(12) STANDARD PATENT APPLICATION (11) Application No. **AU 2017279652 A1**
(19) AUSTRALIAN PATENT OFFICE

(54) Title
Homomorphic Passcode Encryption

(51) International Patent Classification(s)
G06Q 20/38 (2012.01) ***G06Q 20/34*** (2012.01)
G06Q 20/32 (2012.01) ***G06Q 20/40*** (2012.01)

(21) Application No: **2017279652** (22) Date of Filing: **2017.12.20**

(43) Publication Date: **2018.01.18**
(43) Publication Journal Date: **2018.01.18**

(62) Divisional of:
2015217346

(71) Applicant(s)
Square, Inc.

(72) Inventor(s)
Quigley, Oliver S.C.;Waddle, Jason Douglas;Adida, Benjamin Michael;Guise, Max Joseph

(74) Agent / Attorney
EAGAR & MARTIN PTY LTD, P.O. Box 1499, Oxenford, QLD, 4210, AU

Abstract

A method of encrypting a passcode is disclosed. In one embodiment, the method includes: receiving an indication of a portion of the passcode; calculating a plaintext value based at least in part on the indication, wherein the plaintext value represents an encoded portion of the passcode; encrypting the plaintext value into ciphertext using a homomorphic encryption system; and updating a cumulative encryption string by executing a cumulative operation to aggregate the ciphertext corresponding to the encoded portion into the cumulative encryption string computed for a previous portion of the passcode, wherein the cumulative operation is dictated by a homomorphic property of the homomorphic encryption system.

HOMOMORPHIC PASSCODE ENCRYPTION

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application claims the benefit of U.S. Provisional Patent Application No. 61/938,495, filed February 11, 2014, which is incorporated by reference herein in its entirety.

[0002] This application claims the benefit of U.S. Patent Application No. 14/209,381, filed March 13, 2014, which is incorporated by reference herein in its entirety

BACKGROUND

[0003] A financial transaction system, as used herein, is a system that allows a merchant to use, for example, a mobile device, to accept payment for selling a product, a service, or a rental to a purchaser.

[0004] In one example, the financial transaction system includes a mobile device (*e.g.,* a tablet computer, a smartphone, etc.) and a card reader. The card reader is in the form of an accessory and couples to the mobile electronic device (*e.g.,* the card reader couples to the mobile device through the audio jack of the mobile device). In this example, a purchaser uses a financial transaction card (*e.g.,* a credit card, a debit card, a pre-paid gift card, etc.) to purchase the seller's product or service by allowing his/her credit card to be swiped through the card reader. The card reader communicates the card's data to the mobile device, allowing the mobile device to confirm the authenticity of the card and further to initiate authorization of the purchase transaction. In another example, the financial transaction system may include a mobile device that accepts card-less payments from purchasers. In this example, a purchaser may convey his/her credit card information to the seller through a direct or indirect form of wireless communication with the seller's mobile device. A person of any skill in this space would easily be aware of countless other mechanisms that allow similar financial transactions to proceed in the context of such "mobile" payments.

[0005] While such mobile payment opportunities offer convenience and ease of use to both the seller and the purchaser, there are scenarios that may present new security concerns. For example, as part of the transaction flow, the purchaser may sometimes be required to enter a PIN code as an additional layer of security. Such PIN codes are required, for example, in debit card-based purchases and even in some credit card-based (*e.g.,* Europay, MasterCard, Visa (EMV) card-based) purchases. In such scenarios, the financial transaction system needs to protect the PIN from being discovered by, for example, malware or other phishing events.

Brief Description of the Drawings

[0006] **FIG. 1** is a control flow diagram illustrating a technique for encrypting a passcode utilizing homomorphic cryptography.

[0007] **FIG. 2** is a block diagram illustrating an example of a passcode verification system including a passcode entry device and an authentication device.

[0008] **FIG. 3** is a diagram illustrating an example of a timeline for encrypting a sequence of symbols representing a passcode.

[0009] **FIG. 4** is a flow chart of a process to encrypt a passcode using homomorphic cryptography.

[0010] **FIG. 5** is a flow chart of a process to authenticate a passcode entry based at least in part on an encrypted message.

[0011] **FIG. 6** is a block diagram of an example of a financial transaction system including an electronic device and a card reader.

[0012] The figures depict various embodiments of the present invention for purposes of illustration only. One skilled in the art will readily recognize from the following discussion that alternative embodiments of the structures and the methods illustrated herein may be employed without departing from the principles of the disclosure described herein.

