July 2002

ITL Bulletin

ADVISING USERS ON INFORMATION TECHNOLOGY

OVERVIEW: THE GOVERNMENT SMART CARD INTEROPERABILITY SPECIFICATION

By Jim Dray and Teresa Schwarzhoff,
Computer Security Division,
Information Technology Laboratory,
National Institute of Standards and Technology

Background

Smart cards are small, credit card-sized devices containing a microprocessor and semiconductor memory. These cards provide a cost-effective and highly secure mechanism for automated systems to verify the identity of human users. Smart cards are capable of generating digital signatures, encrypting sensitive information, and many other security-related functions.

A typical configuration for a smart card system consists of a host computer with one or more smart card readers attached to hardware communications ports. Smart cards can be inserted into the readers, and software running on the host computer communicates with these cards using a protocol defined by ISO Standard 7816-4. The ISO standard smart card communications protocol defines Application Protocol Data Units (APDUs) that are exchanged between smart cards and host computers. This APDU-based interface is referred to as the “card edge,” and the two terms are used interchangeably.

Client applications have traditionally been designed to communicate with ISO smart cards using the APDU protocol through low-level software drivers that provide an APDU transport mechanism between the client application and a smart card. Smart card families can implement the APDU protocol in a variety of ways, so client applications must have intimate knowledge of the APDU set of the smart card with which they are communicating. This is generally accomplished by programming a client application to work with a specific card, since it would not be practical to design a client application to accommodate the different APDU sets of a large number of smart card families.

The tight coupling between client applications and smart card APDU sets has several drawbacks. Applications programmers must be thoroughly familiar with smart card technology and the complex APDU protocol. If the cards that an application is hard coded to use become commercially unavailable, the application must be redesigned to use different cards. Customers also have less freedom to select different smart card products, since their applications will work only with one or a small number of similar cards.

The Government Smart Card Interoperability Specification (GSC-IS) provides solutions to a number of the interoperability problems associated with smart card technology. The original version of the GSC-IS (version 1.0) was developed by the GSC Interoperability Committee led by the General Services Administration (GSA) and NIST, in association with the Smart Access Common Identification Card contract (Contract No. GS00T00ALD0208). NIST published version 2.0 of the GSC-IS in June 2002 as NIST Interagency Report 6887. The Government Smart Card Interagency Advisory Board unanimously adopted NISTIR 6887 as the GSC-ISv2.0 on July 9, 2002. This specification is the foundation of the federal government’s effort to develop a ubiquitous Smart Card Interoperability Framework that enables large-scale deployment of smart card technology across federal agencies. The specification is available at http://smartcard.nist.gov.

Continued on page 2

ITL Bulletins are published by the Information Technology Laboratory (ITL) of the National Institute of Standards and Technology (NIST). Each bulletin presents an in-depth discussion of a single topic of significant interest to the information systems community. **Bulletins are issued on an as-needed basis** and are available from ITL Publications, National Institute of Standards and Technology, 100 Bureau Drive, Stop 8901, Gaithersburg, MD 20899-8901, telephone (301) 975-2832. To be placed on a mailing list to receive future bulletins, send your name, organization, and business address to this office. You will be placed on this mailing list only.

Bulletins issued since December 2000

- ❐ *A Statistical Test Suite for Random and Pseudorandom Number Generators For Cryptographic Applications*, December 2000
- ❐ *What Is This Thing Called Conformance?* January 2001
- ❐ *An Introduction to IPsec (Internet Protocol Security),* March 2001
- ❐ *Biometrics – Technologies For Highly Secure Personal Authentication*, May 2001
- ❐ *Engineering Principles for Information Technology Security,* June 2001
- ❐ *A Comparison of The Security Requirements for Cryptographic Modules In FIPS 140-1 and FIPS 140-2*, July 2001
- ❐ *Security Self-assessment Guide For Information Technology Systems*, September 2001
- ❐ *Computer Forensics Guidance*, November 2001
- ❐ *Guidelines on Firewalls and Firewall Policy,* January 2002
- ❐ *Risk Management Guidance for Information Technology Systems,* February 2002
- ❐ *Techniques for System and Data Recovery,* April 2002
- ❐ *Contingency Planning Guide for Information Technology Systems*, June 2002

