# An Introduction to Smart Cards

**by Steve Petri**

## Introduction

It has been said that smartcards will one day be as important as computers are today. This statement contains a bit of an error because it implies that smartcards are not computers, when in fact, they are. In this chapter we will describe the history of smartcards, some different types, their low-level properties, the standards that affect their adoption in mainstream society, and how they relate to today’s computer security systems.

Because smartcards are indeed tiny computers, it’s difficult to predict the variety of applications that will be possible with them in the future. It’s quite possible that smartcards will follow the same trend of rapid increases in processing power that computers have, following "Moore’s Law" and doubling in performance while halving in cost every eighteen months.

Smartcards have proven to be quite useful as a transaction/authorization/identification medium in European countries. As their capabilities grow, they could become the ultimate thin client, eventually replacing all of the things we carry around in our wallets, including credit cards, licenses, cash, and even family photographs. (The photographs could be viewed and/or exchanged by capable terminals or personal computers.) By containing various identification certificates, smartcards could be used to voluntarily identify attributes of ourselves no matter where we are or to which computer network we are attached.

We won't try to predict the future of smartcard application possibilities, nor their impact on society, but instead we focus on the state of the art for smartcards and their use in computer and network security systems. It is not scientifically comprehensive regarding every detail of integrated circuit cards, but instead tries to strike a balance between accuracy and comprehensibility. The standards and references that are mentioned throughout the chapter can be used to find more specific information.

## History

The roots of the current day smartcard can be traced back to the US in the early 1950s when Diners Club produced the first all-plastic card to be used for payment applications. The synthetic material PVC was used which allowed for longer-lasting cards than previously conventional paper based cards. In this system, the mere fact that you were issued a Diners Club card allowed you to pay with your "good name" rather than cash. In effect, the card identified you as a member of a select group, and was accepted by certain restaurants and hotels that recognized this group.

VISA and MasterCard then entered the market, but eventually the cost pressures of fraud, tampering, merchant handling, and bank charges made a machine-readable card necessary. The magnetic stripe was introduced, and this allowed further digitized data to be stored on the cards in a machine-readable format. This type of embossed card with a magnetic stripe is still the most commonly used method of payment. Magnetic stripe technology suffers from a critical weakness, however, in that anyone with access to the appropriate device can read, re-write, or delete the data. Thus a mag-stripe card is unsuitable for storing sensitive data and, as such, requires an extensive on-line, centralized, back-end infrastructure for verification and processing.

As it turns out, this type of a back-end infrastructure became available in the US but was not as readily available in the European countries. As in any client/server architecture, one solution to a lack of back-end processing power is to beef up the back-end server side, but another solution is to make the client piece more powerful, thus relieving some of the duties of the back-end. European countries seem to have preferred the client side approach, and made a huge improvement over mag-stripe technology by introducing the integrated circuit card (ICC).

In 1968, German inventors Jürgen Dethloff and Helmut Grötrupp applied for the first ICC related patents. Similar applications followed in Japan in 1970 and France in 1974. In 1984, the French PTT (Postal and Telecommunications services) successfully carried out a field trial with telephone cards. By 1986, many millions of French telephone smartcards were in circulation. Their number reached nearly 60 million in 1990, and 150 million are projected for 1996.

As cryptography made great progress in the 1960s and security mechanisms could be proved mathematically, smartcards proved to be an ideal medium for safely storing cryptographic keys and algorithms. French banks were the first field this type of a card by introducing a chip-incorporating bank card in 1984. German banks began introducing them around 1997. Another application fielded in Germany included over 70 million smartcards issued which carried health insurance information.

## Types of Cards

The International Organization for Standardization (ISO) standard 7810 "Identification Cards – Physical Characteristics" defines physical properties such as flexibility, temperature resistance, and dimensions for three different card formats (ID-1, ID-2, and ID-3). The Smart Card standard, ISO 7816, is based on the ID-1 format. In order to give perspective, several different types of ID-1 cards will be described in this section. One type in particular, namely cryptographic coprocessor cards, are becoming very important to current computer and network security systems.

### Embossed

Embossing allows for textual information or designs on the card to be transferred to paper by using a simple and inexpensive device. ISO 7811 specifies the embossed marks, covering their form, size, embossing height, and positioning. Transfer of information via embossing may seem primitive, but the simplicity of the system has made worldwide proliferation possible.

### Magnetic Stripe

The primary advantage that magnetic stripe technology offers over embossing is a reduction in the flood of paper documents. Parts 2, 4, and 5 of ISO 7811 specify the properties of the magnetic stripe, coding techniques, and positioning. The stripe's storage capacity is about 1000 bits and anyone with the appropriate read/write device can view or alter the data.

### Smartcards

The following Integrated Circuit Cards have conventionally come to be known as "Smartcards". These are the newest and most clever additions to the ID-1 family, and they also follow the details laid down in the ISO 7816 series. These types of cards allow far greater orders of magnitude in terms of data storage – cards with over 20 Kbytes of memory are currently available. Also, and perhaps most important, the stored data can be protected against unauthorized access and tampering. Memory functions such as reading, writing, and erasing can be linked to specific conditions, controlled by both hardware and software. Another advantage of smartcards over magnetic stripe cards is that they are more reliable and have longer expected lifetimes.