Detailed Description

[0013] The financial transaction system described herein involves a card reader and a general-purpose electronic device, such as a mobile device or a stationary/semi-stationary system. For example, the mobile device may be a mobile phone or a tablet computer and the stationary/semi-stationary system may be a point-of-sale system or a desktop computer. The term "card reader" here refers to any object that can be used to obtain information from an object used to make an electronic payment where the object must be in the general vicinity of the object, such as an optical scanner, a near field communications device, a Bluetooth communications device, etc. The card reader may be an external device (*e.g.,* in the form of a mobile phone accessory) that can be coupled to the general-purpose electronic device. When the card reader detects a financial transaction card (based on, for example, the card being swiped through the card reader, the card being brought in proximity to enable radio frequency (RF), Near Field Communication (NFC), or Bluetooth Low Energy (BLE) communication between the card and the card reader, etc.), the card reader retrieves card information stored on the financial transaction card to initiate a financial transaction through the general-purpose electronic device. The term financial transaction card refers to any object that can be used to make an electronic payment, such as a mobile device via a digital wallet application, an object containing an optical code such as a quick response (QR) code, etc. The financial transaction card can store financial account related data to be used in a financial transaction, such as to make a purchase. For example, the financial transaction card may be a credit card, a debit card, or an integrated circuit card in accordance with the EMV standard.

[0014] In some instances, a user (*i.e.,* a purchaser) initiating the financial transaction may need to enter a passcode, such as a personal identification number (PIN) or a password, on a passcode entry interface to authenticate and/or authorize the financial transaction. The passcode entry interface may solicit the user to identify a sequence of symbols/digits representing the passcode. As the user enters the passcode, the electronic device can encrypt the passcode to protect against discovery of the passcode by a malicious third party.

[0015] One security consideration in designing a passcode entry system is whether a memory snapshot of the passcode entry device can compromise the passcode entered by the user. The disclosed technique involves a method of encrypting portions (*e.g.,* characters, digits or alphabets) of the passcode entered by the user separately, and accumulating the encrypted portions into a cumulative encryption string (*e.g.,* a single encrypted digital string). Hence, the disclosed technique protects the data flow through the memory of the passcode

entry device such that any given memory snapshot can reveal at most a single portion (*e.g.,* a character, a digit, or an alphabet) of the passcode.

[0016] The cumulative encryption string can be represented as a tuple, including a cumulative nonce product and a cumulative message string. The cumulative encryption string can be decrypted with a single set of decryption keys. That is, the decryption process needs only be run once to derive the unencrypted plaintext, even though the cumulative encryption string embodies multiple encrypted portions. The electronic device can capture a sequence of user inputs indicative of a sequence of symbols representing the passcode via an input component, such as buttons or a touchscreen. The card reader or an external server coupled to the electronic device can then authenticate and/or authorize the financial transaction by verifying the passcode entered by the user. Conventionally, as a user (*e.g.,* purchaser) enters the passcode, an unencrypted digital representation of the entire passcode may sometimes have to be stored on a memory device(s) of the electronic device. The disclosed technique, instead, encrypts each portion of the passcode separately to prevent discovery of the entire passcode by malicious entities. The electronic device can encrypt each portion of the passcode as the portion is entered or determined, such that no memory device of the electronic device ever stores the entire unencrypted passcode.

[0017] A challenge associated with encrypting multiple portions of a passcode is that delivery of multiple encrypted files puts a burden on the network used for delivery. Further, decryption of the multiple encrypted files is memory and processor intensive and thus may necessitate more memory and computing resources at the receiving device. The disclosed technique overcomes these challenges by encrypting each portion (*e.g.,* a symbol or digit) of the passcode using a homomorphic cryptosystem and aggregating the encrypted portions into a master encryption file (*i.e.,* the cumulative encryption string). The cumulative encryption string includes only a single layer of encryption despite including multiple separately encrypted portions. The electronic device can utilize the homomorphic property of the homomorphic cryptosystem to combine multiple encrypted portions into a final cumulative encryption string, thus reducing the burden on the network and the receiver-side memory and processor. The disclosed technique enables a single encrypted file to be sent from the electronic device to the card reader without having to send multiple encryption files for each symbol/digit of the passcode. Further, the electronic device can encrypt each individual symbol/digit as the user enters the symbol/digit, thus preventing discovery of the passcode by a malicious third party.

