

TOKEN PROCESSING FOR ACCESS INTERACTIONS

5 CROSS-REFERENCES TO RELATED APPLICATIONS

[0001] None.

BACKGROUND

[0002] In access situations, such as entering a building, event venue, or public
10 transit, there is a need to process many access transactions quickly and efficiently,
while maintaining security. Some transactions that take more than few seconds to
process, or may require many messages. Such transactions can result in delays at
access points such as transit terminals. There is a need to improve the speed of
transactions without significantly increasing the complexity of existing access
15 infrastructures. There is also a need to reduce the risk of allowing fraudulent access
to unauthorized users and to reduce the risk of unauthorized use of access
credentials.

[0003] Embodiments address these and other problems individually and
collectively.

20

BRIEF SUMMARY

[0004] One embodiment comprises a method that includes receiving, by an
access operator computer from an access device of an access provider, a first
transaction request for a first transaction, the first transaction request comprising a
25 credential, the first transaction request associated with a first transaction amount.
The method also includes identifying, by the access operator computer, a token,
based upon the credential. The method also includes transmitting, by the access
operator computer to an authorizing computer, a pre-authorization request message
comprising the credential and a preauthorization amount and receiving, by the
30 access operator computer, a pre-authorization response message from the
authorizing computer. The method also includes receiving, by the access operator
computer from the access device or another access device, a second transaction

request for a second transaction, the second transaction request comprising the credential, the second transaction request associated with a second transaction amount. The method also includes identifying, by the access operator computer, the token from the credential, determining, by the access operator computer, that the token has been used at the access provider. The method also includes forming a total amount by aggregating, by the access operator computer, at least the first transaction amount and the second transaction amount and initiating transmitting a clearing message including the total amount.

[0005] Another embodiment includes a server computer comprising a processor and a computer readable medium comprising code, executable by the processor, for performing a method that includes receiving, from an access device of an access provider, a first transaction request for a first transaction, the first transaction request comprising a credential, the first transaction request associated with a first transaction amount. The method also includes identifying a token, based upon the credential. The method also includes transmitting, to an authorizing computer, a pre-authorization request message comprising the credential and a preauthorization amount, and receiving a pre-authorization response message from the authorizing computer. The method also includes receiving, from the access device or another access device, a second transaction request for a second transaction, the second transaction request comprising the credential, the second transaction request associated with a second transaction amount and identifying the token from the credential. The method also includes determining that the token has been used at the access provider, forming a total amount by aggregating at least the first transaction amount and the second transaction amount, and initiating transmitting a clearing message including the total amount.

[0006] Another embodiment comprises a method that includes receiving, by an access device, a credential from a portable device and identifying, by the access device, a token based upon the credential. The method also includes determining, by the access device, that the token is not on a blacklist, admitting, by the access device, a user associated with the portable device, and transmitting, by the access device to an access operator computer, a transaction request comprising the credential, the transaction request associated with a transaction amount.

[0007] Another embodiment includes an access device comprising a processor and a computer readable medium comprising code, executable by the processor, for performing a method including receiving a credential from a portable device, and identifying a token based upon the credential. The method also includes
5 determining that the token is not on a blacklist, admitting a user associated with the portable device, and transmitting, to an access operator computer, a transaction request comprising the credential, the transaction request associated with a transaction amount.

[0008] Further details regarding embodiments can be found in the Detailed
10 Description and the Figures.

BRIEF DESCRIPTION OF THE DRAWINGS

[0009] FIG. 1 is a block diagram of a system according to embodiments.

[0010] FIG. 2 is a block diagram of a transit operator according to embodiments.

15 **[0011]** FIG. 3 is a block diagram of a terminal according to embodiments.

[0012] FIG. 4 is a swim lane diagram of a message according to embodiments.

DETAILED DESCRIPTION

[0013] Prior to discussing embodiments, some terms can be described in
20 further detail.

[0014] A “user” may include an individual or a computational device. In some embodiments, a user may be associated with one or more personal accounts and/or mobile devices. In some embodiments, the user may be a cardholder, account holder, or consumer.

25 **[0015]** A “portable device” may be a device that can be carried by a user. Portable devices may have capabilities for storing, processing, and communicating information. They may also store payment credentials. A portable device may have a way of generating and or storing cryptographic information for additional security.

Examples of portable devices include laptops, mobile phones, payment cards, and key fobs.

[0016] A “server computer” is typically a powerful computer or cluster of computers. For example, the server computer can be a large mainframe, a
5 minicomputer cluster, or a group of servers functioning as a unit. In one example, the server computer may be a database server coupled to a Web server.

[0017] A “processor” may include any suitable data computation device or devices. A processor may comprise one or more microprocessors working together to accomplish a desired function. The processor may include CPU comprises at least
10 one high-speed data processor adequate to execute program components for executing user and/or system-generated requests. The CPU may be a microprocessor such as AMD's Athlon, Duron and/or Opteron; IBM and/or Motorola's PowerPC; IBM's and Sony's Cell processor; Intel's Celeron, Itanium, Pentium, Xeon, and/or XScale; and/or the like processor(s).

[0018] A “memory” may be any suitable device or devices that can store electronic data. A suitable memory may comprise a non-transitory computer readable medium that stores instructions that can be executed by a processor to implement a desired method. Examples of memories may comprise one or more memory chips, disk drives, etc. Such memories may operate using any suitable
20 electrical, optical, and/or magnetic mode of operation.

