WHC-SA-2307-FP

Authentication Techniques for Smart Cards

R. A. Nelson

Date Published
February 1994

To Be Presented at
CardTech SecurTech '94
Arlington, Virginia
April 10-13, 1994

Prepared for the U.S. Department of Energy
Office of Environmental Restoration and
Waste Management

Westinghouse
Hanford Company
P.O. Box 1970
Richland, Washington 99352

Hanford Operations and Engineering Contractor for the
U.S. Department of Energy under Contract DE-AC06-87RL10930

Approved for Public Release

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

AUTHENTICATION TECHNIQUES FOR SMART CARDS

R.A. (Rob) Nelson

Abstract

Smart card systems are most cost efficient when implemented as a distributed system. Which is a system without central host interaction or a local database of card numbers for verifying transaction approval. A distributed system, as such, presents special card and user authentication problems. Fortunately, smart cards offer processing capabilities that provide solutions to authentication problems, provided the system is designed with proper data integrity measures.

Smart card systems maintain data integrity through a security design that controls data sources and limits data changes. A good security design is usually a result of a system analysis that provides a thorough understanding of the application needs. Once designers understand the application, they may specify authentication techniques that mitigate the risk of system compromise or failure. Current authentication techniques include cryptography, passwords, challenge/response protocols, and biometrics. The security design includes these techniques to help prevent counterfeit cards, unauthorized use, or information compromise.

This paper discusses card authentication and user identity techniques that enhance security for microprocessor card systems. It also describes the analysis process used for determining proper authentication techniques for a system.

Introduction

Smart card systems allow a distributed transaction network, without physical connections between network terminals. The smart card is the data distribution tool that provides the information used in transactions. The terminal, or card acceptor device (CAD) processes the smart card supplied information based on business procedures for its application. A smart card system is most cost-efficient when the CAD may verify transaction approval without querying a host computer's databases of valid cards. To ensure system-wide data integrity, the terminal must have a means for validating data.

The terminal validates information by requiring the smart card to prove system membership before it accepts data from the card. The key is that the smart card proves its membership, that is it uses its processing capabilities. Several methods prove smart card membership, in smart card systems, These methods are generally called smart card authentication and include: passwords, cryptography, and challenge response protocols.

The password authentication method mimics computer log on, which is its primary application. The card reveals its identifier during each authentication attempt. This method is effective only if the password changes for each authentication attempt; otherwise, it would be too easy to fake the authentication. Dynamic password authentication is the best approach.

Cryptography[1] is an effective way to prove system membership without revealing the identifying characteristics of the cards to the outside world. But cryptographic methods require key distribution, which increases system complexity, adds extra system administration, and reduces flexibility.

Zero-knowledge protocols provide smart card authentication without passwords and encryption keys[2]. However, they require sophisticated microprocessors that increase smart card costs and reduce the usable memory space.

Smart card authentication is only half of the system integrity battle. Smart card systems allow updates to the card with information supplied by the terminal. Smart cards must authenticate the terminals before accepting data. This prevents the smart card from supplying bogus information to terminals during future transactions. Smart cards use their processing capability with built-in security features and will only accept changes to data after security requirements are met.

Each authentication method has tradeoffs among costs, operating efficiency, and benefits. The authentication methods are judged by several factors: ease of implementation, sophistication of the smart card microprocessor required, system management effort (e.g., secret key distribution), vulnerability of the system to compromise, and time to complete the authentication. The first three are cost-driven factors, the fourth deals with data integrity, and the last with customer satisfaction. The selection of a satisfactory authentication method is based on the individual application's requirements, which are identified through a system analysis process. System analysis is the key to selecting the proper authentication method and ultimately a cost-efficient smart card system.

Dynamic Password Authentication

The dynamic password method improves traditional password approaches by using the processing capability of smart cards for creating a different password for each authentication attempt. The smart card generates new passcodes many times a day. The host executes the same algorithm as the smart card, so it knows the password token's current valid password at any given time.