[0018] Homomorphic encryption is a form of encryption that allows specific type(s) of computational operation(s) to be carried out on ciphertexts (*i.e.,* results of encryption) and obtain an encrypted result which, when decrypted, matches the result of the same computational operation(s) performed on the plaintext (*i.e.,* inputs to the encryption). The specific type of computational operation may be referred to as the homomorphic property of the homomorphic cryptosystem. A homomorphic encryption scheme may preserve the homomorphic property for only multiplication, only addition, or for both addition and multiplication. For example, the unpadded RSA and the ElGamal are cryptosystems that preserve the homomorphic property for multiplication. A cryptosystem refers to a suite of mechanisms/processes for implementing a particular form of encryption and decryption, including, for example, a key generation mechanism, an encryption mechanism, and a decryption mechanism.

[0019] In at least one embodiment, the electronic device may implement a homomorphic encryption process, such as according to the ElGamal cryptosystem, that preserves the homomorphic property for multiplication. Under the ElGamal cryptosystem, in a group G, if the public key is (G, q, g, h), where $h = g^x$, x is the private key, and r is a random variable selected from the set $\{0, \ldots, q\text{-}1\}$, then the encryption of a message m is $\varepsilon(\mathrm{m}) = (g^r, m \cdot h^r)$. The homomorphic property can be illustrated as $\varepsilon(m_1) \cdot \varepsilon(m_2) = (g^{r1}, m_1 \cdot h^{r1})\,(g^{r2}, m_2 \cdot h^{r2}) = (g^{r1+r2}, m_1 \cdot m_2 \cdot h^{r1+r2}) = \varepsilon(m_1 \cdot m_2)$.

[0020] The electronic device can encode each portion (*e.g.,* a symbol or digit) of the passcode as a plaintext value, before encrypting the plaintext value into ciphertext. For example, the electronic device can encrypt a plaintext value calculated based in part on a numeric representation of each symbol/digit with the ElGamal cryptosystem. As a specific example, the plaintext value can be a prime number raised to the power of the digit or to the power of a numeric representation of the symbol. Each ciphertext outputted by the cryptosystem is multiplied together with a cumulative encryption string that is the product of all previous ciphertexts produced from the previous symbols or digits. This product is then updated as the new cumulative encryption string. In this manner, instead of producing a sequence of ciphertexts (*e.g.,* $\varepsilon(m_1)$, $\varepsilon(m_2)$, ...) from a sequence of symbols or digits, the electronic device only has to upkeep a cumulative master encryption string (*e.g.,* $\varepsilon(m_1) \cdot \varepsilon(m_2) \cdot \varepsilon(m_3) \cdot \ldots$). Because of the homomorphic property of the ElGamal cryptosystem, this cumulative encryption string is identical to the ciphertext resulting from encrypting the product of the plaintext values associated with each symbol or digit, *e.g.,* $\varepsilon(m_1 \cdot m_2 \cdot m_3 \cdot \ldots)$.

[0021] The disclosed technique can encode the portions of the passcode as powers on small primes. In particular, if the passcode (*e.g.,* represented as a sequence of digits) is denoted by a vector $\mathbf{d} = [d_1, d_2, ..., d_M]$, then the encoded portions may be denoted as $\mathbf{EPs} = [p_1^{d\_1}, p_2^{d\_2}, ..., p_M^{d\_M}]$, where the p_i denotes a set of prime bases and M denotes the number of portions in the passcode. The set of prime bases can be chosen in accordance with various security restrictions. For example, the set of prime bases may include small primes for ease of calculation.

[0022] The set of prime bases may be chosen for semantic security. For example, under embodiments utilizing the ElGamal cryptosystem or other similar cryptosystems, the prime bases can be chosen to have order q instead of $2q$. Here, "q" denotes the public encryption parameter for the ElGamal cryptosystem. Under the ElGamal cryptosystem, the modulus **N** used for the encryption mechanism is equal to $2q+1$. An order (n) of a prime basis (p) is the number of elements in a cyclic group of modulo **N** for exponents of the prime basis. That is, the order is the smallest number n such that $p^n = e$, where "e" denotes the identity element in the cyclic group. For example, using "2" as one of the prime bases for encoding may lead to a leak of information about the associated passcode portion. A malicious third party can observe that the order of the resulting encoding depends on the parity of the exponent. Exploiting the Legendre symbol of the message, the malicious third party can reverse-engineer the parity of the passcode portion. If the passcode portion is a digit, the malicious third party can determine the parity of the digit through this exploit. To combat this issue, the set of prime bases can be chosen to have order q. That is, every prime number in the set satisfies $p_i^q \equiv 1 \pmod N$. This holds, for example, for the small prime "3" and many other prime numbers.