### Memory Cards

Though referred to as smartcards, memory cards are typically much less expensive and much less functional than microprocessor cards. They contain EEPROM and ROM memory, as well as some address and security logic. In the simplest designs, logic exists to prevent writing and erasing of the data. More complex designs allow for memory read access to be restricted. Typical memory card applications are pre-paid telephone cards and health insurance cards.

### Microprocessor Cards

Components of this type of architecture include a CPU, RAM, ROM, and EEPROM. The operating system is typically stored in ROM, the CPU uses RAM as its working memory, and most of the data is stored in EEPROM. A rule of thumb for smartcard silicon is that RAM requires four times as much space as EEPROM, which in turn requires four times as much space as ROM. Typical conventional smartcard architectures have properties reflected in Table 17-1.

**Table 17-1: Conventional Smartcard Architectures**

| | |
|---|---|
| RAM | 256 bytes to 1 Kbytes |
| EEPROM | 1 Kbytes to 16 Kbytes |

| | |
|---|---|
| ROM | 6 Kbytes to 24 Kbytes |
| Microprocessor | 8 bits at approximately 5 MHz |
| Interface Speed | 9600bps minimum, half duplex |

The serial I/O interface usually consists of a single register, through which the data is transferred in a half duplex manner, bit by bit. Though the chip can be thought of as a tiny computer, the external terminal must supply the voltage, ground, and clock.

**Cryptographic Coprocessor Cards**
Though technically these are in the category of microprocessor cards, they are separated here because of differences in cost and functionality. Because the common asymmetric cryptographic algorithms of the day (such as RSA) require very large integer math calculations, an 8 bit microprocessor with very little RAM can take on the order of several minutes to perform a 1024 bit private key operation. However, if a cryptographic coprocessor is added to the architecture, the time required for this same operation is reduced to around a few hundred microseconds. The coprocessors include additional arithmetic units developed specifically for large integer math and fast exponentiation. There is a drawback, however, and it is the cost. The addition of a cryptographic coprocessor can increase the cost of today's smartcards by 50% to 100%. These cost increases will likely diminish as coprocessors become more widespread.

In spite of the increased cost, the benefits to computer and network security of including the cryptographic coprocessor are great, for it allows for the private key to never leave the smartcard. As we'll see in the following sections, this becomes a critical factor for operations such as digital signatures, authentication, and non-repudiation. Eventually, though, the need for a cryptographic coprocessor and its associated cost will likely go away. The basic processors could become powerful enough to perform the math-intensive operations, or other algorithms such as those based on elliptic curve technology could become popular. Elliptic curve algorithms provide strong security without the need for large integer math, but haven't yet found their way into widespread use.

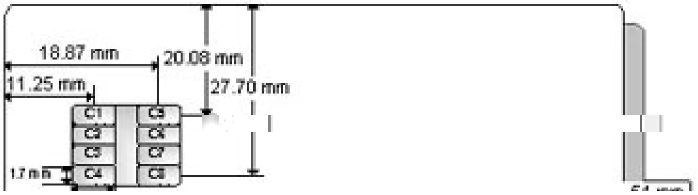
**Contactless Smartcards**
Though the reliability of smartcard contacts has improved to very acceptable levels over the years, contacts are one of the most frequent failure points any electromechanical system due to dirt, wear, etc. The contactless card solves this problem and also provides the issuer an interesting range of new possibilities during use. Cards need no longer be inserted into a reader, which could improve end user acceptance. No chip contacts are visible on the surface of the card so that card graphics can express more freedom. Still, despite these benefits, contactless cards have not yet seen wide acceptance. The cost is higher and not enough experience has been gained to make the technology reliable. Nevertheless, this elegant solution will likely have its day in the sun at some time in the future.

**Optical Memory Cards**
ISO/IEC standards 11693 and 11694 define standards for optical memory cards. These cards can carry many megabytes of data, but the cards can only be written once and never erased with today's technology. Though the read and write devices for optical cards are still very expensive, they may find use in applications such as health care where large amounts of data must be stored.

## Physical and Electrical Properties

The physical size of a smartcard is designated as ID-1, and is described in ISO 7810. The dimensions are 85.6 mm by 54 mm, with a corner radius of 3.18 mm and a thickness of 0.76mm. At the time ISO 7810 was created in 1985, it did not address chip placement but instead addressed embossing, magnetic stripes, etc. Smartcard chip placement is defined in ISO 7816-2, which is dated 1988. These physical characteristics are depicted in Figure 17-1.

**Figure 17-1: Smartcard Physical Dimensions**



The minimum requirements as far as card robustness are specified in ISO 7810, 7813, and 7816 part 1. These specifications address such things as UV radiation, X-ray radiation, the card's surface profile, mechanical robustness of card and contacts, electromagnetic susceptibility, electromagnetic discharges, and temperature resistance. ISO/IEC 10373 specifies the test methods for many of these requirements.

The electrical specifications for smartcards are defined in ISO/IEC 7816 parts 2 and 3, and GSM 11.11. Most smartcards have eight contact fields on the front face, however, two of these are reserved for future use so some manufacturers produce cards with only six contact fields, which slightly reduces production costs. Electrical contacts are typically numbered C1 through C8 from top left to bottom right. Figure 17-2 shows the layout of these contacts for both the 6 field and 8 field configurations. Table 17-2 describes their functions.