[0019] A “secure element” may be a microprocessor chip that can securely store and process data. A secure element can be used to perform cryptographic operations such as encryption and decryption. A secure element can have physical security measures to provide tamper resistance, and can protect the output of
25 decrypted data. Examples of secure elements can include hardware security modules (HSMs) and Trusted Platform Modules (TPMs). Certain cryptographic information, such as secret keys and decryption algorithms, can be stored on a secure element.

[0020] A “public key” may include a cryptographic key that that forms a public
30 key of a public/private key pair. The public key may be designed to be shared (e.g., transmitted between entities) and may be configured such that any information

encrypted with the public key may only be decrypted using a private key associated with the public key.

[0021] A “secret key” or “private key” may include a cryptographic key that forms a private key of a public/private key pair. A private key may be used to decrypt
5 data encrypted with a public key.

[0022] An “access operator” can manage access and provide access processing services for one or more service providers. An access operator may perform backend functions for different access providers. For example, an access operator may be a ticket processor that provides ticket processing services on behalf
10 of many different transit agencies. The access operator may operate an “access operator computer”.

[0023] An “access provider” can be an entity that provides access to a location. For example, an access provider can control access to a building, a subway system, an event venue, etc. A specific example of an access provider may be a
15 transit agency, a building security department, etc. An access provider may control a plurality of access devices.

[0024] An “access device” may be any suitable device for providing access to an external computer system. An access device may be in any suitable form. Some examples of access devices include point of sale (POS) devices, cellular phones,
20 PDAs, personal computers (PCs), tablet PCs, hand-held specialized readers, set-top boxes, electronic cash registers (ECRs), automated fuel dispensers (AFDs), automated teller machines (ATMs), virtual cash registers (VCRs), kiosks, security systems, access systems, Websites, and the like. An access device may use any suitable contact or contactless mode of operation to send or receive data from, or
25 associated with, a mobile device. In some embodiments, where an access device may comprise a POS terminal, any suitable POS terminal may be used and may include a reader, a processor, and a computer-readable medium. A reader may include any suitable contact or contactless mode of operation. For example, exemplary card readers can include radio frequency (RF) antennas, optical
30 scanners, bar code readers, or magnetic stripe readers to interact with a mobile device.

[0025] A “credential” may be any suitable information that serves as reliable evidence of worth, ownership, identity, or authority. A credential may be a string of numbers, letters, or any other suitable characters, as well as any object or document that can serve as confirmation. Examples of credentials include value credentials, 5 identification cards, certified documents, access cards, passcodes and other login information, etc. Other examples of credentials include PANs (primary account numbers), PII (personal identifiable information) such as name, address, and phone number, and the like.

[0026] A “token” may be a substitute value for a credential. A token may be a 10 string of numbers, letters, or any other suitable characters. Examples of tokens include payment tokens, access tokens, personal identification tokens, etc.

[0027] A “payment token” may include an identifier for a payment account that is a substitute for an account identifier, such as a primary account number (PAN) and/or an expiration date. For example, a token may include a series of 15 alphanumeric characters that may be used as a substitute for an original account identifier. For example, a token “4900 0000 0000 0001” may be used in place of a PAN “4147 0900 0000 1234.” In some embodiments, a token may be “format preserving” and may have a numeric format that conforms to the account identifiers used in existing transaction processing networks (e.g., ISO 8583 financial 20 transaction message format). In some embodiments, a token may be used in place of a PAN to initiate, authorize, settle or resolve a payment transaction or represent the original credential in other systems where the original credential would typically be provided. In some embodiments, a token value may be generated such that the recovery of the original PAN or other account identifier from the token value may not 25 be computationally derived. Further, in some embodiments, the token format may be configured to allow the entity receiving the token to identify it as a token and recognize the entity that issued the token.

[0028] “Tokenization” is a process by which sensitive data is replaced with substitute data. For example, a real credential (e.g., a primary account number 30 (PAN)) may be tokenized by replacing the real account identifier with a substitute number that may be associated with the real credential. Further, tokenization can be applied to any other information to substitute the underlying information with a token.

“Token exchange” or “de-tokenization” can be a process of restoring the data that was substituted during tokenization. For example, a token exchange may include replacing a payment token with its associated primary account number (PAN). Further, de-tokenization or token exchange may be applied to any other information to retrieve the substituted information from a token. In some embodiments, token exchange can be achieved via a transactional message, such as an ISO message, an application programming interface (API), or another type of web interface (e.g., web request).

[0029] A “token service computer” can include a system that that services tokens. In some embodiments, a token service computer can facilitate requesting, determining (e.g., generating) and/or issuing tokens, as well as maintaining an established mapping of tokens to primary account numbers (PANs) in a repository (e.g. token vault). In some embodiments, the token service computer may establish a token assurance level for a given token to indicate the confidence level of the token to PAN binding. The token service computer may include or be in communication with a token vault where the generated tokens are stored. The token service computer may support token processing of payment transactions submitted using tokens by de-tokenizing the token to obtain the actual PAN.

[0030] A “token domain” may indicate an area and/or circumstance in which a token can be used. Examples of the token domain may include, but are not limited to, payment channels (e.g., e-commerce, physical point of sale, etc.), POS entry modes (e.g., contactless, magnetic stripe, etc.), and merchant identifiers to uniquely identify where the token can be used. A set of parameters (i.e. token domain restriction controls) may be established as part of token issuance by the token service computer that may allow for enforcing appropriate usage of the token in payment transactions. For example, the token domain restriction controls may restrict the use of the token with particular presentment modes, such as contactless or e-commerce presentment modes. In some embodiments, the token domain restriction controls may restrict the use of the token at a particular merchant that can be uniquely identified. Some exemplary token domain restriction controls may require the verification of the presence of a token cryptogram that is unique to a given transaction. In some embodiments, a token domain can be associated with a token requestor.