[0023] The set of prime bases can be chosen by an external server system along with other static cryptography parameters of the homomorphic cryptosystem that are accessible by both the encryption-side device (*e.g.,* the electronic device) and the decryption-side device (*e.g.,* the card reader). The set of prime bases can be stored in a static file maintained by the external server system. Alternatively, multiple sets of potential prime bases may be associated with particular decryption side devices, such as by hardware identifications of such devices. In either case, the electronic device can receive the set of prime bases and/or the public key for encryption by requesting such parameters from the external server system. Alternatively, the set of prime bases and/or the public key may be part of a mobile application downloaded and installed onto the electronic device.

[0024] It is noted that whilst portions of the passcode that are encoded and encrypted have been illustrated as individual symbols or digits of the passcode, the portions may be partitioned in other ways as well. For example, the portions can be pairs of symbols or digits. The portions can further be values indicative of the symbols/digits that represent the passcode. For example, where the passcode is determined based on touch events on the passcode entry interface, the electronic device can encode and encrypt the (X,Y) coordinates of each touch separately. In some embodiments, the electronic device can encrypt the portions directly without first encoding the portions or values indicative of the portions.

[0025] After the cumulative encryption string captures every portion of the passcode entered by the user, the electronic device sends the cumulative encryption string in a message to verify the passcode. The electronic device can send a message including the cumulative encryption string to an external server or the card reader to verify the passcode. The card reader or the external server can access a decryption key consistent with the encryption key used in the homomorphic cryptosystem implemented on the electronic device. The card reader or the external server can decrypt the cumulative encryption string to determine the product of the encoded portions of the passcode entered by the user.

[0026] The card reader or the external server can verify the user passcode entry against an authentic passcode in the financial transaction card in various manners. For example, the card reader can access the financial transaction card via a read head to determine the authentic passcode associated with the purchaser's financial account. The card reader or the external server can factor the encoded portions of the passcode (*e.g.,* using the set of prime numbers that is used to encode) to determine the passcode entry and use the passcode entry to verify against the authentic passcode. As another example, the card reader can send the decrypted and decoded user passcode entry to the financial transaction card for verification against the authentic passcode.

[0027] In other embodiments, the card reader or the external server can calculate a comparison value based at least in part on the authentic passcode to verify the passcode. For example, the comparison value can be a product of prime numbers, each raised to the power of respective digits in the authentic passcode. The prime numbers used here are the same prime numbers used to calculate the plaintext values (*i.e.,* the encoded portions) during the encryption process. In this way, the card reader or the external server can compare the comparison value against the decrypted cumulative encryption string. When both values match or when both values maintain an equivalent mathematical property, the card reader or the external server can authorize and authenticate the financial transaction.

[0028] **FIG. 1** is a control flow diagram illustrating a technique for encrypting a passcode utilizing homomorphic cryptography. A user can enter a passcode on an electronic device 102, such as the PIN digits "8144" as shown. The electronic device 102 may be a general-purpose electronic device, such as a mobile device or a stationary device. The electronic device 102 may include and execute a general-purpose operating system capable of running one or more third-party applications that may be downloaded, installed, and executed on the electronic device 102.

[0029] For example, the electronic device 102 provides a passcode entry interface 104 to enable the user to identify a sequence of symbols representing the passcode. The sequence of symbols may include numeric digits, alphabetical characters, alphanumeric characters, pictograms, icons, other types of enumeration, or any combination thereof. Each symbol can have a numeric representation, such as a binary representation, that can be used for encryption.

[0030] The passcode entry interface 104 may be implemented as a user interface displayed on a touchscreen. The electronic device 102 generates and displays the passcode entry interface 104 when the electronic device 102 needs to authenticate a financial transaction or other types of user operations. Alternatively, the passcode entry interface 104 may instead be implemented with a set of physical buttons.

[0031] As the user enters the passcode, the electronic device 102 can separately encrypt portions of the passcode utilizing a homomorphic cryptography system 106. The homomorphic cryptography system 106 is a processing module of the electronic device 102 capable of encrypting datasets into ciphertexts according to a homomorphic cryptosystem, such as the ElGamal cryptosystem. The ciphertexts generated by the homomorphic cryptography system 106 maintain a homomorphic property. The homomorphic cryptography system 106 may be implemented as an application specific integrated circuit or as instructions executable by a processor.

[0032] The portions to encrypt can correspond to the individual symbols that make up the passcode. The portions can also correspond to subsets of the symbols. Alternatively, the portions to encrypt can be datasets that are used, at least partially, to determine the individual symbols. For example, the portions can be locations on the passcode entry interface where a touch event has been detected. The electronic device 102 can encrypt each portion whenever each portion is determined. For example, the encryption of a portion can occur in response to identifying a symbol entered by the user or to detecting a touch event on the passcode entry interface 104.