The card issuer initializes each card in the system with a synchronization process that loads an initialization code, or seed, into both the password token and host. The seed and the algorithm for determining the passwords are kept secret. The seed value and initialization code for each card are unique such that no two cards should have the same password at a given time. It is unlikely that anyone could predict the valid password at any given time without knowing the algorithm, seed, and initialization value.

During authentication, the password token displays the current password, which is transmitted to the host. The verifier compares the password received to the expected value. The host accepts the card if the identifiers match. This method ensures card authenticity, because the lifetime of each password is short and the algorithm is variable with each card and kept secret.

This approach performs authentication without using a CAD. Instead, the user enters information (i.e., card identity number and password) into a computer terminal allowing remote log in. Smart cards used for this authentication method require a battery, a display, and sometimes a keypad.

Symmetric Key Authentication

The Digital Encryption Standard (DES) [4] algorithm is a symmetric key cryptography method commonly used smart card systems. This method uses a stored, secret cryptographic key and the public DES algorithm in each smart card and CAD. The steps in symmetric key authentication are listed below.

1) The smart card sends the microprocessor serial number (I) to the CAD, which combines the number with the master key (M_k) to form the smart card's diversified key (K). The issuer loads a diversified key into each smart card during card initialization.

2) The CAD generates a random number (R), then encrypts R to form the value Y that is transmitted as the challenge to the smart card.

3) The smart card decrypts Y, forming the response (X) and returns X to the CAD.

4) The CAD compares R and X, accepting the card if the two values match.

The Telepass 1 algorithm is one-way algorithm used for smart card authentication. This algorithm uses, a diversified secret key, the contents of a specific word in the smart card memory, and a random external value to compute the response to an authentication challenge. The Telepass 1 algorithm provides functionality for data secrecy in key distribution and data integrity through message authentication codes.

In a symmetric key system, the secret key in each smart card should be unique so that discovery of the key does not compromise the entire system. This key diversification provides a unique cryptographic key for each smart card during personalization. The system builds the diversified key from the combination of a system master key and a unique card characteristic (i.e., the microprocessor serial number). Using diversified keys in the smart cards minimizes the vulnerability of the keys being discovered. However, the weakness exists in the master keys stored in the CAD.

