SYSTEM AND METHOD FOR FAST SMART CARD TRANSACTIONS

BACKGROUND OF THE INVENTION

The present invention relates generally to payment systems. More particularly, embodiments of the invention relate to payment systems using smart cards.

Portable transaction cards with processing power, or so-called "smart cards" are being developed in the size of conventional plastic credit cards. These smart cards can include an embedded computer chip having processing power and memory. Variants of these cards were described in the early 1970s by, for example, Moreno in U.S. Patent No. 4,007,355. However, the cards are only recently becoming widely available for payment applications around the world.

The International Standards Organization (ISO) has established a number of specifications for such electronic cards under, for example, the ISO/IEC Standards Series 7810, 7811, 7816, 10536 and 14443. A typical smart card includes a plurality of spaced apart electrical contacts on one surface which provide input and output interfaces with the computer processor and memory in the card. Cards compliant with ISO/IEC Standard 7816-series have contacts which interface with a terminal via ohmic contact ("contact"-type cards). Cards compliant with, e.g., ISO/IEC Standard 10536-series or 14443-series rely on close, or remote-coupling with a terminal (hereinafter generally referred to as "contactless" -type cards). Some smart cards are now being produced with both types of contact schemes, i.e., they are compatible with terminals having contact interfaces or contactless interfaces. These types of cards are sometimes referred to as combination cards.

A block diagram of a smart card 12 is shown in Fig. 1. Smart card 12 can be implemented as a processor card as illustrated in Fig. 1. Of

course, smart card 12 can be implemented in a variety of different ways, and the card of Fig. 1 is intended to illustrate one existing version of card that may be used with the present invention. Smart card 12 has an embedded microcontroller with a microprocessor 14, card memory including random access memory (RAM) 16, read-only memory (ROM) 18, non-volatile memory 20, an encryption module 24, a card reader interface 22, and first and second contact structures 26, 28. Other features of the microcontroller may be present but are not shown, such as a clock, a random number generator, interrupt control, control logic, a charge pump, power connections, and the like.

Microprocessor 14 is any suitable central processing unit for executing commands and controlling the device. RAM 16 serves as storage for calculated results and as a stack memory. ROM 18 stores the operating system, fixed data, standard routines, and look-up tables. Non-volatile memory 20 (such as EPROM or EEPROM) serves to store information that must not be lost when the card is connected to a power source but that must also be alterable to accommodate data specific to individual cards or any changes possible over the card lifetime. This information might include a card identification number, a personal identification number, authorization levels, cash balances, credit limits, etc. Encryption module 24 is an optional hardware module used for performing a variety of encryption algorithms. Card reader interface 22 includes the software and hardware necessary for communication with the outside world. In the smart card shown, two different contact structures are provided on the card, a first contact structure 26 which may be e.g., a ISO/IEC Std. 7816-series style contact, and a second contact structure 28 which may be, e.g., an ISO/IEC Std. 10536-series or 14443-series style contactless or close-coupled interface. Other types of contact schemes may also be possible, including remote-coupled, etc.

One application for smart cards such as smart card 12 of Fig. 1, is as an electronic purse or "stored value card". In this type of application, a prepaid amount of value or representation of currency is stored in the smart card memory for use in the place of conventional cash. The source of the value stored in a card may be, for example, a bank account, a credit card, a debit card, another electronic card, currency, or other sources of funds. In operation, the stored value card is inserted into a terminal device which communicates with the card through the contacts (e.g., via ohmic or close-coupling, depending upon the type of interface supported).

The terminal device may be, e.g., incorporated into a vending machine, cash register, or other point-of-sale device. Value equal to the amount of the purchase is passed from the consumer's card to the merchant terminal or to a central clearing and authorization system so the merchant or vendor may be credited with the appropriate amount of value. Correspondingly, the amount of available currency in the consumer's card is reduced by the amount of the purchase. These stored value applications are becoming more widely used in the U.S. and throughout the world. Currently, the primary applications in which stored value cards are used include purchases at vending machines, telephones, gasoline pumps, and other point of sale devices in which the goods or services purchased are low dollar value items. Other types of stored value systems use a token-type approach to value. For example, when a consumer's card is loaded with value, the consumer can pass that value to another consumer's card or to a merchant's card. The recipient of the value, or the token, can then use the value without needing to redeem the value by going through a bank clearing and settlement process. For the purposes of this disclosure, a stored value system will be described which uses a clearing and settlement function. However, those skilled in

the art will recognize that techniques and features of the invention may be implemented in other types of stored value systems as well.

Stored value card systems are operated as both "open" or "closed" systems. "Closed" systems are limited to specific participating merchants. For example, some universities offer stored value cards to their students for use in university cafeterias and book stores. The cards are not compatible with point of sale devices or terminals outside the university. These "closed" systems are easier to implement from a technical and security perspective because of this closed universe of uses. "Open" systems, on the other hand, allow use of the card at substantially any merchant who has a compatible point of sale device or card reader. An example of such a system is the Visa Cash® system offered by Visa International Service Association of San Francisco California. In many respects, "open" stored value card systems are preferable to "closed" systems, as they allow the cards to be used in a wide variety of places, for various types of purchases. "Open" systems, however, are more difficult to implement and have more rigorous security requirements.

As these stored value card systems become more widely available, it is desirable that they be capable of use in more payment applications. For example, a solution for using stored value cards in making payments over open networks such as the internet is described in commonly-assigned U.S. Patent Application Serial No. 08/951,614, entitled "Internet Payment System Using Stored Value Cards", filed on October 16, 1997, which is incorporated herein by reference for all purposes.

It would also be desirable to use smart cards in transit or access applications, such as toll booths, parking meters, etc. Public transit systems are widely used in most large communities. These systems are relied upon to efficiently transport large numbers of people.

To transport large numbers of people, especially at peak traffic times, it is necessary that passengers be able to enter and exit the transit system quickly in an even flow. Existing transit systems use a variety of payment and authorization methods to make sure passengers can quickly board and disembark. For example, many buses, subways, and trains allow passengers to use currency, tokens or passes to pay for their ride. The objective is to verify a passenger has paid or has a valid pass as quickly as possible to ensure the greatest possible throughput, especially at peak use times. Turnstiles and other access devices are used to permit or deny access depending upon proper verification of a rider's payment. It would be desirable to permit transit passengers to gain access to transportation using smart cards. Further, it is desirable to implement such an access system using an "open" stored value card system (i.e., a system which is not limited to a closed universe of merchants or suppliers).