**Figure 17-2: Smartcard Electrical Contacts**

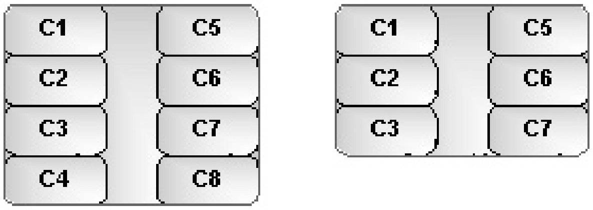


**Table 17-2: Description of Contacts**

| POSITION | TECHNICAL ABBREVIATION | FUNCTION |
|---|---|---|
| C1 | Vcc | Supply Voltage |
| C2 | RST | Reset |
| C3 | CLK | Clock Frequency |
| C4 | RFU | Reserved for future use |
| C5 | GND | Ground |
| C6 | Vpp | External programming voltage |
| C7 | I/O | Serial input/output communications |
| C8 | RFU | Reserved for future use |

The Vpp contact was used several years ago to supply voltage to EEPROMs for programming and erasing. However, with the advent of charge pumps that exist on the chip, the Vpp contact is rarely used today. The Vcc supply voltage is specified at 5 volts ± 10%. There is an industry push for smartcard standards to support 3 volt technology because all mobile phone components are available in a 3 volt configuration, and smartcards are the only remaining component which require a mobile phone to have a charge converter. It is theoretically possible to develop 3-volt smartcards, but interoperability with current 5-volt systems would be a problem. Nonetheless, a wider voltage range handling 3 to 5 volts will probably become mandatory in the near future.

## Operating System

Though typically only a few thousand bytes of program code, the operating system for the smartcard microprocessor must handle such tasks as:

- Data transmission over the bi-directional, serial terminal interface
- Loading, operating, and management of applications
- Execution control and Instruction processing
- Protected access to data
- Memory Management
- File Management
- Management and Execution of cryptographic algorithms

In contrast to personal computer operating systems such as Unix, DOS, and Windows, smartcard operating systems do not feature user interfaces or the ability to access external peripherals or storage media. The size is typically between 3 and 24 Kbytes. The lower limit is that used by specialized applications and the upper limit by multi-application operating systems.

Because smartcard memory space is so severely limited, not all standardized instructions and file structures can be generally implemented in all smartcard operating systems. For this reason, so-called "Profiles" have been introduced in ISO 7816-4 and EN 726-3. A profile defines the minimum requirements for data structures and commands. For example, Profile O in ISO 7816-4 defines the following minimums:

**Table 17-3: Profile O**

| | |
|---|---|
| Data Structures: | Transparent |
| | Linear Fixed |
| | Linear Variable |
| | Cyclic |
| Commands: | READ BINARY, UPDATE BINARY, no implicit selection and maximum length up to 256 bytes |
| | READ RECORD, UPDATE RECORD, without automatic selection |
| | APPEND RECORD |
| | SELECT FILE |
| | VERIFY |
| | INTERNAL AUTHENTICATE |
| | EXTERNAL AUTHENTICATE |
| | GET CHALLENGE |

## Cryptographic Capabilities

Current state of the art smartcards have sufficient cryptographic capabilities to support popular security applications and protocols. This section describes common capabilities found in the crypto-enabled smartcards from leading vendors.

RSA signatures and verifications are supported with a choice of 512, 768, or 1024 bit keylengths. The algorithms typically use the Chinese Remainder Theorem (CRT) in order to speed up the processing. Even at the 1024 bit keylength, the time needed to perform a signature is typically under one second. Usually the EEPROM file that contains the private key is designed such that the sensitive key material never leaves the chip. Even the card holder can’t access the key material in this case. The usage of the private key is protected by the user’s PIN, so that possession of the card does not imply the ability to sign with the card. RSA’s PKCS#1 padding is implemented by some cards.

Though smartcards have the ability to generate RSA keypairs, this can be very slow. Typical times needed for a 1024 bit RSA keypair range from 8 seconds to 3 minutes. The larger times violate the ISO specifications for communications timeout so specialized hardware or software is sometimes necessary. Also, the quality of the keypairs may not be extremely high. The lack of computing power implies a relatively weak random number source as well as relatively weak algorithms for selecting large prime numbers.

The Digital Signature Algorithm (DSA) is less widely implemented than RSA. When it is implemented, it is typically found only at the 512 bit key length.

Smartcards support the ability to configure multiple PINs that can have different purposes. Applications can configure one PIN to be a "Security Officer" PIN, which can unblock the User PIN, after a set number of bad PIN attempts, or re-initialize the card. Other PINs can be configured to control access to sensitive files or purse functions.

DES and triple DES are commonly found in the leading smartcards. They usually have the option to be used in a Message Authentication Code (MAC) function. However, because the serial interface of a smartcard has a low bandwidth, bulk symmetric encryption is very slow.

So that it is difficult to extract information about the chip operating and file systems, various methods of hardware security monitoring are enabled on leading smartcards. A one-time, irreversible fuse typically disables any test code built into the EEPROM. In order to avoid card cloning an unalterable serial number is often burned into the memory. The cards are designed to reset themselves to a power-on state if they detect fluctuations in voltage, temperature, or clock frequency. Reading or Writing of the ROM is usually disabled. Because every vendor has their own, usually proprietary, schemes for these measures, it’s always good to inquire and/or request reports from independent testing laboratories.