[0033] In some embodiments, the electronic device 102 can first encode each portion into a plaintext value and then encrypt the plaintext value corresponding to such portion. For example, the electronic device 102 calculates the plaintext value based at least partly on raising the power of a prime number to a numeric value of the portion (*i.e.,* multiplying the prime number for a number of times equal to the numeric value of the portion). In the example shown in **FIG. 1**, the ciphertexts of the encoded plaintext values are $[p_1^8]$, $[p_2^1]$, $[p_3^4]$, and $[p_4^4]$, wherein the bracket denotes the encryption and "$\mathbf{p}_i$" denotes the prime number used for each sequential position of the encoded portions (*e.g.,* "$\mathbf{p}_1$" being the prime number used for the first portion and "$\mathbf{p}_2$" being the prime number used for the second portion, etc.).

[0034] For each portion, the electronic device 102 encrypts the encoded plaintext and aggregates the resulting ciphertext into a cumulative encryption string 108. The cumulative encryption string 108 is a ciphertext file, which aggregates all previously generated ciphertexts corresponding to previously encrypted portions of the passcode. How the electronic device 102 aggregates the ciphertexts depends on the homomorphic property of the homomorphic cryptography system 106. For example, if the homomorphic property preserves multiplication, the ciphertexts are aggregated as a product; and if the homomorphic property preserves addition, the ciphertexts are aggregated as a summation. As each new ciphertext is generated, the homomorphic cryptography system 106 updates the cumulative encryption string 108 by multiplying or adding the previous instance of the cumulative encryption string 108 with the new ciphertext.

[0035] The passcode entry interface 104 and the homomorphic cryptography system 106 may both be part of a mobile application (commonly referred to as an "app") that is downloaded and installed onto the electronic device 102. The mobile application is stored in a memory device (not shown), such as a volatile or a non-volatile storage medium, and can be executed by a processor (not shown) of the electronic device 102.

[0036] After the user enters the entire passcode, the homomorphic cryptography system 106 would have aggregated ciphertexts corresponding to every portion or encoded portion of the passcode into the final cumulative encryption string 108. The electronic device 102 can then send the final cumulative encryption string 108 to a passcode verification system 110 in a message 112. The passcode verification system 110 is a device capable of verifying the authenticity of the passcode entered by the user (henceforth referred to as the "passcode entry"). The passcode verification system 110 has access to an authentic version

of the passcode (henceforth referred to as the "authentic passcode"). The passcode verification system 110 can use the authentic passcode to verify the passcode entry.

[0037] In some embodiments, the passcode entry and the passcode encryption processes are initiated by the passcode verification system 110. For example, the passcode verification system 110 may be a card reader coupled to the electronic device 102. The card reader can initiate a financial transaction upon detecting a swipe of a payment card, such as a credit card, a debit card, an EMV card (a payment card in accordance with the Europay, MasterCard, Visa standard), etc. The card reader can send the card information of the payment card to the electronic device 102. The electronic device 102 can then try to authenticate the financial transaction by requesting the user to enter the passcode through the passcode entry interface 104.

[0038] The passcode verification system 110 extracts the cumulative encryption string 108 from the message 112 and decrypts the cumulative encryption string 108 to determine an accumulated value indicative of a sequence of symbols that represents the passcode entered by the user.

[0039] The accumulated value can uniquely identify the passcode entry. In some embodiments, the accumulated value can be factored using the set of prime numbers used to calculate the encoded portions of the passcode to determine the sequence of symbols. The sequence of symbols can then be used to verify against an authentic passcode associated with the financial transaction card. In other embodiments, to verify the passcode entry, the passcode verification system 110 can calculate a comparison value based on the authentic passcode associated with the financial transaction card in the same manner that the accumulated value is calculated. That is, the passcode verification system 110 can encode portions of the authentic passcode in the same encoding process as the portions of the passcode entry, and can accumulate the encoded portions of the authentic passcode using the same cumulative operation based on the homomorphic property of the homomorphic cryptography system 106.

[0040] For example, the passcode verification system 110 can calculate the comparison value by taking the powers of the same prime numbers used by the homomorphic cryptosystem 106 to the numeric digits of the authentic passcode. The passcode verification system 110 can then verify the passcode entry by comparing whether the accumulated value and the comparison value match each other. In other embodiments, the passcode verification system 110 can determine the passcode entry from the accumulated value (*e.g.*, by decrypting

and decoding the accumulated value as described in disclosure), and verify the determined passcode entry against the known authentic passcode in the financial transaction card.

