

The `system.config` File

The `system.config` file defines several parameters, only one of which is used by the SATSA 1.0 Reference Implementation. CODE EXAMPLE 3 describes the `microedition.smartcardslots` parameter, which defines the slots where the Java Card platform emulator listens on the machine `localhost` (shown in CODE EXAMPLE 2, above).

CODE EXAMPLE 3 The `system.config` File

```
(.....)
microedition.smartcardslots: 0H, 1H
(.....)
```

For additional information regarding the `microedition.smartcardslots` parameter, see the *Security and Trust Services APIs for the Java 2 Platform, Micro Edition Specification*.

Glossary

AES Advanced Encryption Standard. The successor to the DES algorithm. It has a fixed block size of 128 bits and a key size of 128, 192, or 256 bits.

APDU Application Protocol Data Units. A protocol used to define the structure of data messages exchanged between smart cards and smart card readers.

CLDC Connected Limited Device Configuration. In conjunction with MIDP, provides the Java runtime environment for wireless and handheld devices.

DES Data Encryption Standard. The most well-known and widely-used symmetric cryptographic algorithm.

J2ME Java 2 Platform, Micro Edition. A scaled-down version of the Java platform specifically designed to run in the reduced memory space of a wireless, handheld, or other small device.

J2SE Java 2 Platform, Standard Edition. The core Java technology platform.

Java Card A smart card that has the capability of running Java code.

JCP Java Community Process™ (JCP™). The process used by the world-wide community of Java developers for formulating Java-based standards and evaluating specifications.

JCRE Java Card Runtime Environment. The execution environment for Java Card applets.

JCRMI Java Card Remote Method Invocation. A subset version of Java 2 Platform, Standard Edition RMI, to be used with the Java Card platform.

JSR Java™ Specification Request. A specification submitted to the Java Community Process for consideration and review.

JVM Java™ Virtual Machine.[1] The execution environment for Java programs.

MIDP Mobile Information Device Profile. In conjunction with CLDC, provides the Java runtime environment for wireless and handheld devices.

PKI Public Key Infrastructure. The infrastructure used to create, exchange, and manage user credentials, public and private keys, and digital signatures.

1.The terms “Java Virtual Machine” and “JVM” mean a Virtual Machine for the Java™ platform.

RI Reference Implementation. A software package created to illustrate the concepts and APIs provided in a programming specification.

SATSA Security and Trust Services APIs. A set of four J2ME optional packages that provide communication protocols, such as APDU and JCRMI, and security capabilities, such as PKI and encryption, for ensuring secure transactions between J2ME programs and a security element.

SE Security Element. A smart card or other item that provides secure storage of private keys, certificates, digital signatures, and user data.

Index