June 5, 2003
File name: ref-imp5.doc

Requirements for GSC-IS Reference Implementations

Eric Dalci
Elizabeth Fong
Alan Goldfine

National Institute of Standards and Technology
Information Technology Laboratory

1. Introduction:

NIST is working on a conformance test suite that will enable the testing of candidate smart card systems for conformance with the Government Smart Card – Interoperability Specification (GSC-IS). The suite tests the C and Java bindings of the Basic Service Interface (BSI), and the Card Edge Interface (CEI). To be in conformance with the GSC-IS at the respective interface, an implementation must demonstrate "correct" behavior as expected by the conformance test suite.

To validate the test suite, NIST is looking for a reference implementation (RI) of a Service Provider Module (SPM) that implements at least one of the BSI bindings together with the CEI.

2. Purpose of a Reference Implementation:

A reference implementation is, in general, an implementation of a specification to be used as a definitive interpretation for that specification. During the development of the GSC-IS conformance test suite, at least one relatively trusted implementation of each interface is necessary to (1) discover errors or ambiguities in the specification, and (2) validate the correct functioning of the test suite. No such implementations currently exist, hence the need for a reference implementation.

3. Degree of Conformance of the RI with the Specification:

At present, the GSC-IS requires that all products provide complete implementations of the particular interfaces. Certainly this must be true of a RI. A more complex structure of "levels" of functionality is being considered for future releases of the GSC-IS whereby a product may choose to implement a specified subset of the functionality at a particular interface. If such an approach is adopted, a RI could be developed for each level.

4. Timing Requirement of a RI:

Since a GSC-IS RI will be used only to validate the NIST conformance test suite, and to serve as a prototype implementation before a "real" implementation is available, a RI needs to be available within the next 2-3 months (August, 2003).

5. Version Requirement:

A usable RI should be implemented according to GSC-IS version 2.1.

6. Platform Requirement of a RI:

To be useful for the conformance test development process, a RI must be able to operate on a PC running Microsoft Windows (9X, NT, 2000 or XP) or Linux.

7. Card Reader/Driver Requirement of a RI:

The card reader or card acceptance device associated with a RI must be

- a contact card reader
- able to connect to a PC via the serial port (RS232), USB Port or PCMCIA slot.
- compatible with Microsoft Windows (9X, NT, 2000, XP) and Linux
- ISO 7816 compliant
- PC/SC compliant
- capable of communicating with its reader driver using the appropriate protocols.

The driver associated with a RI must be

- compatible with PC/SC
- capable of supporting both the T=0 and T=1 protocols (ISO 7816 protocols)
- capable of communicating with the smart card reader using the appropriate protocols.

8. Software Requirement of RI:

Each RI must consists of a SPM containing one of the following:

1 - an implementation of the C language binding of the BSI and the CEI.
2 - an implementation of the Java language binding of the BSI and the CEI.

The card edge level testing can be done using the "passthru" command of the BSI to "tunnel" the data from the CEI to the testing client application.

For quality assurance reasons during test suite development only, NIST would like to have access to the source code of the RI.

A BSI C binding implementation must be capable of accepting the BSI C functions issued with appropriate parameters, and returning the expected return code and values specified in GSC-IS Chapter 4.

A BSI Java binding implementation must be capable of accepting the BSI Java methods issued with appropriate parameters, and returning/throwing the expected return condition, return codes and values specified in GSC-IS Chapter 4.

A CEI implementation must be able to

- accept the GSC default APDUs issued with appropriate values as specified in GSC-IS Chapter 5
- parse a card's CCC and read the registered data model’s required fields
- map a card's native APDU to the default APDU using the CCC grammar specified in GSC-IS Chapter 6
- return the expected return status bytes as specified in GSC-IS or as defined by ISO.

9. Access Control Rules Requirement of a RI:

A RI must be able to support all Access Control Rules as specified in the GSC-IS. The RI must also be able to support all cryptographic algorithms as specified in the GSC-IS.