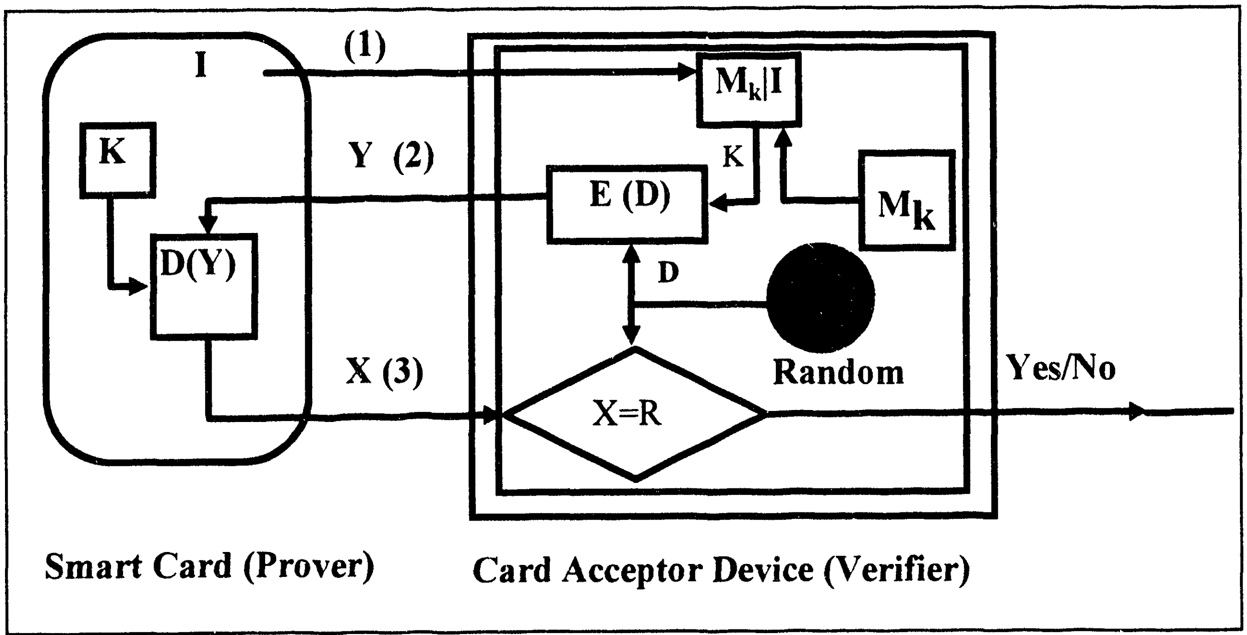

Figure 1. Authentication Using Symmetric Key Cryptography.

The CAD contain a copy of the master key so they can manufacture the derived key for each smart card. Compromise of a diversified key has little effect on the system, provided the compromise is discovered and the card can be removed from the system. The compromise of a master key can have serious consequences. The compromise of a master key requires that all cards must be loaded with new keys, which is costly to the systems and inconvenient for users. To avoid key compromise, the CAD may contain a security module with protected memory (i.e., smart card), where the master key is stored.

Asymmetric Key Authentication

Asymmetric key authentication is a cryptographic technique that uses a different verification key in the CAD than the proving key used by the smart card[5][6]. This method is generally implemented using *trapdoor one-way functions*[7], in which smart card generates an electronic signature with its secret key and the CAD uses a public key to authenticate the signature.

The Rivest-Shamir-Adelman (RSA) public key cryptosystem is the most commonly implemented asymmetric key authentication method. It places the security of the algorithm in the difficulty of factoring large prime numbers. The mathematics behind this method, based on the Euler totient function, are explained in references [1] and [3].

Authentication using the asymmetric key with a trapdoor function is described in the following steps.

1) The CAD transmits a random number (X) to the smart card.

2) The smart card transmits its identification word (I) and the random number encrypted (Y) with the secret key (k) in the smart card. It also provides its public key (n) as a *certificate* formed with n and I. The certificate provides a way for the CAD to check the validity of the public key.

3) The CAD verifies the cards response by deciphering Y (X') and comparing it to the original random number.

Implementation of RSA requires an exponentiation module for computing the electronic signature, a relatively large random access memory (RAM) for storing intermediate values, larger program memory for storing the additional instructions required by the algorithm, and more time for computation. Currently, cards that have implemented asymmetric key cryptography are sold at a higher cost than general purpose cards, but are also able to compute electronic signatures for documents.

The Digital Signature Standard (DSS) developed by the National Institute of Standards and Testing (NIST) may be applied to smart card authentication. This method was developed for generating document signatures in the U.S. Government and may be a preferred choice for government application authentication. Its strength lies in the difficulty in computing discrete logarithms. It has an advantage that some of the intermediate values for an authentication can be "pre-computed" so that the authentication time can be reduced.

The crux of public key systems, especially those that are used for authentication purposes, is key distribution. If each CAD functions totally independently, will it have access to the verification keys for all possible cards? The certificate described above allows the smart cards to provide the verification key to the CAD. The certificate is formed when the card is initialized using a system wide secret key. The CAD are loaded with the system wide verification key that is used to authenticate the certificate supplied by the smart card. When the card supplies the certificate, the CAD deciphers the certificate into the smart card's verification key and the card identity number. The identity number is used as a check value to ensure the public key's authenticity.