[0031] “Token expiry date” may refer to the expiration date/time of the token. The token expiry date may be passed among the entities of the tokenization ecosystem during transaction processing to ensure interoperability. The token expiration date may be a numeric value (e.g. a 4-digit numeric value). In some
5 embodiments, the token expiry date can be expressed as an time duration as measured from the time of issuance.

[0032] A “token request message” may be an electronic message for requesting a token. A token request message may include information usable for identifying a payment account or digital wallet, and/or information for generating a
10 payment token. For example, a token request message may include payment credentials, mobile communication device identification information (e.g. a phone number or MSISDN), a digital wallet identifier, information identifying a tokenization service provider, a merchant identifier, a cryptogram, and/or any other suitable information. Information included in a token request message can be encrypted
15 (e.g., with an issuer-specific key). In some embodiments, the token request message may include a flag or other indicator specifying that the message is a token request message.

[0033] A “token response message” may be a message that responds to a token request. A token response message may include an indication that a token
20 request was approved or denied. A token response message may also include a payment token, mobile communication device identification information (e.g. a phone number or MSISDN), a digital wallet identifier, information identifying a tokenization service provider, a merchant identifier, a cryptogram, and/or any other suitable information. Information included in a token response message can be encrypted
25 (e.g., with an issuer-specific key). In some embodiments, the token response message may include a flag or other indicator specifying that the message is a token response message.

[0034] An “authorizing entity” is an entity that can authorize or approve
30 interactions. An authorizing entity may typically refer to a business entity (e.g., a bank) that maintains an account for a user and is capable of authorizing interactions such as the purchase of goods or services from a merchant. An authorizing entity

may operate an “authorizing computer.” Examples of an authorization entity may be an issuer, a governmental agency, a document repository, an access administrator, etc. An authorizing entity may also issue account parameters associated with the account to a client device. An authorizing entity may be associated with a host
5 system that performs some or all of the functions of the issuer on behalf of the authorizing entity.

[0035] A “clearing message” can be a message that can include information needed for the actual exchange of funds between two entities. Clearing can be a process of exchanging transaction data between processors and issuers. In some
10 cases, information in a clearing message may be used to calculate interchange fees. Clearing may be followed by settlement, where funds are transferred between parties. In some embodiments, clearing messages may be Base II messages.

[0036] A “payment processing network” may be a network that can process messages to facilitate payments. In some embodiments, a payment processing
15 network can include data processing subsystems, networks, and operations used to support and deliver authorization services, exception file services, and clearing and settlement services. An exemplary payment processing network may include VisaNet™. Payment processing networks such as VisaNet™ are able to process credit card transactions, debit card transactions, and other types of commercial
20 transactions. Authorization, settlement, and clearing may be done at the same time (substantially simultaneously, e.g., within a few minutes or hours) or may be done as part of a batch settlement process (e.g., at the end of the day or week). The payment processing network may include a server computer. The payment processing network may use any suitable wired or wireless network, including the internet.

[0037] FIG. 1 shows a block diagram of a system 100 according to
25 embodiments. System 100 can include a user 110 with a portable device 120. System 100 may also include an access provider 130, which may operate a plurality of access terminals 132, 134, 136. The portable device 120 may interact with an access terminal (e.g., 132) to gain access to a secure location (e.g., a train) provided
30 by the access provider 130. The access provider 130 and the access terminals 132, 134, 136 may be in operative communication with an access operator computer 140 and a gateway computer 150. The gateway computer 150 may have access to a

token storage 155 and may be in communication with an authorizing computer 180 via a transport computer 160 and a processing network 170. Any of the devices of system 100 may be in operative communication via any suitable number of types of communication networks or communication channels.

- 5 **[0038]** An exemplary communication network may include any suitable communication medium. The communication network may be one and/or the combination of the following: a direct interconnection; the Internet; a Local Area Network (LAN); a Metropolitan Area Network (MAN); an Operating Missions as Nodes on the Internet (OMNI); a secured custom connection; a Wide Area Network
10 (WAN); a wireless network (e.g., employing protocols such as, but not limited to a Wireless Application Protocol (WAP), I-mode, and/or the like); and/or the like. Message between the entities, providers, networks, and devices illustrated in FIG. 1 may be transmitted using a secure communications protocols such as, but not limited to, File Transfer Protocol (FTP); HyperText Transfer Protocol (HTTP); Secure
15 Hypertext Transfer Protocol (HTTPS), Secure Socket Layer (SSL), ISO (e.g., ISO 8583) and/or the like. The parties may communicate with each other and the ledgers via secure and authenticated communication channels.

- [0039]** User 110 may be a user of an access system, such as a rider or patron of a transit system. The user 110 may have a portable device 120 that can be used
20 to gain access to a secure location. In some embodiments, the portable device 120 can be hand-held and compact so that it can fit into a user's wallet and/or pocket (e.g., pocket sized). Examples of portable device 120 may include cellular phones, personal digital assistants (PDAs), smartphones, payment cards, payroll cards, security cards, access cards, smart media, transponders, and the like. Portable
25 device 120 may interface with terminal 132 or other suitable type of access device using any suitable mechanism including any suitable electrical, magnetic, or optical interfacing system. The portable device 120 may also store credentials such as payment credentials. For example, the portable device 120 may be a payment card such as a credit card and the user 110 may be able to enter a secured area by using
30 the payment card to pay for access.

[0040] Access provider 130 can be an entity that provides access to a secure area. For example, access provider 130 may control access to a subway system. As

another example, access provider 130 can control access to a concert venue, an amusement park, or a government building. Access provider may operate a plurality of terminals. The terminals may be in one or more locations. For example, the terminals may be at a number of subway stations or may be in front of the entrance
5 to a building.