Electronic purse functionalities are often present, but they are typically based on symmetric key technologies such as DES and triple DES. Thus, a shared secret key enforces the security of many of these schemes. Hashing algorithms commonly found include SHA-1 and MD-5; but again the low bandwidth serial connection hinders effective use of bulk hashing on the card.

Random number generation (RNG) varies among card vendors. Some implement a pseudo RNG where each card has a unique seed. In this case, random numbers cycle through, dependent on the algorithm and the seed. Some cards have a true, hardware based RNG using some physical aspect of the silicon. It’s best to check with the vendor for details of the RNG if it will be used in a cryptographically sensitive context.

Communications protocols on smartcards at the command level many times will have a security protocol built in. These are typically based on symmetric key technology and allow the smartcard itself to authenticate the read/write terminal or vice versa. However, the cryptograms and algorithms for these protocols are usually specific to a given application and terminal set.

## Data Transmissions

All communications to and from the smartcard are carried out over the C7 contact. Thus, only one party can communicate at a time, whether it is the card or the terminal. This is termed "half-duplex". Communication is always initiated by the terminal, which implies a type of client/server relationship between card and terminal.

After a card is inserted into a terminal, it is powered up by the terminal, executes a power-on-reset, and sends an Answer to Reset (ATR) to the terminal. The ATR is parsed, various parameters are extracted, and the terminal then submits the initial instruction to the card. The card generates a reply and sends it back to the terminal. The client/server relationship continues in this manner until processing is completed and the card is removed from the terminal.

The physical transmission layer is defined in ISO/IEC 7816-3. It defines the voltage level specifics which end up translating into the "0" and "1" bits.

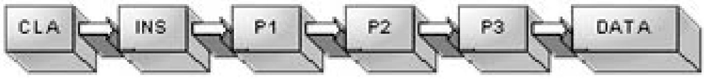
Logically, there are several different protocols for exchanging information in the client/server relationship. They are designated "T=" plus a number, and are summarized in Table 17-4.

Table: 17-4

| PROTOCOL | DESCRIPTION |
|---|---|
| T = 0 | Asynchronous, half-duplex, byte oriented, see ISO/IEC 7816-3 |
| T = 1 | Asynchronous, half-duplex, block oriented, see ISO/IEC 7816-3, Adm.1 |
| T = 2 | Asynchronous, full-duplex, block oriented, see ISO/IEC 10536-4 |
| T = 3 | Full duplex, not yet covered |
| T = 4 | Asynchronous, half-duplex, byte oriented, (expansion of T = 0) |
| T = 5 TO T = 13 | Reserved for future use |
| T = 14 | For national functions, no ISO standard |
| T = 15 | Reserved for future use |

The two protocols most commonly seen are T=0 and T=1, T=0 being the most popular. A brief overview of the T=0 protocol is given below. The references contain more detailed information and descriptions of all the protocols.

Figure 17-3: Typical T=0 instruction



In the T=0 protocol, the terminal initiates communications by sending a 5 byte instruction header which includes a class byte (CLA), an instruction byte (INS), and three parameter bytes (P1, P2, and P3). This is followed optionally by a data section. Most commands are either incoming or outgoing from the card's perspective and the P3 byte specifies the length of the data that will be incoming or outgoing. Error checking is handled exclusively by a parity bit appended to each transmitted byte. If the card correctly receives the 5 bytes, it will return a one-byte acknowledgment equivalent to the received INS byte. If the terminal is sending more data (incoming command) it will send the number of bytes it specified in P3. Now the card has received the complete instruction and can process it and generate a response. All commands have a two-byte response code, SW1 and SW2, which reports success or an error condition. If a successful command must return additional bytes, the number of bytes is specified in the SW2 byte. In this case, the GET RESPONSE command is used, which is itself a 5-byte instruction conforming to the protocol. In the GET RESPONSE instruction, P3 will be equal to the number of bytes specified in the previous SW2 byte. GET RESPONSE is an outgoing command from the card's point of view. The terminal and card communicate in this manner, using incoming or outgoing commands, until processing is complete.

## Instruction Sets

There are four international standards that define typical smartcard instruction sets. More than 50 instructions and their corresponding execution parameters are defined. Though found in four separate standards, the instructions are largely compatible. The specifications are GSM 11.11 (prETS 300608), EN 726-3, ISO/IEC 7816-4, and the preliminary CEN standard prEN 1546. Instructions can be classified by function as follows:

Table 17-5: Sample instruction types

| |
|---|
| File selection |
| File reading and writing |
| File searching |
| File operations |
| Identification |
| Authentication |
| Cryptographic functions |
| File management |

| |
|---|
| Instructions for electronic purses or credit cards |
| Operating system completion |
| Hardware testing |
| Special instructions for specific applications |
| Transmission protocol support |

Typically, a smartcard will implement only a subset of the possible instructions, specific to its application. This is due to memory or cost limitations.

## Smartcard Readers

Though commonly referred to as "smartcard readers", all smartcard enabled terminals, by definition, have the ability to read and write as long as the smartcard supports it and the proper access conditions have been fulfilled. In contrast to smartcards, which all have very similar construction, smartcard readers come in a variety of form factors with varying levels of mechanical and logical sophistication. Some examples include: reader integrated into a vending machine, handheld battery-operated reader with a small LCD screen, reader integrated into a GSM mobile phone, and a reader attached to a personal computer. Mechanically, readers have various options including: whether the user must insert/remove the card versus automated insertion/ejection mechanism, sliding contacts versus landing contacts, and provisions for displays and keystroke entry. Electrically, the reader must conform to the ISO/IEC 7816-3 standard.