Unfortunately, however, existing smart cards and "open" stored value card systems are too slow for acceptable use in most transit systems. The security and authentication processes needed to verify a transaction using existing systems typically takes over 1.0 to 2.0 seconds. This is too long of a wait for, e.g., transit riders trying to gain entry to a transit system at rush hour. It would be desirable to implement an "open" stored value system with a quicker access and authorization time, without compromising the ability to perform full authentication and verification of cards.

Accordingly, a system and method for fast smart card transactions is described.

SUMMARY OF THE INVENTION

Embodiments of the present invention provide a smart card system suited for applications where cardholders demand a very fast

transaction approval and response time and where large numbers of transactions are presented for authorization. In one specific embodiment, a cardholder presents a smart card to an interface device coupled to an access device. First and second sets of information are read from the card. The access device performs a "first phase" of the transaction, including, e.g., low-level authentication of the card using first data read from the card. If the first phase of the transaction is successful, access is granted.

A subsequent, or "second phase" of the transaction process may then be performed in the background between the access device and a terminal device. This second phase of the transaction can be performed in a batch on a predetermined time schedule. The first phase of the transaction may include verifying the card has sufficient funds available to make the requested access, or may simply verify the card has not yet expired. The second phase of the transaction generally includes steps to authenticate that the card was validly issued and that a valid card was involved. In one embodiment, the second phase of the transaction includes performing public key encryption. In another embodiment of the present invention, a fast smart card transaction may be performed using symmetric encryption techniques. Portions of the first and second phases of the transaction may include the use of symmetric encryption and decryption techniques.

In one embodiment of the present invention, the access device is a turnstile such as, for example, an admission turnstile at a light rail station, etc. The terminal device may be a terminal located near the access device, and may be configured to service a number of access devices on a regular basis.

The result is an access system suited for applications where a large number of cardholders need quick authorization response times. Embodiments of the invention can be used in applications requiring

access times of less than about 0.25 to 0.50 seconds. From the cardholder's perspective, systems implemented according to the invention process and approve the transaction almost instantly. Systems implemented using features of the invention allow the use of "open" stored value card system applications in transit or access applications, giving cardholders the ability to use a single card for purchases at point of sale devices and transit applications.

A further understanding of the nature and advantages of the invention may be realized by reference to the remaining portions of the specification and the drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a block diagram of a smart card;

Fig. 2 is a block diagram of a smart card system incorporating features of an embodiment of the present invention;

Fig. 3 is a flow diagram depicting a first phase of a transaction process according to an embodiment of the present invention;

Fig. 4 is a flow diagram depicting a second phase of a transaction process according to an embodiment of the present invention;

Fig. 5 is a flow diagram depicting a first phase of a transaction process according to a further embodiment of the present invention;

Fig. 6 is a flow diagram depicting a second phase of a transaction process according to a further embodiment of the present invention;

Fig. 7 is a flow diagram depicting a first phase of a transaction process according to a further embodiment of the present invention;

Fig. 8 is a flow diagram depicting a second phase of a transaction process according to a further embodiment of the present invention; and

Fig. 9 is a flow diagram depicting a general process flow according to embodiments of the present invention.

DESCRIPTION OF SPECIFIC EMBODIMENT(S)

Embodiments of the present invention permit existing stored value card applications to be used in environments, such as transit or access control, which require fast authorization response times.

Embodiments of the present invention will now be described by referring first to Fig. 2, where a specific implementation of a stored value card system 10 is shown. Stored value card system 10 includes a consumer card 12, an interface device 30, an access device 40, a terminal 60, a concentration point 80, and a clearing, authorization, and settlement system 90.

Consumer card 12 is a smart card such as the card shown in Fig. 1. For the purposes of describing embodiments of the invention, consumer card 12 has at least a stored value application program 13 stored in card memory, e.g., in ROM 18. Additionally, consumer card 12 may have a transit application program 15 stored in memory which, although a separate application, may run in conjunction with stored value application program 13. Other application programs may also be stored in memory of consumer card 12, such as credit/debit applications, loyalty

programs, health card information programs, licensing programs, or the like.

For the purposes of describing the invention, consumer card 12, access device 40 and terminal 60 will be described as passing commands in a "command-response" format, as set forth in, e.g., "EMV '96 Integrated Circuit Card Specification for Payment Systems", Version 3.0, June 30, 1996, published jointly by Europay, MasterCard, and Visa International Services Association, and incorporated herein by reference. For each command generated by, e.g., access device 40, a response from, e.g., card 12 is expected in a certain format. Those skilled in the art will recognize that other protocols may be used to implement features of the invention.

Interface device 30 operatively couples consumer card 12 with access device 40. Interface device 30, in a presently preferred embodiment, is a contactless interface device which is designed to transmit close-coupled power and data signals between consumer card 12 and access device 40.

Access device 40 includes a router 42, card handler 44, access device application hardware and software 46, a terminal interface application 48, and a buffer 50. Router 42 is hardware and software adapted to route information between functional blocks. Card handler 44, which may be integrally designed with interface device 30, provides functionality to accept and communicate with consumer card 12. For example, card handler 44 may include routines to control interface device 30 to perform initial power-on and detection of a card, such as in accordance with the protocol set forth in the ISO/IEC 10536 standards. Terminal interface application 48 forwards batch transaction information from access device 40 as well as receives software updates and commands

from terminal 60. Buffer 50 stores batch information regarding transactions made at access device 40.

Access device hardware and software 46 include functionality to perform a first phase of the transaction process, including performing a low-level authentication of consumer card 12 as well as functionality to permit, e.g., access where a successful low-level authentication occurs. In one specific embodiment, access device 40 is a transit access gate, and access device hardware and software 46 includes hardware and software to permit cardholder access through a turnstile if the transaction is authenticated. In other embodiments, access device 40 may be a toll booth gate, parking meter, vending machine, video game machine, controlled entry facility door, or other applications where it is desirable to quickly authorize stored value or other types of smart card transactions.