[0041] Terminals 132, 134, and 136 may each be an example of an access device. The terminals 132, 134, 136 may include a physical access control element such as a turnstile or a gate. In some embodiments, a terminal may be a controlled-access door. The terminal may have a reader to receive information from portable
10 device 120. For example, it may have a swipe reader or contactless interface. The terminal may have a gate or some other kind of physical barrier that can prevent a the user 110 from entering a secure location before the user 110 is authorized to do so.

[0042] Access operator computer 140 may provide access processing
15 services to one or more access providers 130. For example, access operator computer 140 may manage transit operations for a plurality of city's public transit services.

[0043] Gateway computer 150 may be a computer that provides access to a payment processing system. Examples of tasks performed by gateway computer
20 150 may include processing access transactions, detecting fraud, and managing transaction data security. In some embodiments, gateway computer 150 may provide an interface between resource providers (e.g., access provider 130, access operator computer 140), and transport computer 160 or processing network 170. Gateway computer 150 may also be in communication with a token storage 155.
25 Token storage 155 may store credentials and tokens associated with the credentials.

[0044] Transport computer 160 may be a computer that routes information between the gateway computer 150 and the processing network 170. In some embodiments, the transport computer 160 may be operated by an acquirer of access provider 130 or access operator computer 140. Transport computer 160 may also
30 reformat received messages to an appropriate format (e.g., reformatting a message from gateway computer 150 to ISO 8583 before sending to processing network 170).

[0045] Processing network 170 may be a payment processing network. Examples of a payment processing network can include VisaNet. Processing network 170 may route messages and facilitate transactions between other entities connected to the processing network 170. Messages may include authorization messages, clearing messages, and settlement messages. In some embodiments, processing network 170 may also process pre-authorization messages.

[0046] Authorizing computer 180 may authorize and process transactions. Authorizing computer 180 may provide an account number or other credentials associated with the user 110 that may be used to initiate payment transactions at access devices other than the terminals (e.g., with merchants that provide goods and services, without providing access to a transit system or sports venue, etc.). In some embodiments, authorizing computer 180 may be an issuer of user 110.

[0047] FIG. 2 is a block diagram of an access operator computer 140 according to embodiments. Access operator computer 140 may comprise a processor 140A that is in operative communication with a memory 140B, a network interface 140C, and a non-transitory computer-readable medium 140D. The computer-readable medium may comprise a token identification module 140D-1, an authorization request module 140D-2, an aggregation module 140D-3, and an encryption/decryption module 140D-4.

[0048] Processor 140A may be implemented as one or more integrated circuits (e.g., one or more single core or multicore microprocessors and/or microcontrollers). The processor 140A may be used to control the operation of the access operator computer 140. The processor 140A can execute a variety of programs in response to program code or computer-readable code stored in memory 140B. The processor 140A may include functionality to maintain multiple concurrently executing programs or processes.

[0049] The memory 140B may be implemented using any combination of any number of non-volatile memories (e.g., flash memory) and volatile memories (e.g., DRAM, SRAM), or any other non-transitory storage medium, or a combination of media.

[0050] Network interface 140C may be configured to connect to one or more communication networks to allow access operator computer 140 to communicate

with other entities such as terminal 132, gateway computer 150, etc. For example, communication with terminal 132 can be direct, indirect, and/or via an API.

[0051] Computer-readable medium 140D may comprise one or more non-transitory media for storage and/or transmission. Suitable media include, as
5 examples, a random access memory (RAM), a read only memory (ROM), a magnetic medium such as a hard-drive, or an optical medium such as a CD (compact disk) or DVD (digital versatile disk), flash memory, and the like. The computer-readable medium 140D may be any combination of such storage or transmission devices.

10 **[0052]** The computer readable medium 158 may comprise any suitable number of software modules. The software modules illustrated in FIG. 2 may include a token identification module 140D-1, an authorization request module 140D-2, an aggregation module 140D-3, and an encryption/decryption module 140D-4.

[0053] Token identification module 140D-1, in conjunction with the processor
15 140A, can identify tokens based on credentials. In some embodiments, identifying a token may comprise querying a token storage 155 with the credential to identify the associated token. In other embodiments, identifying the token may comprise generating the token with a cryptographic function, such as by hashing the credential. For example, the token may be generated by hashing the credentials. In
20 other embodiments, identifying the associated token may include obtaining a pre-generated token. The pre-generated token may be mapped to the credential in a data table in the token storage 155. The token identification module 140D-1, in conjunction with the processor 140A, may also determine if the token that is stored in the token storage 155 is valid. In this regard, conditions for token use (e.g.,
25 circumstances, times, dates, etc.) may be stored in a data table along with the credential that is associated with the token. For example, a token in the token storage 155 may only be valid for one day. The times (e.g., 12 am to 12 am the following day) that the token is valid may be stored in the token storage 155 with the token.

30 **[0054]** Authorization request module 140D-2, in conjunction with the processor 140A, can generate and send authorization request messages. The authorization request messages can be sent to processing network 170 (e.g.,

through transport computer 160). In some embodiments, the authorization request message can be a pre-authorization request message. In some embodiments, the authorization request message can be an ISO 8583 message.