The GSC Architectural Model

The GSC-IS provides interoperability at two levels: the service call level and the card command (APDU) level. A brief explanation of these interoperability levels follows:

- **Service Call Level**: This level is concerned with functional calls required to obtain various services from the card (e.g., encryption, authentication, digital signatures, etc.). The GSC-IS addresses interoperability at this level by defining an Applications Programming Interface (API) called the Basic Services Interface (BSI) that defines a common high-level model for smart card services. The module that implements the BSI and thus provides an interoperable set of smart card services to client applications is called the Smart Card Service Provider Module (SCSPM). These services are logically divided into three modules that provide utility, secure data storage, and cryptographic services. Since a SCSPM could potentially be implemented through a combination of hardware and software, the software component of the SCSPM is referred to as the Service Provider Software (SPS).
- **Card Command Level**: This level is concerned with the exact APDUs that are sent to the card to obtain the required service. The GSC-IS addresses interoperability at this level by defining the API called the Virtual Card Edge Interface (VCEI), consisting of a card-independent standard set of APDUs that supports the functions defined in the BSI and implemented by the SCSPM.

Certain data sets need to be available in the card to support the interoperability provided by the BSI and VCEI. To ensure that there is a standard format (or schema) for storing these data sets and to enable uniform access and interpretation, the GSC-IS also defines Data Models. These Data Models provide data portability across GSC-IS compliant card implementations, ensuring that a core set of data elements is available on all cards. The storage entities for various categories of data sets are called containers. One of these containers, the Card Capabilities Container (CCC), is mandatory, while the other containers composing a Data Model are optional. The CCC describes the differences between a smart card's native APDU set and the standard APDU set defined by the VCEI. An SPS retrieves a smart card's CCC and uses it to perform the translation between the VCEI and the card's native APDU set. The GSC-IS accommodates any smart card whose APDU set can be mapped to the VCEI via a CCC definition.

The Basic Services Interface

All Smart Card Service Provider Modules implement the BSI. The BSI is logically organized into three provider modules:

- **Utility Provider Module:** Provides utility services for obtaining a list of available card readers, establishing and terminating logical connections with a smart card, etc.
- **Generic Container Provider Module:** Provides a unified abstraction of the storage services of smart cards, presenting applications with a simple interface for managing generic containers of data elements in Tag/Length/Value format.
- **Cryptographic Provider Module:** Provides fundamental cryptographic services such as random number generation, authentication, and digital signature generation.

The capabilities of a given SCSPM depend on the smart card available to the SCSPM when a client application requests a service through a BSI call. In cases where a service is not available, the BSI call returns an error code indicating that the requested service is not available. For example, a user may insert a smart card that does not have public key cryptographic capabilities and then perform an operation that causes a client application to request a digital signature calculation from the associated SCSPM. Since the smart card cannot provide this service, the BSI returns a "service not available" error code to the client application.

Extended Service Interfaces

Because the BSI is not a complete operational interface, real world SCSPM implementations must support additional functionality outside the BSI domain. Because the BSI provides an interoperable interface, it is unable to address all operational requirements. Therefore, real world SCSPM implementations must support additional functionality outside the BSI domain. An SCSPM may include an Extended Service Interface (XSI) that provides non-interoperable functions. Since XSIs are implementation and applications specific, they are accommodated by the GSC-IS architectural model but are not defined in the GSC-IS. Card initialization and cryptographic key management are examples of functions that must currently be implemented in the XSI domain.

The Virtual Card Edge Interface

ISO 7816-4 defines a hierarchical file system structure for smart cards. Smart cards that conform to ISO 7816-4 are therefore known as "file system" cards. The Card Operating System program of a file system card is usually hard coded into the logic of the smart card integrated circuit during the manufacturing process and cannot be changed thereafter.

In recent years, other smart card architectures have been created that allow

ITL Bulletins Via E-Mail

We now offer the option of delivering your ITL Bulletins in ASCII format directly to your e-mail address. To subscribe to this service, send an e-mail message from your business e-mail account to listproc@nist.gov with the message **subscribe itl-bulletin**, and your name, e.g., John Doe. For instructions on using listproc, send a message to listproc@nist.gov with the message **HELP**. To have the bulletin sent to an e-mail address other than the From address, contact the ITL editor at 301-975-2832 or elizabeth.lennon@nist.gov.

developers to load executable programs onto smart cards after the cards have been manufactured. As one example, JavaCard™ defines a Java Virtual Machine (VM) specification for smart card processors. Developers can load compiled Java applets onto a smart card containing the JavaCard™ VM, programmatically changing the behavior of the card.