[0041] **FIG. 2** is a block diagram illustrating an example of a passcode verification system 200 including a passcode entry device 202 and an authentication device 204. The passcode verification system 200 may be consistent with various other embodiments described in this disclosure. The passcode entry device 202 is illustrated as a mobile device with a touchscreen, but may be implemented as another type of electronic device with computing capabilities. For example, the passcode entry device 202 may be the electronic device 102 of **FIG. 1**. The authentication device 204 is illustrated as a card reader, but may be implemented also as another type of electronic device with computing capabilities. For example, the authentication device 204 may be the passcode verification system 110 of **FIG. 1**.

[0042] The authentication device 204 includes logic circuitry 206. The logic circuitry 206 controls and executes processes operated by the authentication device 204. The logic circuitry 206 may comprise one or more of an application-specific integrated circuit (ASIC), field programmable gate array (FPGA), a controller, a microprocessor, and other types of digital and/or analog circuitry.

[0043] The logic circuitry 206 can communicate with a read head 208. The read head 208 can detect a financial transaction card, such as detecting the card being swiped or the card entering the proximity of the read head 208 (*e.g.,* via RF or BLE). The read head 208 may be a magnetic strip reader, an NFC smart card, a radiofrequency identification (RFID) reader, a Bluetooth reader, a radio frequency receiver, an optical reader, or any combination thereof.

[0044] In various embodiments, the logic circuitry 206 may include a signal converter 214, a signal processor 216, and a security module 218 to facilitate communication with the read head 208. The signal converter 214 is circuitry that converts analog readings from the read head 208 into digital data. The signal processor 216 is circuitry for processing and interpreting the digital data from the signal converter 214 into card information, representative of financial account information of a user. Optionally, the logic circuitry 206 may include the security module 218 to encrypt and/or decrypt information to and from the financial transaction card or otherwise provide secure access to a financial transaction card via the read head 208.

[0045] The logic circuitry 206 can communicate with the passcode entry device 202 via a first connection interface 220. The first connection interface 220 is adapted to

communicate with the passcode entry device 202, such as via a wired connection (*e.g.*, an audio cable or a bus) or wirelessly (*e.g.*, RF or Bluetooth communication). The logic circuitry 206 can initiate a financial transaction with the passcode entry device 202 through the first connection interface 220 when the read head 208 detects the financial transaction card. The logic circuitry 206 can send a message to the passcode entry device 202 through the first connection interface 220 to initiate a process to authenticate and authorize the financial transaction. In some embodiments, the message can cause the electronic device to generate and display a passcode interface.

[0046] The passcode entry device 202 includes a second connection interface 222 that couples to the first connection interface 220 of the authentication device 204 directly or indirectly, including via a wired connection or a wireless connection. The passcode entry device 202 includes a transaction manager module 224 to facilitate authorization of the financial transaction. The transaction manager module 224 can manage identity of goods, services, and/or rentals associated with the financial transaction. The transaction manager module 224 can process messages received via the second connection interface 222, including the message to initiate the process to authenticate the financial transaction. The transaction manager module 224 can notify a passcode interface module 226 to solicit a user to enter a passcode in response to the message. For example, the passcode interface module 226 can generate and display a passcode entry interface on a touchscreen 228. The passcode entry interface may include buttons labeled with symbols (*e.g.*, digits, characters, icons, etc.) that the user may interact with to indicate a sequence of symbols representing the passcode. The passcode interface module 226 monitors the touchscreen 228 and records locations of any "touch" events to determine the sequence of symbols.

[0047] The passcode entry device 202 includes an encryption module 230 to implement a piecemeal encryption of portions of the passcode using a homomorphic cryptosystem. The encryption module 230 can encrypt the portions sequentially as the user enters each portion. For example, each portion can correspond to a touch event or a set of touch events from the user. Each portion can also correspond to a symbol in the passcode entry.

[0048] In some embodiments, the encryption module 230 can encode each portion of the passcode as a plaintext value. That is, as each portion is identified, the encryption module 230 can calculate the plaintext value based at least partly on the identified portion and encrypt the plaintext value into ciphertext utilizing a homomorphic cryptosystem, such as with the ElGamal asymmetric encryption. In one example, the plaintext values may just be

numeric representations (d_i) of the symbols of the passcode. Alternatively, a hash function or an encoding process can be used to determine the plaintext value from the numeric representation of each portion. For example, the encryption module 230 can calculate the plaintext value for each symbol by selecting a prime number from a set of prime bases (p_i) and calculating the plaintext value by raising the small prime to the power represented by the numeric representation of the symbol. In this example, the plaintext is equal to $p_i^{d\_i}$. The encryption module 230 can select the small prime based at least in part on a sequential counter of the symbol entered, such as selecting p_1 for the first portion entered by the user and selecting p_2 for the second portion entered by the user.