- [0055]** Aggregation module 140D-3, in conjunction with the processor 140A, can aggregate transaction amounts. In some embodiments, aggregation module 140D-3 can form the total amount from a plurality of transaction amount at one time (e.g., when a clearing message is generated). For example, aggregation module 140D-3, in conjunction with the processor 140A, can query a transaction log for all transactions performed by a user then combine the individual transaction amounts.
- In other embodiments, aggregation module 140D-3 can aggregate each new transaction amount for a user to maintain a running total. Aggregation module 140D-3 can perform other calculations to determine a total amount from transaction amounts, such as discounts or other adjustments (e.g., a cap on the price, discounted fare after a certain time, etc.)
- [0056]** Encryption/decryption module 140D-4, in conjunction with the processor 140A, can encrypt and/or decrypt information. For example, encryption/decryption module 140D-4 may receive credentials that have been encrypted with a public key, and may then use an associated secret or private key to decrypt the encrypted credentials. Additionally, encryption/decryption module 140D-4, in conjunction with the processor 140A, can use a symmetric key to encrypt data. Additionally, encryption/decryption module 140D-4 may verify encrypted data (e.g., with a public key) and ensure that the encrypted data (e.g., signed data) has not been tampered with.

- [0057]** In some embodiments, the computer-readable medium 140D may comprise code, executable by the processor 140A, to perform a method comprising: receiving, from an access device of an access provider, a first transaction request for a first transaction, the first transaction request comprising a credential, the first transaction request associated with a first transaction amount; identifying a token, based upon the credential; transmitting, to an authorizing computer, a pre-authorization request message comprising the credential and a pre-authorization amount; receiving a pre-authorization response message from the authorizing computer; receiving, from the access device or another access device, a second

transaction request for a second transaction, the second transaction request comprising the credential, the second transaction request associated with a second transaction amount; identifying the token from the credential; determining that the token has been used at the access provider; forming a total amount by aggregating
5 at least the first transaction amount and the second transaction amount; and initiating transmitting a clearing message including the total amount.

[0058] FIG. 3 shows a block diagram of a terminal 132 according to embodiments. Similar diagrams may apply to terminal 134 and terminal 136, or other terminals operated by access provider 130. Terminal 132 may comprise a processor
10 132A that is in operative communication with a memory 132B, a network interface 132C, an input component 132E, a gate 132F, and a non-transitory computer-readable medium 132D. Computer-readable medium 132D may comprise an encryption module 132D-1, a token identification module 132D-2, a decision module 132D-3, and an authorization request module 132D-4. Terminal 132 may also
15 comprise a secure element (not pictured). The processor 132A, memory 132B, network interface 132C, and the computer readable medium 132D may have the same or different characteristics as the processor 140A, the memory 140B, the network interface 140C, and the computer readable medium 158 in the access operator computer 140 illustrated in FIG. 2. The descriptions of those components
20 can be incorporated herein.

[0059] Input component 132E may interface with a portable device to receive information from the portable device. Input component 132E may be a magnetic stripe reader, an NFC reader, optical reader, contactless elements, or other device that can receive data. In some embodiments, the input component 132E may be a
25 contact or contactless reader that can read data from a portable device through a short-range contactless communication medium.

[0060] Gate 132F may be a physical structure such as a bar or turnstile, door, or other physical barrier that can prevent unauthorized access to a secure location such as a train terminal. In some cases, the gate 132F may physically block a user
30 from entering a controlled area. In some embodiments, the gate 132F may be a light, speaker, or other signal emitter that can inform a user that the user is or is not

allowed to enter a secure location. For example, gate 132F may be an LED that can be red when access is denied and that can be green when access is permitted.

[0061] The computer readable medium 132D may comprise any suitable number of software modules. In FIG. 3, an encryption module 132D-1, a token
5 identification module 132D-2, a decision module 132D-3, and an authorization request module 132D-4 may be on the computer readable medium 132D.

[0062] Encryption module 132D-1, in conjunction with the processor 132A, may encrypt data. For example, encryption module 132D-1 may encrypt credentials received from a portable device, such as payment credentials. The encryption
10 module 132D-1 may comprise or be in communication with a secure element to perform the encryption. The encryption may be done with a cryptographic key that is injected into the terminal 132 and stored in the secure element. The key may be injected when the terminal 132 is being constructed.

[0063] Token identification module 132D-2 may be substantially similar to or
15 different than the token identification module 158A of FIG. 2, and the token identification module 132D-2 and the processor 132A can perform similar or different functions than the token identification module 140D-1 and the processor 140A.

[0064] Decision module 132D-3, in conjunction with the processor 132A, may determine whether a user should be allowed access. The decision module 132D-3,
20 in conjunction with the processor 132A, may check determine if a token is on a blacklist. If a token that is associated with a user is on a blacklist, then the user is denied access to the secure location by the terminal 132. The blacklist may be stored in the memory 132B. The decision module 132D-3 can also use a credential received by the input component 132E to query the blacklist. If the decision module
25 132D-3, in conjunction with the processor 132A, determines that the user should be denied entry to a secure location, the decision module 132D-3 and the processor 132A can send a notification to the gate 132F to prevent the user from entering. If the decision module 132D-3 and the processor 132A determine that the user is allowed access to the secure location, the decision module 132D-3 can send a
30 notification to the gate 132F to allow the user access to the secure location.

[0065] Authorization request module 132D-4, in conjunction with the processor 132A, can send authorization request messages. The authorization

request message may be a transaction request. This module can be similar to the authorization request module 158B in FIG. 2. The authorization request module 132D-4 and the processor 132A in the terminal 132 in FIG. 3 can perform similar or different functions as the authorization request module 140D-2 and the processor 5 140A in the access operator computer 140 in FIG. 2.

[0066] The computer-readable medium 132D may also comprise code executable by the processor 132A for performing a method comprising: receiving a credential from a portable device; identifying a token based upon the credential, determining that the token is not on a blacklist; admitting a user associated with the 10 portable device; and transmitting, to an access operator computer, a transaction request comprising the credential, the transaction request associated with a transaction amount.