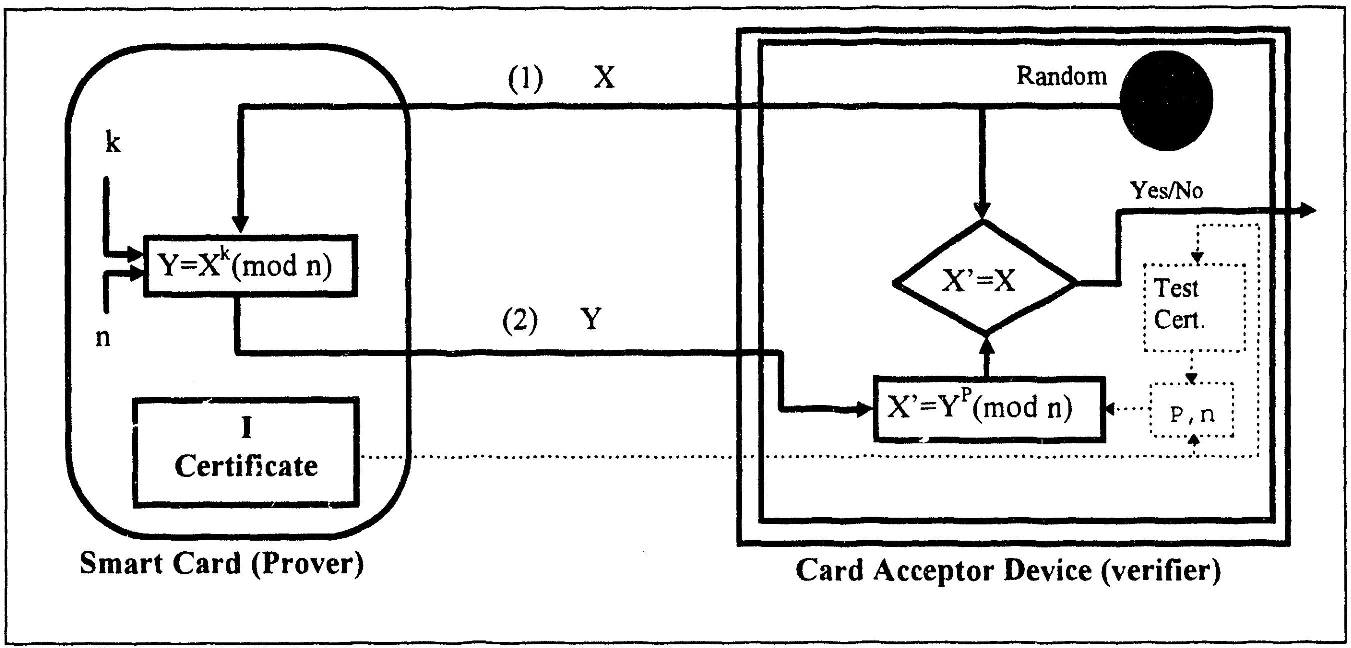

Figure 2. Public Key Authentication Using the Rivest Shamir Adelman Cryptosystem

Zero-Knowledge Authentication

Zero-knowledge authentication is another challenge/response protocol, however it does not use cryptographic methods for authenticating smart cards. Unlike the previous methods, the CAD contains no passwords or keys to compare against the information stored in the smart card. The CAD can deduce that the smart card possesses the secret accreditation without possessing any part of the accreditation. This is accomplished by the verifier issuing one or more challenges and the prover responding with an equal number of responses. Regardless of the number of challenge response pairs between the smart card and the CAD, the smart card reveals no evidence as to the value of the secret accreditation.

The first practical zero-knowledge approach was developed by A. Fiat and A. Shamir (Fiat Shamir) [8]. In this method, the card issuer computes a public system constant from the product of two large (i.e., 256 bits) prime numbers. A set of (k) secret accreditation values, formed using a hash function with the card identification word, is loaded into each card. The accreditation values and public system constant are used during the card authentication process, which is summarized below:

1) The smart card sends its identification word to the CAD. The CAD computes calculates k verification values (v) that are formed by executing the hash function on the identification word.

2) The smart card generates a random number (r) and sends its square to the CAD. This is the initial witness

3) The CAD generates its own random number, of length k, which represents a bit vector (binary word) that is used to randomly select the secret accreditation values. This value is sent to the smart card as a challenge.

4) The smart card computes a response by forming a product between the secret accreditation values that were represented by the bit vector sent by the CAD and the smart card's random number, r.