Terminal 60 includes a router 70, an access device interface 72, a security card handler 62, a security card 64, a data store 66, a terminal application hardware and software 68, and a concentration point handler 74. Router 70 is hardware and software adapted to route information between functional blocks. In conjunction with security card handler 62, security card 64 controls the command sequence of the terminal and provides transaction and batch security. Security card 64 may be a small form-factor card with a single-chip processor thereon, and is used to authenticate and validate each transaction. Because the security card may contain secret information, such as private encryption keys as will be discussed below, it is desirable that security card 64 be physically secured and controlled within terminal 60.

Security card 64 may contain a transaction identifier for each transaction (such as a counter incremented for each transaction), a session key, and master keys. Those skilled in the art will recognize that

other types of security modules may be used. For example, dedicated hardware security modules, removable smart card modules, or, in certain controlled environments, software modules may also be used. Further, security card 64 may be a smart card which has the ability to participate in the stored value system by receiving token value which can then be passed on to other smart cards in the system.

While access device 40 and terminal 60 are shown to be separate or discrete entities, they may also be implemented as a single logical or physical entity.

Access device application 72 receives batch transaction information from access device 40 and forwards software updates and commands to access device 40. Terminal application hardware and software 68 includes functionality to control terminal actions.

Data store 66 controls the storage of purchase transactions and totals. Concentration point handler 74 controls the sending and receiving of information to and from a concentration point 80. Concentration point 80 is a staging computer that communicates with any number of terminals 60 to collect batches of transactions. For example, concentration point 80 may sequentially make contact with a number of terminals 60 at prearranged times on a daily basis to collect batches of transactions stored in the data stores of each terminal. Further, after concentration point 80 has collected the batch information from each terminal, concentration point 80 may forward update information to each terminal, such as: acknowledgment information acknowledging receipt of the batch information; date changes for the terminal and access device; encryption key updates and changes; and bad card lists for the terminal and access device.

In one embodiment, concentration point 80 acts as a store and forward device and does not alter or validate the data it receives. The concentration point sends the collected transaction batches to a clearing, authorization and settlement system 90 for processing. Once processed, batch acknowledgments, along with other system updates, are sent to terminal 60 via the concentration point 80. The concentration point ensures a safe transfer of data between terminals and the clearing, authorization and settlement system 90, and prevents overloading of the system. A service provider, such as the party running the transit access terminals, contracts with a concentration point for collection of the transit payments. The concentration point may also be an existing central facility such as a central office of, e.g., a transit company which collects payments from all of the access devices operated by the company. Systems implemented according to the invention need not have concentration points, they may instead be implemented by having the terminal devices connect directly to clearing, authorization, and settlement system 90. As an alternative implementation where features are implemented in a token-type stored value system, the clearing, authorization and settlement system may not even be necessary (e.g., value may be simply passed from the consumer smart card to the security smart card and then from the security smart card to other recipient smart cards in the system).

Software functionality in consumer card 12, access device 40, terminal 60 and concentration point 80 may be implemented using a wide range of programming languages known in the art. For example, terminal interface application 48, or any of the other software modules, may be implemented using "C", "C++" Java or similar code. It is contemplated that Java code will work well for implementing the functionality described herein, as Java is platform independent, allowing the functionality to be used on a variety of device types.

In an alternative embodiment depicted in Fig. 2, a number of interface devices 30a-30n are provided with a number of access devices 40a-40n. These interface devices and access devices may be grouped to operate in conjunction with a single terminal 60 by providing a number of security cards 64a-64n and security card handlers 62a-62n in terminal 60. Each security card and handler will operate to interact with each access device. Further, a number of terminals 60a-60n may also be provided to work with a number of different groups of access devices. The result is an ability to support a number of devices with a relatively small number of terminals. A number of terminals 60a-60n may be coupled to each concentration point 80. Other configurations and groupings of devices may also be used, as will now be apparent to those skilled in the art.

Referring now to Fig. 3, a flow diagram depicting a first phase of a transaction process 100 involving consumer card 12 and access device 40 is shown. In operation, access device 40 has a wait state 102 where it awaits presentation of the next card. For example, access device 40 may be an admission turnstile at, e.g., a metro rail station, or any of the other access devices discussed herein, or other access devices designed to permit access to goods or services upon confirmation of payment. When a cardholder presents consumer card 12 in step 104, access device 40 detects that a card is present (step 106). Access device application hardware and software 46 and card handler 44 then generate a command or request for card information from the card (step 108). This command or request is presented to the card through interface device 30. The command or request presented to consumer card 12 may include transaction information such as an identification of the value which needs to be transferred (e.g., the amount of the fare to be paid).

Consumer card 12, upon receipt of the request for information, performs manipulations and calculations to prepare and forward the requested card information (step 110). These calculations may include specific calculations dictated by transit application program 15 if present, or may simply involve a determination of whether sufficient funds exist in stored value application program 13. For the purposes of this disclosure, it will be assumed that the calculations will be related solely to verifying whether funds exist in the stored value application; however, those skilled in the art will recognize that other calculations may be performed, such as checking transit attributes in transit application 15, e.g., checking the cardholder's status (senior, student, monthly passholder, etc.) or the like.

In this exemplary embodiment, the card information forwarded back to access device 40 may include information such as: card identification information; card issuer identification information; the value requested by access device 40; the expiration date of the card; and any other information needed to authorize and authenticate the transaction.

This card information is passed to access device 40 through interface device 30. When the card information is received, access device 40, using access device application hardware and software 46, performs a low level authentication (step 112). The type of low level authentication performed is chosen to reduce the amount of time needed to authenticate the transaction. For example, in one embodiment, low level authentication step 112 involves ensuring the card 12 has sufficient value to pay the requested value. Other steps may be included in the low level authentication process while still maintaining a fast response time, such as: checking the card number against a bad card list, performing some velocity or floor limit checking, verifying that transit application program

15 is not expired or incompatible with the requested access, etc. In an embodiment to be described in conjunction with Figs. 7-8, the first phase of the transaction process also serves to authenticate the access device to the card and the card to the access device.

If low level authentication step 112 produces a satisfactory result, access device application hardware and software 46 will authorize (step 114), e.g., entry or admission of the cardholder. If low level authentication step 112 fails (e.g., the cardholder had insufficient funds to cover the cost of the transaction or had an incompatible currency type, etc.), admission or entry is declined (step 114) and access device 40 may revert to a wait state to await presentation of the next card (step 102), or representation of the declined card.