10. Smart Card Requirement:

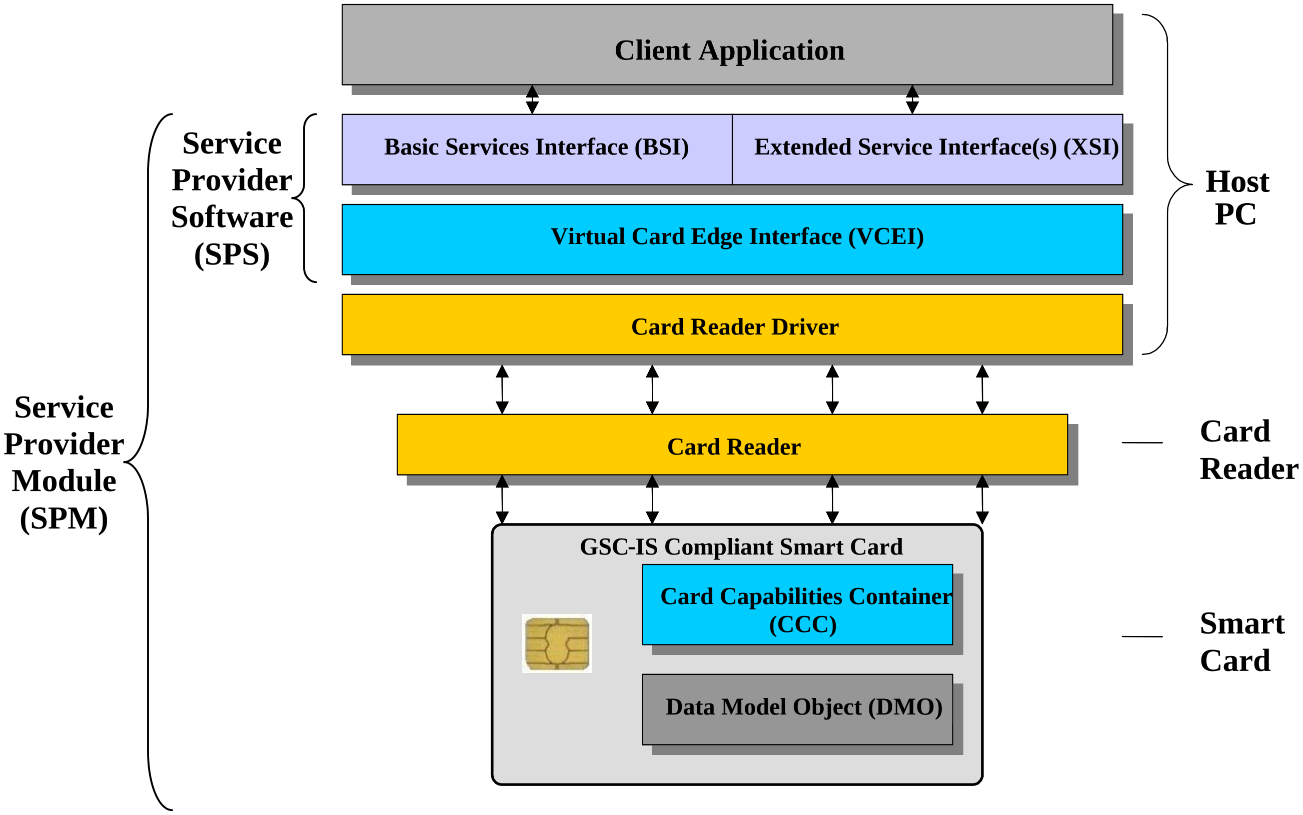
It would be useful to have a GSC-IS reference smart card. Such a card (file system or virtual machine) must contain a CCC and a registered data model as defined in the GSC-IS.

11. Implementation Guidance:

The Service Provider Software of a RI can be built using OCF or PC/SC.

12. Service Provider Software:

The following diagram show the composition of the SPS and the SPM.


Client Application
Service Provider Software (SPS)
Basic Services Interface (BSI)
Extended Service Interface(s) (XSI)
Virtual Card Edge Interface (VCEI)
Card Reader Driver
Host PC
Service Provider Module (SPM)
Card Reader
Card Reader
GSC-IS Compliant Smart Card
Card Capabilities Container (CCC)
Data Model Object (DMO)
Smart Card

Note : CEI is a synonymous for VCEI (Virtual Card Edge Interface)