5) The CAD evaluates the response by squaring it and multiplying it by all the values, v, in which the bit vector had a value of one. The smart card is accepted if the result matches the value r received from the card in step 2.

The security level of this method increases exponentially with the product of the number of challenge/response pairs (repeating steps 2 through 5) and the number of accreditation values. However, increasing the number of transactions and accreditation values is expensive in time and memory. A second approach was developed that minimizes the transactions and accreditation values.

The Guillou-Quisquater method[9] accomplishes the same level of security with fewer accreditation values and transmissions; however, the number of computations is increased. Each smart card carries an identity that consists of a name (e.g., microprocessor serial number) and validity period. This identity is formed into a representative by the verifier. The smart card sends a witness, generated from a random number, to the CAD. The smart card also combines the random number with the challenge received from the CAD to form a response. The CAD combines the representative and the smart card's response to create a verification value that should match the witness sent by the card. Identity is verified if the two values match.

The security of this scheme relies on the truth of three statements to ensure that a counterfeit smart card cannot be used in the system:

- The challenge must be random and independent of the initial witness
- The initial witness must be constructed and accepted by the verifier before generating and transmitting the response
- The response for any two distinct challenges must also require two distinct initial witnesses.

Challenges must be random and independent to preclude the impostor from recording and re-transmitting a response to a certain challenge. The initial witness must be accepted before the response to prevent an impostor using information supplied by the system to mimic an authentic card. The last item states that the proving entity must respond to only a single challenge with each initial witness. If a smart card could respond to two challenges with a single witness, an identity could be found that would make computing of the secret accreditation value possible.

The Guillou-Quisquater authentication method steps are as follows:

1) The card sends the identification word (I) to the CAD and generates a random number (R). The initial witness (T) is computed by raising R to the system constant (V) modulus the system wide constant (n). The certificate (J) is formed by concatenating J with itself.

2) The CAD generated a random number (d) and transmits it to the smart card.

3) The smart card raises its secret value (B) to the d^{th} power, then multiplies the result by the random value R. The smart card returns the modulus n of this computation (D) to the CAD.

4) The CAD raises J to the d^{th} power, and D to the V^{th} power and multiplies the results modulus n giving a result (T'). The CAD accepts the card if the value T equals T'.