The options for readers are numerous. This section will focus on readers attached to personal computer systems, because those have the largest impact on computer and network security. Many reader types are available off-the-shelf in today’s market, and each has its pros and cons. Table 17-6 lists some of these.

Table 17-6: Pros and Cons for various readers

| PHYSICAL CONNECTION | PROS | CONS |
|---|---|---|
| Serial Port | Very common; robust, inexpensive. Cross platform support for Windows, Mac, and Unix. | Many desktop computers have no free serial ports. Requires external power tap or battery. |
| PCMCIA | Excellent for travelling users with laptop computers | Can be slightly more expensive. Many desktop systems don't have PCMCIA slots. |
| PS/2 Keyboard Port | Easy to install with a wedge adapter. Supports protected PIN path. | Slower communication speeds. |
| Floppy | Very easy to install | Requires a battery. Communications speed can be an issue. |
| USB | Very high data transfer speeds. | Not yet widely available. Shared bus could pose a security issue. |
| Built-in | No need for hardware or software installation. | Not yet widely available. |

## Security Related Standards

Many of the standards thus far mentioned focus on the details of the smartcard, read/write terminal, and low-level software layers. Another important class of standards focuses on how smartcards are integrated into applications that provide computer and network security. This section discusses the principles of these standards, prominent standards, and the players that define and utilize them.

### Principles of Smartcard Security Standards

Any standard designed to facilitate the integration of smartcards into computer security systems should follow certain principles in order to be useful and gain acceptance. A few examples of these principles are found in Table 17-7.

Table 17-7:

| |
|---|
| **Multi-platform:**<br>Standard should be applicable to numerous modern day operating systems and computer architectures such as Windows, Unix, Mac, x86, Sparc, etc. |
| **Open participation:**<br>Standard should accept input and peer review from members of industry, academia, and government. |
| **Interoperability:**<br>Standard should be interoperable with other leading standards and protocols. |
| **Real, Functional:**<br>Standard should apply to real world problems and markets and adequately address. their requirements. |
| **Experience, Products:**<br>Standard should be created by a group of people with experience in security-related products and standards. |
| **Extensibility:**<br>Standard should facilitate expansion to new applications, protocols, and smartcard capabilities that weren't yet around when the standard was created. |

### Prominent Smartcard Specifications and Standards

The following are emerging as important standards with respect to the integration of smartcards into computer and network security applications:

- **PKCS#11: Cryptographic Token Interface Standard**
  This standard specifies an Application Programming Interface (API), called Cryptoki, to devices which hold cryptographic information and perform cryptographic functions. Cryptoki, pronounced crypto-key and short for cryptographic token interface, follows a simple object-based approach, addressing the goals of technology independence (any kind of device) and resource sharing (multiple applications accessing multiple devices). PKCS#11 presents to applications a common, logical view of the device called a cryptographic token. The standard was created in 1994 by RSA with input from industry, academia, and government.

- **PC/SC**
  The PC/SC Workgroup was formed in May 1997. It was created to address critical technical issues related to the integration of smartcards with the PC. PC/SC Workgroup members include Bull Personal Transaction Systems, Gemplus, Hewlett-Packard, IBM, Microsoft Corp., Schlumberger, Siemens-Nixdorf Inc., Sun Microsystems, Toshiba Corp., and VeriFone. The specification addresses limitations in existing standards that complicate integration of ICC devices with the PC and fail to adequately address interoperability, from a PC application

perspective, between products from multiple vendors. It provides standardize interfaces to Interface Devices (IFDs) and the specification of common PC programming interfaces and control mechanisms. Version 1.0 was released in December of 1997.

- **OpenCard**
  OpenCard is a standard framework announced by International Business Machines Corporation, Inc., Netscape, NCI, and Sun Microsystems Inc. that provides for interoperable smartcard solutions across many hardware and software platforms. The OpenCard Framework is an open standard providing an architecture and a set of APIs that enable application developers and service providers to build and deploy smartcard aware solutions in any OpenCard-compliant environment. It was first announced March, 1997.

- **JavaCard**
  The JavaCard API is a specification that enables the Write Once, Run Anywhere™ capabilities of Java on smartcards and other devices with limited memory. The JavaCard API was developed in conjunction with leading members of the smart card industry and has been adopted by over 95% of the manufacturers in the smart card industry, including Bull/CP8, Dallas Semiconductor, De La Rue, Geisecke & Devrient, Gemplus, Inside Technologies, Motorola, Oberthur, Schlumberger, and Toshiba.

- **Common Data Security Architecture**
  Developed by Intel, the Common Data Security Architecture (CDSA) provides an open, interoperable, extensible, and cross-platform software framework that makes computer platforms more secure for all applications including electronic commerce, communications, and digital content. The CDSA 2.0 specifications were adopted by The Open Group in December 1997.

- **Microsoft Cryptographic API**
  The Microsoft® Cryptographic API (CryptoAPI) provides services that enable application developers to add cryptography and certificate management functionality to their Win32® applications. Applications can use the functions in CryptoAPI without knowing anything about the underlying implementation, in much the same way that an application can use a graphics library without knowing anything about the particular graphics hardware configuration.

## Importance of Smartcards to Computer Security

### Importance of Smartcards as a Design Mechanism for Computer Networks

This section highlights the fundamental security challenges that face us in this increasingly computer network oriented world, and how smartcards can provide key advantages towards security.