Where authorization is given, a confirmation signal or message may be sent to consumer card 12 (in step 118) and the card can store a receipt or other proof of transaction (step 120) in card memory. The cardholder is then permitted access or entry. For example, access device hardware and software 46 may generate an access permission signal (step 116) which activates a turnstile or other entry device. The same signature or receipt should also be sent to access device for use in subsequent processing. This information may be, e.g., stored in the batch queue.

Once access has been granted, access device application hardware and software 46 stores the card information received during step 112. The card information received for each transaction is stored in buffer 50. This information is retained in the buffer for subsequent processing to be described. Once access device 40 has stored the transaction information in buffer 50, the device increments an internal counter (NT) (step 124). Access device 40 then reverts to a wait state to await presentation of the next card (step 102).

Access devices or other terminal devices implemented using features of the invention are suited for applications where a large number of cardholders need quick authorization response times. Typically, these applications are low-dollar transactions, such as transit terminals, where users can not be expected to patiently wait for response times in excess of, e.g., 1.0 to 2.0 seconds. From the cardholder's perspective, systems implemented according to the invention process and approve the transaction almost instantly. By reducing the amount of information processing required to perform this access authorization, systems implemented with features of the present invention are capable of authorizing access, transferring value, and storing a confirmation message using stored value cards in approximately less than 0.25 to 0.5 seconds, for example. Systems implemented using features of the invention allow the use of "open" stored value card system applications in transit or access applications, giving cardholders the ability to use a single card for purchases at point of sale devices and transit applications.

At this point in the transaction, although the first phase of the transaction process 100 is completed and the cardholder has been granted access, the transaction is not yet fully authorized. Referring now to Fig. 4, a flow diagram depicting a second phase of the transaction process 140 is shown. Performance of this second phase of the transaction process 140 is transparent to cardholders who use the access device, and will typically be performed some time after the cardholder has left the vicinity of the access device (e.g., after the cardholder has been granted access to board or enter a transit system, or after the cardholder has passed through a toll gate, etc.).

The second phase of the transaction process commences in step 142, when a batch process is initiated by access device application hardware and software 46. This batch process may be initiated based on,

e.g., passage of time (e.g., once every 10 minutes, hour, or day, etc.) or after a certain number of transactions have accumulated in buffer 50 (e.g., once every 100 transactions, etc.). Alternatively, the process is performed whenever there is any information in buffer 50 and terminal 60 has CPU cycles available (i.e., the process is constantly running, but at a lower priority). While Fig. 3 depicts the access device as having control over initiation of the batch process, upon reading this disclosure those skilled in the art will recognize that terminal 60 may instead be responsible for initiating the batch process. For example, terminal 60 may initiate the process after a certain amount of time has elapsed.

Once batch process (step 142) is initiated, access device application hardware and software 46 selects the first record in the buffer as the current record (step 144) (e.g., buffer 50 is a first-in first-out device or FIFO). In a currently-preferred embodiment, the first record to be processed in the batch is the record with the lowest associated counter number. This current record is passed to terminal 60 via terminal interface application 48 and access device interface application 72 for processing. Terminal 60, using security card 64 and security card handler 62, performs a high level authentication on the current record in step 146.

This high level authentication may include, e.g., steps necessary to verify that the card used was a valid card and that the card was validly issued (steps which typically involve the decryption of information). By placing the responsibility for these computationally-intensive (and relatively slow) actions on access device 40 and terminal 60, interaction between card 12 and access device 40 is sped up considerably, without substantial compromise in authentication integrity and security.

The high level authentication process 146 may also include, e.g., steps necessary to generate a signed message for transmission to the

stored value system's clearing, authorization and settlement system. The signatures and computations required to perform this portion of the process are relatively time consuming; therefore, by performing these steps in the second phase of the transaction, the overall transaction time, from the consumer's perspective, is reduced substantially. This information is stored in data store 66 (step 148). Upon completion of high level authentication for the current record, a transaction counter is incremented in security card 64 (step 150). A counter in access device 40 is similarly incremented (152), and the next transaction in buffer 50 is then processed. Alternatively, the counter in access device 40 may be synchronized with the counter in security card 64 after all transactions in the current batch have been processed. Those skilled in the art, upon reading this disclosure, will recognize that the access device counter should be synchronized with the security card counter at some point before a new batch process is initiated. Alternatively, a synchronized clock or other means may be used to ensure access device 40 and terminal 60 process each transaction appropriately.

Access device 40 checks to determine if any more records exist in the batch (step 154). If more records exist, the next record is selected as the current record (step 144) and steps 146-154 are repeated until the batch is exhausted, at which point access device 40 will await initiation of the next batch process (step 156). Those skilled in the art will recognize that the batch process may alternatively involve passing the entire contents of buffer 50 to terminal data storage 66 at which time terminal 60 can iteratively process the batch without interacting with access device 40. Buffer memory 50 may be partitioned to permit a batch process to be completed in the background while initial authentication process 100 is completed in the foreground, allowing access device 40 to be used by, e.g., transit riders, while the batch is processed.

There may be instances where a card presented to access device 40 passes the low level authentication process of Fig. 3, but which for some reason does not pass the high level authentication process of Fig. 4. In this event, declined or failed transaction information is stored in data store 66 (step 149). This information is passed to clearing and authorization system 90 for subsequent resolution. For example, the cardholder or card issuing bank may be contacted to resolve the discrepancy, or the card information may be added to a bad card list to be stored in each access device 40 of a transit system.

Terminal 60 stores the authorized and failed transaction information in data store 66 and, e.g., at some subsequent time, forwards the information to the stored value system's clearing, authorization and settlement system 90 (step 158). This information may be initially forwarded to, e.g., a concentration point 80 or some other form of intermediary storage location, or may be forwarded directly into clearing, authorization and settlement system 90.

The result is a stored value system suited for applications where cardholders demand a very fast transaction approval and response time and where large numbers of transactions are presented for authorization.

One specific implementation of a two-phase transaction process incorporating features of the present invention will now be described by referring to Fig. 2 in conjunction with Figs. 5 and 6. In this embodiment, consumer card 12 is a card which carries public key generated certificates. While any of a number of cryptosystems may be used, the following discussion describes an implementation using public key cryptography. This specific embodiment of the present invention employs a known authenticating technique utilizing "digital signatures" which allows a first party to employ a secret key to "sign a message"

which the receiving party or a third party can validate using the first party's public key.