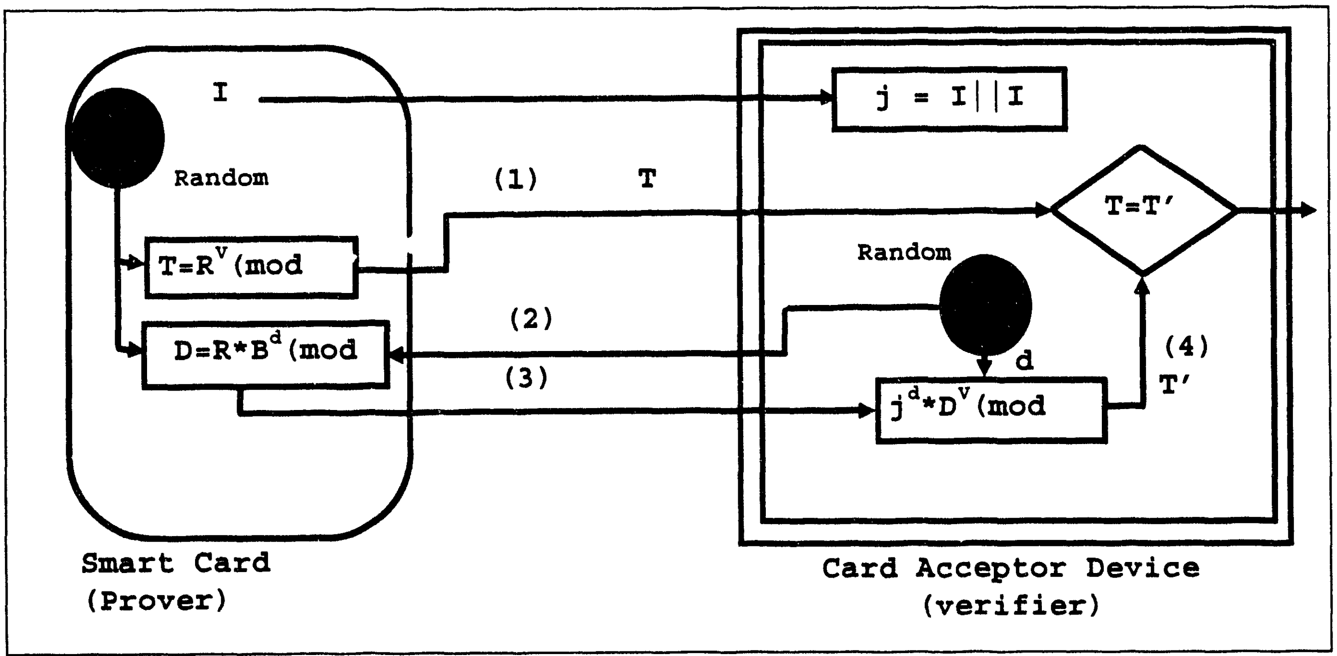

Figure 3. Zero Knowledge Authentication with the Guillou-Quisquater Method.

Zero-knowledge techniques are less vulnerable to key compromise than cryptographic methods. The vulnerabilities in the cryptographic methods lie in the key management aspects. Zero knowledge requires that the smart cards are personalized at a single location, but once personalized, the secret accreditation values remain in the cards. These techniques require smart cards with random number generators and exponentiation units. This requirement increases the costs of the smart cards. However, zero knowledge requires no key management, so the savings and increased security may offset the costs of smart cards.

User Authentication

While some applications accept card possession as proof of authorized users, other applications require assurance that only authorized users have card access. User authentication associates a smart card to a specific user by employing a personal identification number (PIN) or biometric trait. It is required for many applications.

The PIN serves as a password that is intended to be known only by the card holder and is the card holder's identity to the smart card. When the user enters a PIN on a keypad in the CAD, the CAD transmits the PIN to the smart card. The card compares the PIN to a reference number stored in its memory. If the values match, the user identity is verified. When the card identifies the user, it may open access to certain files or signal the CAD of the match.

Recording the PIN presents a threat of system compromise. To avoid compromise, the CAD often encrypts the PIN before presentation to the smart card. Enciphered PIN presentation is effective only if the CAD uses an encryption key that is valid for a single session.

Of course, PINs are subject to compromise through social engineering, "Hey, what's your password?". Some applications increase security by using biometric identity verification to authenticate users. Biometrics are reserved for applications with a large security risk mainly because of the cost of biometric devices and acceptance by the users.

Biometric verification can be handled in two ways: comparison on the biometric device or comparison in the smart card. When the biometric is compared in the biometric device, the card is signaled with the outcome of the comparison. Comparison on the card means that the biometric data is sent to the card for comparison with reference data stored in the smart card. When the data match, the card may grant access to files and signal the CAD that user identity was accepted. In either case, CAD authentication is necessary to prevent spoofing the smart card into accepting false data as valid.

Armed with these tools for authenticating cards, CAD, and users, the question, "which is the best authentication method for an application?". The answer in selecting the "best" method comes from a thorough system analysis, which is the subject of the next section.

System Analysis

A new automated system replaces a manual system or provides a technological improvement of an existing automated system. Sometimes the system replacement provides cost savings in the way the organization performs business; other times the system fails to perform up to expectations. Most cases of failure result from an inadequate understanding of the business process, the objectives, and the requirements or from a poor technology selection.

An effective system meets users' expectations; provides data and functional integrity; maintains system and information availability; stays within development, maintenance, and operational budgets; and guarantees information confidentiality.