### Fundamental Security Challenges

Because computers and networks are becoming so central to our lives in this digital age, many new security challenges are arising. This is the era of full connectivity, both electronically and physically. Smartcards can facilitate this connectivity and other value added capabilities, while providing the necessary security assurances not available through other means.

On the Internet, smartcards increase the security of the building blocks Authentication, Authorization, Privacy, Integrity, and Non-Repudiation. Primarily, this is because the private signing key never leaves the smartcard so it’s very difficult to gain knowledge of the private key through a compromise of the host computer system.

In a corporate enterprise system, multiple disjointed systems often have their security based on different technologies. Smartcards can bring these together by storing multiple certificates and passwords on the same card. Secure email and Intranet access, dial-up network access, encrypted files, digitally signed web forms, and building access are all improved by the smartcard.

In an Extranet situation, where one company would like to administer security to business partners and suppliers, smartcards can be distributed which allow access to certain corporate resources. The smartcard’s importance in this situation is evident because of the need for the strongest security possible when permitting anyone through the corporate firewall and proxy defenses. When distributing credentials by smartcard, a company can have a higher assurance that those credentials can not be shared, copied, or otherwise compromised.

**The Smartcard Security Advantage**

Some reasons why smartcards can enhance the security of modern day systems are:

- **PKI is better than passwords – smartcards enhance PKI**
  Public Key Infrastructure systems are more secure than password based systems because there is no shared knowledge of the secret. The private key need only be known in one place, rather than two or more. If the one place is on a smartcard, and the private key never leaves the smartcard, the crucial secret for the system is never in a situation where it is easily compromised. A smartcard allows for the private key to be usable and yet never appear on a network or in the host computer system.

  **Smartcards Increase the Security of Password Based Systems**
  Though smartcards have obvious advantages for PKI systems, they can also increase the security of password based systems. One of the biggest problems in typical password systems is that users write down their password and attach it to their monitor or keyboard. They also tend to choose weak passwords and share their passwords with other people. If a smartcard is used to store a user's multiple passwords, they need only remember the PIN to the smartcard in order to access all of the passwords. Additionally, if a security officer initializes the smartcard, very strong passwords can be chosen and stored on the smartcard. The end user need never even know the passwords, so that they can't be written down or shared with others.

- **Two Factor Authentication, and more**
  Security systems benefit from multiple factor authentication. Commonly used factors are: Something you know, Something you have, Something you are, and Something you do. Password based systems typically use only the first factor, Something you know. Smartcards add an additional factor, Something you have. Two factor authentication has proven to be much more effective than single because the "Something you know" factor is so easily compromised or shared. Smartcards can also be enhanced to include the remaining two features. Prototype designs are available which accept a thumbprint on the surface of the card in addition to the PIN in order to unlock the services of the card. Alternatively, a thumbprint template, retina template, or other biometric information can be stored on the card, only to be checked against data obtained from a separate biometric input device. Similarly, Something you do such as typing patterns, handwritten signature characteristics, or voice inflection templates can be stored on the card and be matched against data accepted from external input devices.

- **Portability of Keys and Certificates**
  Public key certificates and private keys can be utilized by web browsers and other popular software packages but they in some sense identify the workstation rather than the user. The key and certificate data is stored in a proprietary browser storage area and must be export/imported in order to be moved from one workstation to another. With smartcards the certificate and private key are portable, and can be used on multiple workstations, whether they are at work, at home, or on the road. If the lower level software layers support it, they can be used by different software programs from different vendors, on different platforms, such as Windows, Unix, and Mac.

- **Auto-disabling PINs Versus Dictionary Attacks**
  If a private key is stored in a browser storage file on a hard drive, it is typically protected by a password. This file can be "dictionary attacked" where commonly used passwords are attempted in a brute force manner until knowledge of the private key is obtained. On the other hand, a smartcard will typically lock itself up after some low number of consecutive bad PIN attempts, for example 10. Thus, the dictionary attack is no longer a feasible way to access the private key if it has been securely stored on a smartcard.

- **Non Repudiation**
  The ability to deny, after the fact, that your private key performed a digital signature is called

repudiation. If, however, your private signing key exists only on a single smartcard and only you know the PIN to that smartcard, it is very difficult for others to impersonate your digital signature by using your private key. Many digital signature systems require "hardware strength Non Repudiation", meaning that the private key is always protected within the security perimeter of a hardware token and can't be used without the knowledge of the proper PIN. Smartcards can provide hardware strength Non Repudiation.

- **Counting the Number of Private Key Usages**
  So many of the important things in our lives are authorized by our handwritten signature. Smartcard based digital signatures provide benefits over handwritten signatures because they are much more difficult to forge and they can enforce the integrity of the document through technologies such as hashing. Also, because the signature is based in a device that is actually a computer, many new benefits can be conceived of. For example, a smartcard could count the number of times that your private key was used, thus giving you an accurate measure of how many times you utilized your digital signature over a given period of time.

## Legalities

As with any technology, there are legal issues to keep in mind when dealing with smartcards. Commonly, a smartcard has the ability to perform certain licensed algorithms, such as the RSA asymmetric cipher. Usually any license fees associated with the algorithm are bundled into the cost of the smartcard.

If a smartcard can perform restricted technologies such as encryption at large keylengths, it is classified as munitions by certain US Commerce laws. As such, it can be considered illegal to export or import such an item in certain regions.