Using such digital signatures, it is possible for any digital message to be signed so that the recipient is assured that the message is received as sent, and that it is not a forgery. This is done by using the "public key" and digital signature methodology such as described by at least U.S. Pat. No. 4,405,829, hereinafter referred to as RSA technique. There are other public key and signature techniques which use methodologies other than RSA. Examples of other public key or signature techniques include Fiat-Shamir, Ong-Schnorr-Shamir, and several others derived from zero-knowledge proof techniques. The present invention is not limited to any particular public key or signature technique. These techniques, and others, are described in "Applied Cryptography", by Bruce Schneier, Second Edition 1996, which is incorporated herein by reference for all purposes.

A user who has a public/private key pair can digitally sign a message by "encrypting" (or "signing") the message or a hash of it with the user's private key before transmitting the message. Recipients of the message can verify the message or signature by decrypting it with the sender's public encryption key. Anyone who has the user's public encryption key can read the message or signature, but only the sender having the secret decryption could have created the message or signature. In general, the digital signature assures the recipient of the integrity of the message at the time the signature was computed. Again, the use of these public key techniques is not necessary for implementation of systems according to the present invention, and are merely described herein as one specific embodiment.

In an exemplary embodiment to be described in conjunction with Figs. 2, 5, and 6, consumer card 12 stores the information shown below in Table 1.

DATA ITEM	DESCRIPTION
ICERT	Card Issuer's Public Key certificate
CCERT	Card Public Key certificate
BIN	Card Issuer identification number
ID	Card serial number
ALGCA	Algorithm used for purchase
DEXP	Expiration date of card
BAL	Balance of the card purse
CURR	Currency of the card purse
CEXP	Currency exponent of the purse
AM	Authentication mode
NT	Transaction number
VPKI	Certificate authority public key number
CSKEY	Card secret key

--Table 1--

Some of this information is stored in the card when the card is personalized (e.g., the BIN, ID, ALGCA, DEXP, CURR, CEXP, AM, VPKI, CKEY, CCERT, and ICERT). Other information regarding the card balance (BAL) and the counter or transaction number (NT) are updated as the card is used.

A public key certificate hierarchy is used in this specific embodiment. As an exemplary hierarchy, a first public key pair (CAPK, CASK) is generated for a certificate authority (CA). A second public key pair (IPK, ISK) is generated for the card issuing bank. And a third public

key pair (CPK, CSK) is generated for each card 12. The issuing bank sends their public key pair (IPK, ISK) to the certificate authority (CA) which generates an issuer certificate (ICERT) using CASK. ICERT is sent to the issuing bank. All cards issued by the issuing bank are personalized with this issuer certificate (ICERT) as provable evidence that the card is issued by a valid issuer. Also during personalization, the issuing bank generates a card certificate (CCERT) using the card public key pair (CPK, CSK) and the issuing bank's secret key (CASK). This serves as provable evidence that the card is a valid card. CCERT and ICERT are stored in consumer card 12 (see Table 1, above).

In this particular exemplary embodiment, the card certificate (CCERT) is formed using RSA encryption techniques and includes information uniquely identifying the card, such as: information identifying the card and the issuer of the card (e.g., ID and BIN); information regarding the formation and expiration of the certificate; and information regarding the algorithm used to generate the certificate. The card certificate may be formed and stored on the card when the card is personalized, and may be generated using public key encryption techniques. The card certificate (CCERT) also contains a hash (using, e.g., the Secure Hash Algorithm SHA, set forth in the Digital Signature Standard NIST FIPS PUB 186, U.S. Department of Commerce, May 1994) of the card's public key CPK.

The issuers certificate (ICERT) may also be formed using RSA techniques and include information identifying the card issuer, information identifying the certificate (e.g., expiration date, etc.), and information regarding the algorithm used to generate the card certificate (CCERT). ICERT may also contain a hash of the issuer's public key IPK.

In this manner, a party having the certificate authority (CA) public key (CAPK) and the issuer's certificate (ICERT) can recover the

issuer's public key (IPK), while a party having the issuer's public key (IPK) and the card's certificate (CCERT) can recover the card's public key (CPK).

Referring now to Fig. 5, one specific implementation of a first phase of a transaction process 200 according to the invention is shown. Access device 40 loops in a wait state 204 for the next card to be presented for access authorization. Consumer card 12 is presented to an interface device 30 of access device 40 in step 202. Access device 40 provides power and performs initial protocol setup (e.g., generates an ATR sequence as described in ISO/IEC Standard 10536, etc.) to establish contact with card 12 (step 206). Card 12 responds to this initial sequence (208).

Access device 40, upon verifying that card 12 is a compatible card, proceeds to send a SELECT command to the card (step 210) to select stored value application program 13 as the application to be used for this particular transaction. In response to the SELECT command, card 12 replies (step 212) with information regarding stored value application program 13. Those skilled in the art will recognize that this SELECT command and response may vary from system to system, and is provided here only as an example.

At this point, access device 40 sends an INITIALIZE IEP FOR PURCHASE command to card 12 (step 214). Card 12, in response, first increments a transaction counter (NT), and then forwards information regarding the card, including, for example: BIN, ID, ALGCA, DEXP, BAL, CURR, CEXP, AM, NT, and VPKI. At this point, access device 40 may perform a low-level authentication (step 218) to determine if the transaction can proceed. For example, this low-level authentication may include verifying that the currency type (CURR) is supported (e.g., if the access device requires U.S. Dollars, the value of CURR must indicate

the card holds value in U.S. Dollars). Low-level authentication step 218 may also include verifying that the card balance BAL is greater than or equal to the cost of the purchase (e.g., the cost of the fare purchased). Alternatively, access device 40 can rely on card 12 performing the verification of funds availability and compatible currency, and access device 40 may simply (or additionally) verify that card 12 has not yet expired (i.e., check card expiration date CEXP). Whatever the authentication steps chosen for performance by access device 40, they should be capable of performance quickly, and preferably without the need for computationally-intensive cryptography.

Access device 40 checks to see if the required conditions are satisfied (step 220). If low-level authentication is not satisfied, access device 40 terminates the transaction and reverts to wait state 204 without further processing.