[0049] The encryption module 230 then "accumulates" the ciphertexts generated for the portions into a cumulative encryption string 232 by means of a cumulative operation, *e.g.*, the cumulative encryption string 232 including a summation or product of the ciphertexts. For example, the cumulative operation can be a running product, where each ciphertext is multiplied together to form the cumulative encryption string 232. In some embodiments, the ciphertexts are multiplied together when all portions of the passcode is entered by the user. In other embodiments, the encryption module 230 iteratively updates the cumulative encryption string 232 to include each newly calculated ciphertext in its cumulative product. The cumulative encryption string 232 can be stored in a cryptography storage 234. In embodiments, the ciphertexts of each portion is deleted from the memory of the passcode entry device 202 after the ciphertext is "accumulated" into the cumulative encryption string 232.

[0050] Cryptography provisions 236 may be required to implement the homomorphic cryptosystem, particularly for the encryption module 230. The cryptography provisions 236 may include, for example, an encryption key (*e.g.*, a public key), a set of prime bases, a modulus, a generator, other static parameters or random variables, or any combination thereof. Some or all of the provisions/parameters may be associated with a hardware identification data of the authentication device 204 (*i.e.*, where decryption occurs). The encryption module 230 can receive the hardware identification data (henceforth referred to as "hardware ID") from the authentication device 204, such as when the authentication device 204 first connects with the passcode entry device 202. Subsequently, the hardware ID may be used to query an external server system, via a network interface 238, for some or all of the cryptography provisions 236. Alternatively, the encryption module 230 can receive some of the cryptography provisions 236, such as the public key for encryption, directly from the authentication device 204. The cryptography provisions 236 may be stored in the

cryptography cache 232 as well or another memory device of the passcode entry device 202. In some embodiments, an association between each set of the cryptography provisions 236 (*e.g.*, the public key) and the hardware ID of a corresponding device for decryption can be stored in the cryptography storage 234. Accessing that association enables the encryption module 230 to select the correct set of provisions corresponding to the intended destination of the cumulative encryption string 232, such as using the public key that matches the private key used for decryption at the intended destination.

[0051] Other examples of the cryptography provisions 236 may be necessary to implement the homomorphic encryption, such as the ElGamal cryptosystem, including a modulus (henceforth referred to as "*N*"), a generator (henceforth referred to as "*g*"), and the set of prime bases p_i. These parameters may be provisioned as part of a mobile application that is downloaded and installed on the passcode entry device 202. Alternatively, these parameters may be requested from an external server system or be part of a factory setting of the passcode entry device 202.

[0052] In some embodiments, to verify the authenticity of the passcode entry, the transaction manager module 224 transmits a message containing the cumulative encryption string 232 indicative of the passcode entry through the second connection interface 222 to the authentication device 204. In alternative embodiments, the transaction manager module 224 transmits the message through the network interface 238 to an external server system to verify the passcode entry. The logic circuitry 206 receives the message through the first connection interface 220 and extracts the cumulative encryption string 232 from the message. The logic circuitry 206 can implement a decryption module 240. The decryption module 240 is configured to decrypt data encrypted by the homomorphic cryptosystem implemented in the encryption module 230. For example, the decryption module 240 can access cryptography provision storage 242. The cryptography provision storage 242 can include a private key associated with the public key used by the encryption module 230. The cryptography provision storage 242 can include other parameters needed for decryption, such as the modulus *N*, the generator g, and the set of prime bases p_i.

[0053] The decryption module 240 determines an accumulated value indicative of the passcode entry by decrypting the cumulative encryption string 232. For example, the accumulated value can be a product of multiplying together encoded portions of the passcode entered by the user. The logic circuitry 206 can then use the accumulated value to verify against an authentic passcode. For example, the logic circuitry 206 can access the authentic passcode when the read head 208 detects the financial transaction card. A sequence of

symbols representative of the passcode entry can be factored from the accumulated value using the set of prime numbers used to encode the symbols of the passcode. The logic circuitry 206 can then verify the determined passcode entry against the authentic passcode in either the financial transaction card (*e.g.,* by sending the determined passcode entry to the financial transaction card) or the authentication device 104.