New digital signature laws are being written by many states that make it the end user's responsibility to protect their private key. If the private key can never leave an automatically PIN disabling smartcard, then the end user can find it easier to meet these responsibilities. Certificate authorities can help in this area by supporting certificate extensions that specify the private key was generated in a secure environment and has never left the confines of a smartcard. With this mechanism, higher levels of non-repudiation can be achieved when verifying a smartcard based signature while using a certificate containing such an extension. In other words, a digital signature carries more weight if its associated certificate validates that the private key resides on a smartcard and can never be extracted.

## Smartcard Enabled Products

This section lists popular security products and explains how smartcards can be used to enhance their security.

- **Web Browsers (SSL, TLS)**
  Web browsers use technology such as Secure Sockets Layer (SSL) and Transport Layer Security (TLS) to provide security while browsing the World Wide Web. These technologies can authenticate the client and/or server to each other and also provide an encrypted channel for any message traffic or file transfer. The authentication is enhanced because the private key is stored securely on the smartcard. The encrypted channel typically uses a symmetric cipher where the encryption is performed in the host computer because of the low data transfer speeds to and from the smartcard. Nonetheless, the randomly generated session key that is used for symmetric encryption is wrapped with the partner's public key, meaning that it can only be unwrapped on the smartcard. Thus it is very difficult for an eavesdropper to gain knowledge of the session key and message traffic.

- **Secure Email (S/MIME, OpenPGP)**
  S/MIME and OpenPGP allow for email to be encrypted and/or digitally signed. As with SSL, smartcards enhance the security of these operations by protecting the secrecy of the private key and also unwrapping session keys within a security perimeter.

- **Form Signing**
  Web based HTML forms can be digitally signed by your private key. This could prove to be a

very important technology for internet based business because it allows for digital documents to be hosted by web servers and accessed by web browsers in a paperless fashion. Online expense reports, W-4 forms, purchase requests, and group insurance forms are some examples. For form signing, smartcards provide portability of the private key and certificate as well as hardware strength non repudiation.

- **Object Signing**
  If an organization writes code that can be downloaded over the web and then executed on client computers, it is best to sign that code so the clients can be sure it indeed came from a reputable source. Smartcards can be used by the signing organization so the private key can't be compromised by a rogue organization in order to impersonate the valid one.

- **Kiosk / Portable Preferences**
  Certain applications operate best in a "kiosk mode" where one computer is shared by a number of users but becomes configured to their preferences when they insert their smartcard. The station can then be used for secure email, web browsing, etc. and the private key would never leave the smartcard into the environment of the kiosk computer. The kiosk can even be configured to accept no mouse or keyboard input until an authorized user inserts the proper smartcard and supplies the proper PIN.

- **File Encryption**
  Even though the 9600 baud serial interface of the smartcard usually prevents it from being a convenient mechanism for bulk file encryption, it can enhance the security of this function. If a different, random session key is used for each file to be encrypted, the bulk encryption can be performed in the host computer system at fast speeds and the session key can then be wrapped by the smartcard. Then, the only way to easily decrypt the file is by possessing the proper smartcard and submitting the proper PIN so that the session key can be unwrapped.

- **Workstation Logon**
  Logon credentials can be securely stored on a smartcard. The normal login mechanism of the workstation, which usually prompts for a username and password, can be replaced with one that communicates to the smartcard.

- **Dialup Access (RAS, PPTP, RADIUS, TACACS)**
  Many of the common remote access dial-up protocols use passwords as their security mechanism. As previously discussed, smartcards enhance the security of passwords. Also, as many of these protocols evolve to support public key based systems, smartcards can be used to increase the security and portability of the private key and certificate.

- **Payment Protocols (SET)**
  The Secure Electronic Transactions (SET) protocol allows for credit card data to be transferred securely between customer, merchant, and issuer. Because SET relies on public key technology, smartcards are a good choice for storage of the certificate and private key.

- **Digital Cash**
  Smartcards can implement protocols whereby digital cash can be carried around on a smartcard. In these systems, the underlying keys that secure the architecture never leave the security perimeter of hardware devices. Mondex, VisaCash, EMV ( Europay-Mastercard-Visa ), and Proton are examples of digital cash protocols designed for use with smartcards.

- **Building Access**
  Even though the insertion, processing time, and removal of a standard smartcard could be a hassle when entering a building, magnetic stripe or proximity chip technology can be added to smartcards so that a single token provides computer security and physical access.

## Problems with Smartcards

Even though smartcards provide many obvious benefits to computer security, they still haven't caught on with great popularity in countries like the United States. This is not only because of the prevalence, infrastructure, and acceptability of magnetic stripe cards, but also because of a few problems associated with smartcards. Lack of a standard infrastructure for smartcard reader/writers is often cited as a complaint. The major computer manufactures haven’t until very recently given much thought to offering a smartcard reader as a standard component. Many companies don’t want to absorb the cost of outfitting computers with smartcard readers until the economies of scale drive down their cost. In the meantime, many vendors provide bundled solutions to outfit any personal computer with smartcard capabilities.

Lack of widely adopted smartcard standards is often cited as a complaint. The number of smartcard related standards is high and many of them address only a certain vertical market or only a certain layer of communications. This problem is lessening recently as web browsers and other mainstream applications are including smartcards as an option. Applications like these are helping to speed up the evolution of standards.