If low-level authentication is satisfied, access device 40 proceeds to issue a READ RECORD command (step 222). In response to this first READ RECORD command, card 12 responds by sending issuer's certificate (ICERT) (step 224). This certificate is stored in access device 40 (step 226). Access device 40 then issues a second READ RECORD command (step 228), in response to which card 12 sends card certificate (CCERT) (step 230) to access device. To speed up the process, these certificates may be stored in memory or cached by storing frequently-used certificates or at least a list of recently-used certificates.

Whether retrieved from the card or from a cache, access device 40 stores the retrieved certificate (step 232) and issues a DEBIT IEP command to the card (step 234). This DEBIT IEP command is passed to card 12 along with information regarding the purchase amount (M). Information regarding the type of currency, the currency exponent, and the terminal identifier and location may also be passed with this

command. In addition, information to ensure the transaction message generated by the card is unique may also be passed, including: time, date, and the current value of the access device's transaction counter (NT).

Card 12 responds to this DEBIT IEP command by first performing several internal calculations, including verifying that sufficient value is available to make the purchase. The amount of the purchase (M) is decremented from the card balance (BAL) and a record of the transaction is stored in non-volatile memory of the card (step 240). In addition, card 12 generates a signature of the transaction (SP) for transmission to access device 40. This signature (SP) is generated using the card's secret key (CSK) and includes, for example, information which may be used to prove the fact of the transaction, such as: the amount debited; the new balance of the card; the card's expiration date; the currency code; the currency exponent; the card's transaction number; the access device's transaction number; access device identifying information (including information regarding the security card 64 associated with the particular access device); etc. This signature (SP) may also be stored in non-volatile memory of card 12 (step 240).

Access device 40 receives and stores this information (step 238) in buffer 50. Access device 40 then permits cardholder access to the goods or services purchased (e.g., entry to a transit system may be permitted by activating a turnstile), or provides confirmation to a system operator that the transaction has completed the first phase of the transaction process 200 (e.g., a visual signal may be presented to a bus driver, etc.)

This first phase of the transaction process 200 may be completed in a very quick manner (e.g., on the order of 0.25-0.5 seconds from the time the card is presented to the interface device until the time of access permission), while providing a relatively sure means of

authenticating the existence of funds to pay for the transaction. The access device 40 now has sufficient information stored in buffer 50 to complete the second phase of the transaction process (e.g., to perform a full authentication of the card and funds).

Referring now to Fig. 6, a second phase of transaction process 300 according to one specific embodiment of the present invention is shown. This authentication process takes place between access device 40 and terminal 60 independent from the interaction between access device 40 and consumer cards. In the specific embodiment shown, terminal 60 is shown as assuming the responsibility to contact access device 40 to initiate a batch second phase transaction process 300.

In step 302, terminal 60 determines whether it is time to initiate a batch process (step 304). This determination may be made based on the lapse of time or some other factor. When a batch process is initiated, access device 40 is notified and sets a pointer to the top transaction record stored in buffer 50 (step 306), and provides some indication to terminal 60 that it is ready for batch processing. Terminal 60 then proceeds to sequentially perform high-level authentication and processing for each transaction record stored in buffer 50.

Terminal (or, more particularly, security card 64) controls the high-level authentication and processing for each transaction record. This process begins in step 308 when security card 64 generates an INITIALIZE IEP FOR PURCHASE command which is passed to access device 40. The access device, under control of terminal interface application 48, responds with card information from the first transaction record, including information such as: BIN, ID, ALGCA, DEXP, BAL, CURR, CEXP, AM, NT, and VPKI (information which was sent from consumer card 12 in response to the INITIALIZE IEP FOR PURCHASE command described in conjunction with Fig. 5, above).

Security card 64 then issues a READ command (step 314), prompting access device 40 to send the issuer's certificate ICERT from the current transaction record (step 316). Security card 64, using a Certification Authority public key (CAPK) stored in security card memory, recovers the issuer's public key (IPK) from the issuer's certificate (ICERT). If this recovery cannot be performed, the transaction fails high-level authentication, as the consumer card used to perform the purchase transaction was not validly issued.

If ICERT is successfully recovered, security card 64 generates another READ command (step 320), prompting access device 40 to reply with the card certificate (CCERT) in step 322. Security card 64 recovers the card public key (CPK) using the issuer's public key (IPK) recovered in step 318. If the card public key cannot be successfully recovered, the high-level authentication process fails.

If the card public key (CPK) has been successfully recovered, the authentication process continues to step 326, where security card 64 issues a DEBIT IEP command. This DEBIT IEP command is passed to access device 40 along with information regarding an identification of the security card 64. In addition, information to ensure the transaction message generated by the card is unique may also be passed, including: time, date, and the current value of transaction counter (NT) of security card 64.

Access device 40 responds to this DEBIT IEP command by forwarding the signature of the transaction (SP) generated by consumer card 12. Security card 64 performs the necessary cryptographic steps needed to authenticate the signature of the transaction (330). If this authentication is successful, the transaction from the current record in buffer 50 of access device 40 has been successfully high-level authenticated, and the paid party (e.g., a transit system) will be

guaranteed payment of funds equal to the transaction amount, while the account of the paying party (i.e., the holder of consumer card 12) will be decremented in an amount equal to the transaction amount.

Security card 64 may then sign the transaction record (step 332) using, e.g., a message authentication code (MAC) or the like. Additionally, security card 64 may sign a batch header (step 334) and store the record (step 336) for subsequent forwarding to, e.g., concentration point 80 as described above in conjunction with Fig. 2 and Fig. 4.

Terminal 60 then requests the next transaction from access device 40. If the transaction was not the last transaction in buffer 50 (step 340), access device increments counter NT and retrieves the next record. Steps 306-342 are then repeated until all records in buffer 50 have been subject to high-level authentication.

Embodiments of the present invention, thus, permit a stored value system which uses public key techniques to be used in conjunction with transit applications or other applications which demand fast authorization response times.

A further embodiment of the present invention will now be described by referring to Fig. 2 in conjunction with Figs. 7 and 8. In this embodiment, a specific embodiment of a fast smart card transaction process will be described which may be used in conjunction with a symmetric key based smart card system. Symmetric encryption techniques, such as the commonly-used Data Encryption Standard (DES), are used in many smart card systems. Symmetric encryption techniques are used to authenticate participants to a transaction and to encrypt data.