System analysis is a key element in the design of an effective system. Analysis provides an understanding of the issues that the system must address. It should also identify the consequences of system failure or system compromise. Systems analysts determine system objectives, requirements, and threats against the system. System designers choose an appropriate technology and plan the architecture, procedures, and contingencies that meet the system objectives and requirements all within a limited budget. A thorough system analysis should result in a technology selection that is appropriate for the problem or business objectives of the organization.

Smart card technology is an effective architecture for a certain set of problems. These problems are characterized by the requirement for data distribution and off-line processing with a high level of data integrity. Many smart card systems meet these needs in a cost-effective manner. Smart card technology should be chosen only after the system analysis indicates it as the best solution to the system requirements. The analysis should consider the costs verses benefits for defining an architecture that best meets the system's needs.

An Approach To Authentication

A sample application is described in which the customer's demands had a profound outcome on the system specification. There were several tradeoffs between costs, efficiencies and factors such as Government policy and user expectations. The customer requirements were as follows:

- Low cost for cards
- Multiple application cards
- Easy for multiple sites to implement
- High security
- Cards meet established standards
- Cryptographic methods conceptually simple and meeting U.S. Government standards.

The low initial card costs were given higher priority than the reduced administrative costs for public key cryptographic and zero knowledge protocols. Microprocessors containing exponentiation cells, or code to execute the public key mathematical functions are too expensive for this application. .The customer was not willing to accept zero knowledge protocols as a secure method for authentication. We chose authentication methods that best fit these requirements.

The card authentication method uses the DES algorithm symmetric key approach discussed earlier. Diversified authentication keys were used in the smart cards to reduce system vulnerabilities. The CAD randomly selected keys with two sets of secret keys in the card to reduce the chances of system compromise. The second set of secret keys are activated if the first set was compromised.

The CAD authentication used an encrypted presentation of the password exchange. The card accepts the terminal if the password matches the stored password.

The user authentication method used depends on requirements of a particular application. Our cards are used for different applications, each having different requirements for user authentication. The application needs ranged from no user authentication to positive user authentication.

User authentication is accomplished with the PIN comparison functions offered by most cards, or through a hand geometry biometric comparison. PIN-based user authentication is appropriate for access to files stored on the smart card and for proving user identity. The PIN comparison in the card is performed by selecting a file, presenting the appropriate PIN, then checking the result. If the operation succeeded, the PIN was valid. This check can be performed without revealing the value of the PIN stored in the smart card, reducing the chance of compromise.

A separate PIN, not associated with a file access control, is used for proving identity of the user to other devices. This PIN is supplied to the CAD on request and used to activate another device after card authentication is performed. The separate PIN does not match any PIN used in the card for access to files, because these PINs should not be revealed outside the smart card.

This system uses the processing capability of smart cards for user authentication by actually executing a biometric template comparison in the microprocessor on the smart card. On user identification, the biometric device collects the hand template and sends it to the card. The card compares the template with a reference template and returns the result to the CAD. The CAD must successfully complete authentication or the smart card will refuse the comparison command. The card also updates the template on successful comparison.

The vulnerability in this approach lies in the possible compromise of the secret keys from a CAD. The master key used to initialize the smart cards resides in a security module in the CAD. A compromised master key means that the CAD must receive new verification keys.

Conclusion

The smart card authentication method used depends on the system requirements and constraints. The costs associated with key management must be weighed against the costs of higher performance smart cards. The security level needed must be compared against the convenience factor for the user. The answers to proper authentication technology selection result from a thorough understanding of the system requirements.

Symmetric Key Cryptography

Telepass 1

Data Encryption Standard (DES)

Asymmetric Key Cryptography

Digital Signature Standard (DSS)

Rivest Shamir Adelman (RSA)

Zero Knowledge

Fiat-Shamir

Guillou-Quisquater

Schorr

Figure 4. Authentication Methods For Smart Cards

Glossary

Asymmetric Key Authentication — A cryptographic method in which the key in the prover does not match the key in the verifier (e.g., RSA, DSS). The prover generates an electronic signature with a secret key. The verifier accepts the prover's identity if it can verify the electronic signature with a publicly distributed key

Biometric — A measurement of a physical or behavioral trait used to prove personal identity to an order of statistical significance.