[0067] FIG. 4 shows a swim lane diagram of a process according to embodiments. FIG. 4 illustrates a process in which a user (not shown in FIG. 4) 15 uses a portable device 120 to access a secure location such as a transit system at two different points in time.

[0068] In step S402, a user (not shown in FIG. 4) can tap their portable device 120 to terminal 132 at a first time during a particular day. The terminal 132 may be a transit terminal such as an access gate at a train station. For example, the user may 20 tap a transit payment card, a credit card, or their mobile phone to a contact point on the terminal 132. As another example, the user may swipe a transit card through a magnetic stripe reader on the terminal 132. The terminal 132 can receive a credential from the portable device 120, such as a PAN. The terminal 132 may also receive additional information such as a CVV (card verification value). The credential 25 may be associated with an account such as a payment account or a transit account.

[0069] In step S404, the terminal 132 can identify a token based upon the credential. In some embodiments, identifying the token may comprise generating the token with a cryptographic function, such as by hashing. In other embodiments, the terminal 132 can retrieve the token from the token storage 155. The token may be a 30 unique string of characters or numbers. In some embodiments, the token may be a payment token.

[0070] In some embodiments, after identifying the token, the terminal 132 may send the token to the token storage 155 with an identifier of the credentials to map the credentials to the token. The identifier for the credentials may be the credentials themselves, or it may be data derived from the credential (e.g., a hashed or encrypted credential). If it is the latter, the real credential may be converted to data derived from the credential each time the user uses the portable device 120 at a terminal of the access provider. This can be advantageous since sensitive data is not stored in the token storage 155.

[0071] In step S406, the terminal 132 can determine if the identified token is or is not on a blacklist. The blacklist may be stored on the terminal 132, and may be updated periodically (e.g., every 20 minutes with information from external sources, such as the access operator computer 140) as information about problematic tokens becomes available. For example, a user's token may be on a blacklist because their account did not have sufficient funds for a prior or current transaction or the account associated with the token was not valid for some reason. The terminal 132 may also determine if the credential is or is not on the blacklist. If the current transaction is the first transaction of the day conducted by the user, then a token for user may not be on the blacklist, but the user's credential could be on the blacklist.

[0072] In step S408, the terminal 132 can allow the user associated with the portable device 120 to access the secure location if the token is not on the blacklist. For example, the terminal 132 can admit the user by raising a barrier or unlocking a door. Alternatively, the terminal 132 can also admit the user by producing an audio or visual indication that the user is authorized to proceed to the secure location (e.g., by producing a green light). If the terminal 132 determines that the token is on the blacklist, then the terminal 132 can prevent the user from entering the secure location. The terminal 132 can do so by not moving a physical barrier such as a turnstile or by emitting a signal (e.g., a red light) indicating that the user cannot proceed to the secure location. In some embodiments, if this is the user's first transaction of the day (or any other periodic time period), the blacklist determination may be the only criteria used to decide if the user can access the secure location.

[0073] In step S410, the terminal 132 can transmit to an access operator computer 140, a transaction request comprising at least the credential obtained from

the portable device 120, and optionally a transaction amount and other data (e.g., a time stamp, a transaction identifier, etc.). For example, the transaction amount may be a transit fare or venue entrance fee. In some embodiments, the transaction amount may be associated with the transaction request, but may not be contained within the transaction request. For example, all transactions originating at terminal 132 may be for a flat fee of \$2.50 so a transaction amount need not be in the transaction request.

[0074] In this example, the transaction request may be a first transaction request, and prior to sending the credential to the access operator computer 140, the terminal 132 can encrypt the credential. In some embodiments, the credential may be encrypted with a cryptographic key stored on a secure element of the terminal 132. In some embodiments, the transaction request also comprises the token if the transaction request is not an initial transaction request for the user.

[0075] In step S412, the access operator computer 140 can identify a token based upon the credential. In some embodiments, the token may be in the transaction request. In other embodiments, the access operator computer 140 may generate the token with the credential (e.g., by hashing the credentials). In yet other embodiments, the access operator computer 140 can retrieve the token from the token storage 155. In such embodiments, the access operator computer 140 may use the credential to query the token storage 155, then receive the token in response.

[0076] In step S414, after identifying the token, the access operator computer 140 can transmit a pre-authorization request message to the authorizing computer 180 for approval. The pre-authorization request message can be used to determine if the credential is valid. The pre-authorization request message may not necessarily be used to pre-authorize a transaction amount for the transaction as is normally done for pre-authorization requests. In some embodiments, the pre-authorization request may comprise the credential and a preauthorization amount. In some embodiments, the pre-authorization amount may be a small amount that is less than the transaction amount (e.g., less than one dollar). In some embodiments, the pre-authorization amount may be 0 dollars. The pre-authorization request message may

also include a pre-authorization identifier that can identify the pre-authorization request that is being made.

[0077] In some embodiments, the authorizing computer 180 may be operated by an issuer of an account of the user. The access operator computer 140 may then
5 receive a pre-authorization response message from the authorizing computer 180. The pre-authorization response message may indicate whether the account is valid. For example, if the pre-authorization fails, even for a nominal pre-authorization amount, this may indicate that the user's account is invalid or fraudulent.

[0078] Pre-authorization may fail for a number of reasons. For example, the
10 credential may be evaluated by the authorizing computer 180 and may be determined to be invalid or on an issuer blacklist (e.g., due to excessive fraudulent use).