These processes can consume a substantial amount of processing time during a typical transaction. For example, using existing available smart card technology, each DES encryption operation conducted by a card (including both the consumer card and a security module, for example) may take around 10-20mS. A typical smart card stored value transaction can include between 15-25 DES operations in the interaction between a security module and a consumer card. This time is in addition to any time required to go through an ATR and power on sequence, and any time required to transfer data and do other computations in the transaction sequence. Thus, a DES-based (or other symmetric key) stored value card transaction can take 0.5 seconds or even longer. While this type of a transaction is quicker than most public key or asymmetric-type transactions, it is still too time consuming for most transit or access environments. Embodiments of the present invention reduce the amount of processes which must be performed while the consumer smart card is present in a transaction so that the transaction may occur more quickly as is desirable in, e.g., access-type applications.

Referring now to Figs. 2 and 7, a first phase 400 of a transaction process will now be described. In this process, consumer card 12 is presented to access device 40a via interface device 30a (step 402). Access device 40a loops in a wait state for a card to be presented (step 404). Access device application hardware and software 46 and card handler 44 then initiate a card power-on and ATR sequence in step 406. Card 12 may perform some calculations or processing to determine that, for example, transit application 15 requires funds from stored value application 13. Those skilled in the art will recognize that a number of different calculations and processes may take place at this time.

Once card 12 has responded, and access device 40a has determined that, e.g., value needs to be withdrawn from stored value

application program 13 of consumer card 12, access device generates an INITIATE IEP FOR PURCHASE command (step 410) and sends this command to card 12. Card 12 replies with a variety of card data elements, including, for example: BIN, ID, ALGCA, DEXP, BAL, CURR, CEXP, AM, NT and VKP (these data elements are stored in card 12 as generally described above in conjunction with Table 1). At this time, card 12 may also generate a session key (KSESP) for use in authentication during the transaction. The session key is not sent to access device 40, but the other data elements are (step 412).

In step 414, access device 40 selects an Entry ID number to keep track of the transaction. This Entry ID number is preferably unique for every open transaction in the system, and is used to ensure that access device 40 and security card 64a are able to keep track of each specific open transaction. Only a single transaction will be open at any time for a given Entry ID. Once an available Entry ID is selected (step 414), access device 40 generates an INITIATE PAYMENT command and routes the command, along with the newly selected Entry ID, and the card information received from step 412, to security card 64a (step 416).

Security card 64a, using the card information, generates two signatures (S2 and S3) which will be used to perform mutual authentication between security card 64a and consumer card 12. Those skilled in the art will recognize that other authentication processes may also be used and that this particular approach is for the purposes of disclosure only. In a currently preferred-embodiment, the S3 signature is generated by signing (using KSES generated by security card 64a) data including: an ID of security card 64a; a counter of security card 64a; a balance amount of consumer card 12; CURR of consumer card 12; and CEXP of consumer card 12.

The S2 signature (in a currently-preferred embodiment) is formed by signing (again using the session key KSES) data including: an ID of security card 64a; a counter of security card 64a; and a balance amount of consumer card 12. Security card 64a stores these signatures. Security card 64a then sends a GENERATE DEBIT IEP command back to access device 40a (step 418). Included in the GENERATE DEBIT IEP command, in a currently-preferred embodiment, are the S2 signature and the two most significant bytes of the S3 signature. Only the two MSB of the S3 signature are passed to access device 40a to ensure that access device 40a is unable to create a full transaction record. This guards against an attacker posing as an access device or taking control of an access device to defraud the system.

Access device 40a, upon receipt of the data from security card 64a, generates a DEBIT IEP command (step 420) and sends information including S2 and other information about security card 64a to consumer card 12. Consumer card 12 proceeds to independently generate its own version of S2 (S2') using the received information (step 422). Consumer card 12 then compares S2' to S2 to authenticate security card 64a and terminal 60a (step 424). If the two signatures do not match, the transaction should be aborted (step 426). If they do match, consumer card 12 proceeds by generating its own version of S3 (S3') (step 428) and sends this signature to access device 40a for authentication. Access device 40a compares the card's version of the signature (S3') to the two MSBs of the security card's version of the signature (S3) to determine the authenticity of the card (step 430). If the 2 MSBs do not match, the transaction should fail (step 434). If they are equal, it is likely that both card 12 and security card 64 are authentic and consumer card 12 has properly reduced the card balance by the correct transaction amount. If this is the case, access device 40 writes the transaction information to a

queue (step 436) and authorizes the transaction by, e.g., granting access to a product or services (step 438).

In a currently-preferred embodiment, the following information is written to the queue: data from card 12's response to INITIALIZE IEP FOR PURCHASE; data from the card 12's response to DEBIT IEP FOR PURCHASE (including the card signature S3); a transaction counter from the security card for the specific transaction; an identification of the security card; and the debit amount of the transaction.

Once access has been granted (step 438) and information has been written to queue (step 436), the first phase 400 of the transaction process are complete. Several relatively time-consuming encryption and data transfer processes have been avoided in this first phase 400. To increase throughput and efficiency from the perspective of a cardholder, these encryption and data transfer processes are relegated to a second phase 500 of the transaction process which will now be described in conjunction with Fig. 8. Those skilled in the art, upon reading this disclosure, will recognize that fast smart card transactions can be achieved by performing other processes in later phases while the cardholder is not present, and that the techniques described herein are examples of techniques according to embodiments of the present invention.

Referring now to Fig. 8 in conjunction with Fig. 2, a second phase 500 of the transaction process is shown. Second phase 500 may take place at any time after a cardholder has presented a card at access device 40 and after first phase 400 (Fig. 7) has completed. In general, second phase 500 is preferably performed in the background as a batch process. The batch process may be initiated by any of a number of factors, e.g., a timer may initiate the batch process after a period of time has

elapsed, or a counter can keep track of the number of transaction records stacked in the queue. Whatever factor is used, once a determination is made to process the batch (step 502), a batch access is initiated (step 504) and access device 40a acts to set a pointer to the first Entry ID in the queue (step 506). The transaction record corresponding to the first Entry ID is read from the queue (step 508) and access device 40a issues a COMPLETE PAYMENT command, transmitting data from the transaction record to terminal 60a and security card 64a (step 510). Security card 64a, upon receipt of the information along with the COMPLETE PAYMENT command, compares the S3 signature it had previously generated and stored (see step 418 of Fig. 7) during first phase process 400 with the S3' signature generated by card 12 and received with COMPLETE PAYMENT command. If the two signatures match, security card 64a has verified that consumer card 12 was an authentic card. If the comparison (step 512) fails (i.e., S3 is not equal to S3') the transaction fails and security card 64a stores information regarding the failure for later resolution (step 514).