Due to the widespread adoption of the JavaCard™ specification, the term "virtual machine smart card" is often used generically to refer to any smart card whose Card Operating System can be extended by loading executable programs onto the card (regardless of whether that card conforms to the JavaCard™ specification). This GSC-IS uses the term "virtual machine smart card" in the general sense. A virtual machine smart card can theoretically be programmed to support any communications protocol, including the APDU-based protocols of the ISO 7816-4 standard.

The VCEI defines default sets of interoperable APDU level commands for both virtual machine and file system smart cards. The SPS of an SCSPM uses the information provided by a smart card's CCC to map that card's native APDU set to the VCEI default set. The VCEI consists of:

- A card edge definition for file system cards
- A card edge definition for VM cards, composed of three providers:
  - A generic container provider
  - A symmetric key (SKI) cryptographic service provider
  - A public key infrastructure (PKI) cryptographic service provider.

The card level providers of the VCEI directly support the service provider modules of the BSI. Card level providers are concrete implementations of the services that comprise the VCEI and are physically implemented on GSC compliant smart cards.

Data Models

Each GSC-IS compliant smart card conforms to a GSC data model. GSC data models define the set of containers and data elements within each container for cards supporting that data model. This version of the GSC-IS defines two data models: the original J.8 data model from the GSC-IS v1.0 and the Department of Defense Common Access Card data model. The Card Capabilities Container is the only mandatory container in either data model. The remaining containers and data elements are optional. However, if an implementation requires any of the containers and data elements defined in the data models, the containers and data elements must conform to the data model definitions.

Containers are accessed through the Generic Container Provider Module of the BSI. Access to the containers is subject to the Access Control Rules (ACRs) of the GSC-IS Security Model.

Card Capabilities Container

Each GSC-IS compliant card carries a Card Capabilities Container (CCC). The CCC is one of the mandatory containers that must be present in all GSC data models. The purpose of the CCC is to describe the differences between a given card's APDU set and the APDU set defined by the GSC-IS Virtual Card Edge Interface. The GSC-IS provides standard mechanisms for retrieving a CCC from a smart card. Once the CCC for a particular card is obtained, software on the host computer (specifically, the SPS) uses this information to translate between the VCEI and the card's native APDU set. Deviations from the card's data model structure can also be represented in a CCC.

The CCC allows each GSC-IS smart card to carry the information needed by the SPS to communicate with that card. This general mechanism for dynamically translating APDU sets eliminates the need to distribute, install, and maintain card specific APDU level drivers on host computer systems.

Service Provider Software

The SPS component of an SCSPM implements the BSI and the VCEI. It is responsible for retrieving CCCs from cards, using this information to translate between the smart card's native APDU set and the VCEI, and for handling the details of APDU level communications with the card. SPS implementations work with a particular card reader driver layer that transports APDUs between the SPS and the smart card.

NIST's Role in the GSC Program

NIST serves as the principal architect for the GSC program and is currently working with government and industry partners under the auspices of the GSC Interagency Advisory Board to:

- Pursue standardization and adoption of the GSC-IS
- Provide guidance on GSC security testing
- Establish a GSC interoperability conformance test program
- Establish a GSC developer community support program.

JavaCard™ is a trademark of Sun Microsystems, Inc., in the United States and other countries.

Disclaimer: Any mention of commercial products or reference to commercial organizations is for information only; it does not imply recommendation or endorsement by NIST nor does it imply that the products mentioned are necessarily the best available for the purpose.

Who we are

The Information Technology Laboratory (ITL) is a major research component of the National Institute of Standards and Technology (NIST) of the Technology Administration, U.S. Department of Commerce. We develop tests and measurement methods, reference data, proof-of-concept implementations, and technical analyses that help to advance the development and use of new information technology. We seek to overcome barriers to the efficient use of information technology, and to make systems more interoperable, easily usable, scalable, and secure than they are today. Our web site is http://www.itl.nist.gov/.

U.S. DEPARTMENT OF COMMERCE
National Institute of Standards and Technology
100 Bureau Drive, Stop 8900
Gaithersburg, MD 20899-8900