[0079] After the access operator computer receives the pre-authorization response, the access operator computer 140 may store the information from the pre-
15 authorization request (or any authorization request, or pre-authorization response) in a transaction log. Each entry in the transaction log (which may in the form of a data table) may comprise one or more of the pre-authorization identifier, the token, a time and date of the transaction, an identifier of the terminal 132, an identifier of the access provider, the pre-authorization amount, and/or the transaction amount. The
20 transaction log may also store the approval or disapproval indicator, or any other suitable data, from the pre-authorization response message. In some embodiments, the access operator computer 140 may discard the plaintext credentials, and may also discard any encrypted credentials.

[0080] In step S416, if the pre-authorization response message is negative
25 (including a decline), the access operator computer 140 may add the token (and optionally the credential) to the blacklist and send the updated blacklist to the terminal 132. The updated blacklist may be sent to all terminals operated by the access provider. In some embodiments, the updated blacklist may be sent to terminals operated by multiple access providers (e.g., a bus system and a subway
30 system in the same city). In some embodiments, the updated blacklist may be sent to the terminal 132 after the user has been admitted. In these embodiments, the user may be admitted to the secure location even if they do not have a valid account.

[0081] In some embodiments, the terminal 132 can then receive a second credential from a second portable device associated with a second user. The terminal 132 can then identify a second token based on the second credential, by a similar process as step S412. The terminal 132 may determine that the second token
5 is on the blacklist and deny access to the second user associated with the second portable device.

[0082] In step S418, the portable device 120 may be used to initiate a second transaction. For example, the user may use the portable device 120 to access a secure location provided by the access agency at a second time during the day. In
10 some embodiments, the portable device 120 may be used at the same terminal 132 (e.g., if there is only one access point at a venue). In other embodiments, the personal device may be used at a different terminal (e.g., at a subway turnstile at a different station from the first transaction). In the second transaction, the terminal 132 may receive the credential from the portable device 120.

[0083] Steps S420-S424 may be similar to steps S404-S408, are incorporated herein, and need not be repeated here.

[0084] In step S426, the access operator computer 140 can receive a second transaction request for a second transaction from the terminal 132. The second transaction request can comprise the credential and a second transaction amount. In
20 some embodiments, the second transaction amount may be associated with the second transaction request but not contained within the second transaction request. The second transaction amount may or may not be equal to the first transaction amount. In some embodiments, the second transaction request may also include the token. In some embodiments, the terminal 132 may encrypt the credential and/or the
25 token prior to sending it to the access operator computer 140. In some embodiments, the credential and/or token may be encrypted with a cryptographic key stored on a secure element of the terminal 132.

[0085] In step S428, the access operator computer 140 can identify the token from the credential. In some embodiments, the token may be in the transaction
30 request and the access operator computer 140 can identify the token by analyzing the transaction request message. In other embodiments, the access operator computer 140 may generate the token with the credential (e.g., by hashing the

credentials). Alternatively, the access operator computer 140 can retrieve the token from token storage 155. The access operator computer 140 may use the credential to query the token storage 155, then receive the token in response. As noted above, in some cases, the credential may be hashed or encrypted, and the token may be
5 retrieved using the hashed or encrypted credential.

[0086] The access operator computer 140 can then determine whether the token has been used at the access provider. In some embodiments, the access operator computer 140 can determine whether the token has been used that day. Determining whether the token has been used may comprise searching the
10 transaction log. If the access operator computer 140 stores a transaction log entry including the token for that day, then the token has been used.

[0087] If the token has been used, the access operator computer 140 may not send a pre-authorization message to the authorizing computer 180. The token, a time and date of the second transaction, an identifier of the terminal 132, an identifier
15 of the access provider, and/or the second transaction amount associated with the second transaction can be added to the transaction log by the access operator computer 140. In some embodiments, the first transaction request in the transaction log can be updated with information from the second transaction request.

[0088] If the token is currently in a valid state and there is nothing that would
20 prevent the user from accessing the secure location, then the user of the portable device 120 may be allowed to access the secure location, even though no pre-authorization request is sent to the authorizing computer 180 and even though no authorization request message is sent to the authorizing computer 180.

[0089] At this point, the user of the portable device 120 may continue to enter
25 the secure location provided by the access agency for any other number of times. The limit on the number of times may be restricted by time (e.g., one day). In this case, the token stored for the credential may need to be refreshed and the process needs to start from the beginning (e.g., at step S402) at the start of the new time period (e.g., the next day).

[0090] In step S430, in some embodiments, after the requisite period of time
30 has elapsed (e.g., 1 day) or after the allowed number of transactions is conducted by the user of the portable device 120, the access operator computer 140 can generate

and submit a single authorization request message to the authorizing computer 180 (e.g., via the transport computer 160 and the processing network 170) to request authorization for the total amount of all transactions conducted by the user during the predetermined period of time. The authorization request message may comprise a total amount of the transactions, the pre-authorization request identifier, and optionally, the token. The pre-authorization request identifier may also have sufficient information to identify the authorizing computer 180. For example, the pre-authorization request identifier may comprise an identifier (e.g., a bank identification number) for the authorizing computer 160.

10 **[0091]** The access operator computer 140 can form the total amount by aggregating at least the first transaction amount and the second transaction amount. This may be done periodically, such as once per day before sending clearing messages. Alternatively, the total amount may be aggregated every few hours to maintain a running total. When forming the total amount, other factors may affect the total amount in addition to the first transaction amount and the second transaction amount. For example, there may be a daily cap on the total amount (e.g., subway trips are free after spending \$10 in one day) or a volume discount (e.g., a 10% discount on all rides after taking 5 or more trips). In order to aggregate the total amount, the access operator computer 140 can retrieve at least the data associated with the first transaction and the second transaction from the transaction log.