If security card 64a is successful at authenticating consumer card 12, security card 64a can complete processing for the transaction. The transaction is completed by the performance of several encryption steps. First, security card 64a signs the transaction record using a secret key of the security card (step 516), and then security card 64a signs a batch header of all transactions processed by the security card (step 518), again using a secret key. The signed information is then stored in security card 64a or elsewhere in terminal 60a. Those skilled in the art will recognize that other types of transaction completion steps may also be required, depending upon the type of smart card system used.

Once security card 64a has completed the transaction, the Entry ID is incremented (step 522) and the next transaction is processed.

In a preferred embodiment, access device 40a controls the Entry ID and it is access device 40a which increments the Entry ID. If the last transaction in the queue has been reached (determined by checking at step 524), the Entry ID number may be reset (step 526). Second phase process 500 may repeat until all transaction information in a queue are processed. The processed transactions may then be sent to concentration point 80 for processing through clearing, authorization and settlement system 90.

In the specific embodiment described in connection with Figs. 7 and 8, a relatively substantial amount of processing time is done in the background (e.g., in second phase process 500) so that a cardholder does not need to wait an unreasonably long time to receive access authorization at, e.g., a transit turnstile. In particular, a relatively substantial amount of time is saved by performing steps 512-522 in the background while the cardholder is not present. Those skilled in the art, upon reading this disclosure will realize that features of the invention may be used to split other smart card transaction processes into multiple phases to reduce the amount of processing time required while a cardholder is present.

Referring now to Fig. 9, a generalized overall transaction process 600 according to embodiments of the present invention is shown. Overall transaction process 600 includes a first step of selecting an application (step 602). This step can involve a variety of decisions and operations depending upon the nature of the application and environment. For example, once a card is presented to a transit turnstile, the turnstile may first select a card's transit application and then, if required by the transit application, then select a stored value application of the card. Once an application is selected, a first phase of the transaction may be performed (step 604). In general, this first phase

should encompass any low-level authentication required to authorize access to products or services using the card. If the low-level authentication fails (step 608), the transaction should be aborted. If the low-level authentication succeeds, transaction data should be stored (step 610).

Later, or contemporaneously in background processing, a subsequent phase(s) of processing may take place (step 612). This subsequent phase(s) should be designed to perform any additional, or high-level authentication processes, and should also perform any required formatting or encryption of transaction information. If these steps are successful, the transaction data may be finalized (step 618). If unsuccessful, the transaction should be aborted (step 616) and any appropriate remedial steps may be taken.

While the above is a complete description of the preferred embodiments of the invention, various alternatives, modifications, and equivalents may be used. For example, while specific functional responsibilities were allocated to the access device or terminal, such as initiation of the batch process, the allocations may be reversed in some instances while still producing the multi-phase transaction process of the invention. Further still, those skilled in the art, upon reading this disclosure, will recognize that transaction processes may also be split into more than two phases in some situations, thereby resulting in further speed and processing efficiencies.

Therefore, the above description should not be taken as limiting the scope of the invention which is defined by the appended claims.

WHAT IS CLAIMED IS:

1. A method for using a smart card to gain access through an access device upon payment of a value (V), the method comprising the steps of:

operatively coupling said smart card to said access device;

reading at least a first and a second set of data from said smart card;

performing a first authentication process on said at least first set of data;

permitting access if said step of performing a first authentication process meets a required condition;

storing said at least first and second sets of data in a memory of said access device; and

performing a second authentication process on said at least first and second sets of data after said step of permitting access is accomplished.

2. The method of claim 1, wherein said first set of data includes information indicating a balance (B) of funds available in said smart card, and an expiration date (DEXP) of said smart card.

3. The method of claim 1, wherein said second set of data includes a card public key certificate, a card issuer public key certificate, and a signature generated by said smart card.

4. The method of claim 2, wherein said first authentication process includes the step of verifying that said balance (B) is greater than said value (V).

5. The method of claim 2, wherein said first authentication process includes the step of verifying that said expiration date (DEXP) has not passed.

6. The method of claim 2, wherein said required condition is said balance (B) being greater than said value (V).

7. The method of claim 3, wherein said second authorization process includes the steps of verifying said card certificate, verifying said issuer certificate, and verifying said card signature.

8. The method of claim 1, wherein said second authorization is performed between a terminal device and said access device.

9. The method of claim 8, wherein said terminal device includes a security card.

10. An access system, including:

an interface device, having contacts for coupling with a smart card to read at least a first and a second set of information;

an access device, coupled to said interface device, having at least a first processor for performing a low-level authentication of said first set of information received from said smart card;

a memory in said access device, storing said at least first and second set of information; and

a gate device, coupled to said access device, said gate device granting access to a cardholder after said low-level authentication is performed.

11. The access system of claim 10, wherein said gate device is a turnstile.

12. The access system of claim 10, wherein said first set of information includes at least a balance (B).

13. The access system of claim 12, wherein said low-level authentication includes verifying that said balance (B) is greater than a cost of access.

14. The access system of claim 10, further comprising:

a terminal device, coupled to said access device, said terminal device having at least a first security module for performing a high-level authorization of said at least first and second information received from said card.

15. The access system of claim 14, further comprising:

a concentration point, coupled to said terminal device, for receiving information regarding said high-level authorization.

16. An access control system, comprising:

a plurality of card reading devices, each having a contact structure for reading at least first and second data from a smart card;

a plurality of access devices, coupled to said card reading devices, for performing a low-level authentication on said first data from said smart card, said access devices having a memory for storing said first and second data from said smart card;

a terminal device, coupled to said plurality of access devices, for performing a high-level authentication on said at least first and second data from said smart card.

17. The access control system of claim 16, wherein said terminal device includes a plurality of security cards, each security card for use in

performing a high-level authorization of said first and second data from a specific one of said plurality of access devices.