Certificate — A code that can be verified by a receiving entity. Generally, the code is a value, encrypted by a central authority's private key that is verified by decrypting the code with the authority's public key.

Challenge-Response Protocol — A process where a verifier sends a value (challenge) to the prover, expecting a certain returned value, or expecting information that can be used to verify authenticity of the prover.

Hash Function — Computes a fixed length code (hash) from a message of arbitrary length. The properties of a good Hash function are: easy to compute for any message, generates a unique value for any message, and must be a one-way function.

Key-Management — Administrative process of controlling, distributing, and protecting cryptographic keys.

One-way Function — Given current computing capability, it is not feasible to find an inverse of a function f such that given a result y computed from f acting on another value, x.

Prover — an entity that is proving its membership in the system.

Symmetric Key Authentication — a cryptographic method where the key used for verifying is identical to the key used to generate the proof (e.g., DES). The verifier accepts the identity of the prover if it can decrypt a random number, encrypted with the secret key of the verifier.

Trap Door Function — for a family of cryptographic functions, there exists a pair $\{E,D|E\neq D\}$ of functions for each member in the family such that, a value x encrypted by E, known as y, can be decrypted by D (i.e., $y = E(x)$ & $x = D(y)$). These functions must also be one-way functions.

Verifier — An entity that determines the authenticity of the prover.

Zero Knowledge Authentication — A challenge-response protocol where the verifier deduces that the prover holds the secret identifying information, without having any knowledge to the content of the identifying information.

Bibliography

[1] **The Science of Information Integrity**, Simmons, Ed.; pp. vii-xv; IEEE Press; New York, N.Y.; 1992

[2] Guillou, L.C., et al; "The Smart Card: A Standardized Device Dedicated to Public Cryptology," **The Science of Information Integrity**, Simmons, Ed.; pp. 561-613; IEEE Press; New York, N.Y.; 1992

[3] Miller, Ben; "An Overview of Enhanced User Authentication," Proceedings of CardTech; Arlington, Va. April 1993, pg. 847.

[4] **Federal Information Processing Standard Publication (FIPS PUB) 140-1;** National Institute of Standards and Testing.

[5] Nechvatal, J. "Public Key Cryptography," **The Science of Information Integrity**, Simmons, Ed.; pp. 561-613; IEEE Press; New York, N.Y.; 1992.

[6] Difie, W; "The First Ten Years of Public Key Cryptography,' **The Science of Information Integrity**, Simmons, Ed.; pp. 561-613; IEEE Press; New York, N.Y.; 1992.

[7] Koenigs, Hans-Peter; "Cryptographic Identification Methods for Smart Cards in the Process of Standardization," **IEEE Communications Magazine**; June 1991.

[8] Fiat, A. And Shamir, A.; "Unforgable Proofs of Identity," in **Proceedings, SECURICOM '87: 5th Worldwide Congres on Computer Communications Security and Protection**, Paris, France, March 4-6, 1987, pp. 147-153.

[9] Guillou, L.C. and Quisquarter, J.-J.; "A Practical Zero-Knowledge Protocol Fitted to Security Microproccessor Minimizing both Transmission and Memory," in **Lecture Notes in Computer Science 330; Advances in Cryptology: Proceedings Eurocrypt '88**; C.G. Guenter, Ed.; Davos, Switzerland, May 25-27, 1988, pp 123-128. Berlin: Springer-Verlag, 1988.

DISTRIBUTION

Number of copies

OFFSITE

CardTech
SecurTech
P. O. Box 5758
Bethesda, Maryland 20824-5758

CardTech/Secure
11619 Danville Drive
Rockville, Maryland 20855

ONSITE

U.S. Department of Energy-
Richland Operations Office

D. O. Baker	A6-35

Westinghouse Hanford Company

R. A. Nelson (6)	L6-09
Information Release Administration (3)	L8-07

DATE
FILMED

5/20/94

END