20 **[0092]** The authorizing computer 180 can use the pre-authorization identifier to retrieve the pre-authorization request message and the credential (e.g., real PAN) associated with the pre-authorization request message stored during the pre-authorization process, such as by extracting the credential from the pre-authorization request message.

[0093] The authorizing computer 180 can then use the total amount and credentials to complete the transaction. For example, the credential can be used to determine the user's account and to determine if there are sufficient funds and credit in the user's account to pay for the total amount.

30 **[0094]** In step S432, in addition to or instead of the authorization request message described above with respect to step S430, the access operator computer 140 can initiate the transmission of a clearing message. For example, the access

operator computer 140 can transmit a clearing message or communicate with the transport computer 160 associated with the access operator operating the access operator computer 140 to cause the transmission of a clearing message including the total amount and the pre-authorization identifier to the processing network 170.

- 5 The clearing message may then be received by the processing network 170, which may in turn communicate with the authorizing computer 180 in step S434. The authorizing computer 180 can use the pre-authorization identifier to retrieve the pre-authorization request message and the credential (e.g., real PAN) associated with the pre-authorization request message stored during the pre-authorization process,
10 such as by extracting the credential from the pre-authorization request message.

[0095] The authorizing computer 180 can then use the total amount and credentials to complete the transaction. For example, the credential can be used to determine the user's account and to determine if there are sufficient funds and credit in the user's account to pay for the total amount.

- 15 **[0096]** The authorizing computer 180 may also verify that a clearing message has not already been received with the pre-authorization identifier to prevent double charging. If the authorizing computer 180 cannot locate a pre-authorization request message, the authorizing computer 180 may return an error message or a decline to the access operator computer 140. Authorizing computer 180 may also return an
20 error message or decline if it is unable to complete the transaction (e.g., invalid account, insufficient funds). If the access operator computer 140 receives an error message or a decline, the access operator computer 140 may add the credentials and/or tokens associated with the credentials to the blacklist and send the updated blacklist to terminal 132 and all other terminals.

- 25 **[0097]** After step S432, the processing network 170 can initiate a transfer of funds from the authorizing computer 180 to the transport computer 160 which holds the account of the access operator operating the access operator computer 140.

- [0098]** The access operator computer 140 may periodically attempt to complete the outstanding transactions of tokens on the blacklist. For example, once
30 per day, the access operator computer 140 can generate a clearing message for a token on the blacklist along with the outstanding total amount and the pre-authorization identifier, and transmit it as in step S432. This clearing message may

be for a recurring transaction, repeating the attempt to finish the transaction until the access operator computer 140 receives a successful transaction response message. If the access operator computer 140 receives an approval response, the access operator computer 140 can remove the token from the blacklist.

- 5 **[0099]** Additionally, or alternatively, a user can have themselves removed from the blacklist. For example, the user can go onto a website associated with the access operator computer or access provider. The user can log in (e.g., by entering a username and password, by entering their credentials) and view failed transactions. After logging in, the user may then be able to address any failed transactions. For example, the user can update their credential to a valid account or enter an alternate credential for paying the outstanding debt. The site can then generate a debt recovery message including the credential and send it to the access operator computer. After receiving the debt recovery message, the access operator computer can retrieve (and/or generate) the total amount and transmit the total amount to the website. The website can then generate a debt recovery request comprising the updated credential and the total amount and send it to the authorizing computer. The debt recovery message may be sent through the access operator computer, the payment gateway and/or the processing network.

- 20 **[0100]** The data associated with the debt recovery message may also be used to generate a clearing message. The website can then receive a debt recovery response from the authorizing computer, indicating whether or not the transaction was successfully completed. After receiving an approval response from the site, the access operator computer can remove the credential and associated token from the blacklist, then send the updated blacklist to the terminals.

- 25 **[0101]** Embodiments of the invention provide a number of advantages. Embodiments can protect sensitive credential information (e.g., secure access credentials, transit credentials, payment credentials) using a combination of security mechanisms. Credentials can be encrypted when they are transmitted between terminals and access operators, or transmitted over secure channels to a processing network. Plaintext credentials need not be stored at the terminal or access operator computer, reducing the risk that a hacker gain access to a large database of sensitive data. Additionally, embodiments can be implemented without requiring

significant changes to an existing infrastructure. A city's transit system may have hundreds of turnstiles and needing to update each and the existing transit infrastructure may be prohibitive

[0102] Any of the software components or functions described in this
5 application, may be implemented as software code to be executed by a processor
using any suitable computer language such as, for example, Java, C++ or Perl
using, for example, conventional or object-oriented techniques. The software code
may be stored as a series of instructions, or commands on a computer readable
medium, such as a random access memory (RAM), a read only memory (ROM), a
10 magnetic medium such as a hard-drive or a floppy disk, or an optical medium such
as a CD-ROM. Any such computer readable medium may reside on or within a
single computational apparatus, and may be present on or within different
computational apparatuses within a system or network.

[0103] The above description is illustrative and is not restrictive. Many
15 variations of the invention may become apparent to those skilled in the art upon
review of the disclosure. The scope of the invention can, therefore, be determined
not with reference to the above description, but instead can be determined with
reference to the pending claims along with their full scope or equivalents.

[0104] One or more features from any embodiment may be combined with
20 one or more features of any other embodiment without departing from the scope of
the invention.

[0105] A recitation of "a", "an" or "the" is intended to mean "one or more"
unless specifically indicated to the contrary.

[0106] All patents, patent applications, publications, and descriptions
25 mentioned above are herein incorporated by reference in their entirety for all
purposes. None is admitted to be prior art.