## Attacking Smartcards

Attacks on smartcards generally fall into four categories.

- **Logical attacks**
  Logical attacks occur when a smartcard is operating under normal physical conditions, but sensitive information is gained by examining the bytes going to and from the smartcard. One example is the so-called "timing attack" described by Paul Kocher. In this attack, various byte patterns are sent to the card to be signed by the private key. Information such as the time required to perform the operation and the number of zeroes and ones in the input bytes are used to eventually obtain the private key. There are logical countermeasures to this attack but not all smartcard manufacturers have implemented them. This attack does require that the PIN to the card be known, so that many private key operations can be performed on chosen input bytes.

- **Physical attacks**
  Physical attacks occur when normal physical conditions, such as temperature, clock frequency, voltage, etc, are altered in order to gain access to sensitive information on the smartcard. Most smartcard operating systems write sensitive data to the EEPROM area in a proprietary, encrypted manner so that it is difficult to obtain cleartext keys by directly hacking into the EEPROM. Other physical attacks that have proven to be successful involve an intense physical fluctuation at the precise time and location where the PIN verification takes place. Thus, sensitive card functions can be performed even though the PIN is unknown. This type of attack can be combined with the logical attack mentioned above in order to gain knowledge of the private key. Most physical attacks require special equipment.

- **Trojan Horse attacks**
  This attack involves a rogue, Trojan horse application that has been planted on an unsuspecting user’s workstation. The Trojan horse waits until the user submits a valid PIN from a trusted application, thus enabling usage of the private key, and then asks the smartcard to digitally sign some rogue data. The operation completes but the user never knows that their private key was just used against their will. The countermeasure to prevent this attack is to use a "single-access device driver" architecture. With this type of architecture, the operating system enforces that only one application can have access to the serial device (and thus the smartcard) at any given time. This prevents the attack but also lessens the convenience of the smartcard because multiple applications can not use the services of the card at the same time. Another way to prevent the attack is by using a smartcard that enforces a "one private key usage per PIN entry" policy model. In this model, the user must enter their PIN every single time the private key is to be used and therefore the Trojan horse would not have access to the key.

- **Social Engineering attacks**
  In computer security systems, this type of attack is usually the most successful, especially when the security technology is properly implemented and configured. Usually, these attacks rely on the faults in human beings. An example of a social engineering attack has a hacker impersonating a network service technician. The serviceman approaches a low-level employee and requests their password for network servicing purposes. With smartcards, this type of attack is a bit more difficult. Most people would not trust an impersonator wishing to have

their smartcard and PIN for service purposes.

Any security system, including smartcards, is breakable. However, there is usually an estimate for the cost required to break the system, which should be much greater than the value of the data being protected by the system. Independent security labs test for common security attacks on leading smartcards, and can usually provide an estimate of the cost in equipment and expertise of breaking the smartcard. When choosing a smartcard for an architecture, one can ask the manufacturer for references to independent labs that have done security testing. Using this information, designers can strive to ensure that the cost of breaking the system would be much greater than the value of any information obtained.

## Conclusion

We have focused on the state of the art for smartcards and their use in computer and network security systems. Smartcards have proven to be useful for transaction, authorization, and identification media. As their capabilities grow, they could become the ultimate thin client, eventually replacing all of the things we carry around in our wallets, including credit cards, licenses, cash, and even family photographs. By containing various identification certificates, smartcards could be used to voluntarily identify attributes of ourselves no matter where we are or to which computer network we are attached.

Current state of the art smartcards have sufficient cryptographic capabilities to support popular security applications and protocols.

## Bibliography

ISO 7810: 1995, Identification cards – Physical characteristics

ISO 7811: 1995, Identification cards – Recording technique

ISO 7813: 1995, Identification cards – Financial transaction cards

ISO 7816: 1987-1995, Identification cards – Integrated circuit(s) cards with contacts

ISO/IEC 10373: 1993, Identification cards – Test methods

ISO/IEC 11693: 1995, Optical memory cards

ISO/IEC 11694: 1995, Optical memory cards – Linear recording method

GSM 11.11: 1995, European digital cellular telecommunications system

EN 726-3: 1994, Identification card systems – Telecommunications integrated circuit(s) card and terminals

prEN 1546: 1995, Identification card systems – Inter-sector electronic purse

Renkl, W. And Effing, W.: Smartcard Handbook, Chichester: John Wiley & Sons, 1997

PC/SC: http://www.smartcardsys.com/

PKCS#11: http://www.rsa.com/rsalabs/pubs/PKCS/html/pkcs-11.html

Opencard: http://www.opencard.org/

Javacard: http://www.javasoft.com/products/javacard/index.html

CDSA: http://developer.intel.com/ial/security/

Microsoft CryptoAPI: http://www.microsoft.com/workshop/security/default.asp

Paul Kocher: Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems: http://www.cryptography.com/timingattack/paper.html

Mondex: http://www.mondexusa.com/

Proton: http://www.protonworld.com/

VisCash: http://www.visa.com/cgi-bin/vee/nt/chip/main.html?2+0

**SSP Solutions, Inc.**
17861 Cartwright Road
Irvine, CA 92614
(949) 851-1085
http://www.sspsolutions.com/

**SSP-Litronic Division**
11490 Commerce Park Drive, Suite #520
Reston, VA 20191-1557
(703) 905-9700
sales@sspsolutions.com