[0054] Alternatively, the logic circuitry 206 can calculate a comparison value based at least in part on the authentic passcode to compare against the accumulated value of the encoded portions representing the passcode entry. Based on the comparison, the logic circuitry 206 can either authorize or deny the financial transaction.

[0055] Regarding **FIG. 1** and **FIG. 2**, blocks, components, and/or modules associated with the electronic device 102, the passcode verification system 110, the passcode entry device 202 and the authentication device 204, each may be implemented in the form of special-purpose circuitry, or in the form of one or more appropriately programmed programmable processors, or a combination thereof. For example, the modules described can be implemented as instructions on a tangible storage memory capable of being executed by a processor or a controller on a machine. The tangible storage memory may be a volatile or a non-volatile memory. In some embodiments, the volatile memory may be considered "non-transitory" in the sense that it is not a transitory signal. Modules may be operable when executed by a processor or other computing device, *e.g.,* a single board chip, application specific integrated circuit, a field programmable field array, a network capable computing device, a virtual machine terminal device, a cloud-based computing terminal device, or any combination thereof. Caches, memory space, and storages described in the figures can be implemented with the tangible storage memory as well, including volatile or non-volatile memory.

[0056] Each of the modules may operate individually and independently of other modules. Some or all of the modules may be executed on the same host device or on separate devices. The separate devices can be coupled via a communication module to coordinate its operations via an interconnect or wirelessly. Some or all of the modules may be combined as one module.

[0057] A single module may also be divided into sub-modules, each sub-module performing separate method step or method steps of the single module. In some embodiments, the modules can share access to a memory space. One module may access data accessed by or transformed by another module. The modules may be considered "coupled" to one another if they share a physical connection or a virtual connection, directly

or indirectly, allowing data accessed or modified from one module to be accessed in another module. In some embodiments, some or all of the modules can be upgraded or modified remotely. The electronic device 102, the passcode verification system 110, the passcode entry device 202, or the authentication device 204 may include additional, fewer, or different modules for various applications.

[0058] **FIG. 3** is a diagram illustrating an example of a timeline 300 for encrypting a sequence of symbols representing a passcode. The disclosed technique involves piecemeal encryption of portions of the passcode. This technique minimizes the available time window during which an unencrypted passcode symbol, such as a PIN digit, is present (*i.e.,* available on memory) on a passcode entry device, such as the electronic device 102 of **FIG. 1** or the passcode entry device 202 of **FIG. 2**. The piecemeal encryption technique decouples the digits from each other.

[0059] Time tags 302A-F (collectively, "time tags 302") indicate events related to a user's entry of the sequence of symbols. For example, time tag 302A ("t_0") indicates when a passcode entry interface is provided on the passcode entry device. Time tags 302B to 302E ("t_1" to "t_4") indicate when the passcode entry device identifies/determines each symbol/digit of the passcode entry. For example, t_1 may correspond to when the first digit entered by the user is identified and t_2 may correspond to when the second digit is identified. Where the passcode entry device uses a touchscreen to display an interactive passcode entry interface, each of the time tags 302B-E may correspond with when a touch event is detected or when the passcode entry device determines which symbol corresponds to that touch event. Under the disclosed technique of passcode encryption, the passcode entry device can immediately or substantially immediately encrypt each of these symbols/digits after the symbol/digit is identified on the device.

[0060] One security consideration in designing a passcode entry system is whether a memory snapshot of the passcode entry device can compromise the passcode entered by the user. As shown, if memory were captured shortly after t_i, then the memory snapshot would reveal no information about the first digit through the $(i-1)^{th}$ digit of the passcode. The memory snapshot would also reveal no information about subsequent digits, such as $(i+1)^{th}$ digit through the last digit, of the passcode. In the example of using the ElGamal cryptosystem, once the per-digit secret/multiplier is deleted, the encryption is not reversible by any party who does not possess the private key. In the end, the passcode entry device can safely transmit the cumulative encryption to a passcode verification system at time tag 302F.

[0061] **FIG. 4** is flow chart of a process 400 to encrypt a passcode using homomorphic cryptography. The process 400 may be consistent with various other embodiments described in this disclosure. The process 400 performs a piecemeal encryption of the passcode in a cumulative manner. Step 402 includes an electronic device, such as the electronic device 102 of **FIG. 1** or the passcode entry device 202 of **FIG. 2**, initializing a financial transaction (*e.g.,* a payment transaction). Step 402 may include receiving card data from a card reader in response to the card reader detecting a financial transaction card. For example, the card reader may be the passcode verification device 110 of **FIG. 1** or the authentication device 204 of **FIG. 2**.

[0062] The electronic device then displays a passcode entry interface to solicit a user to enter the passcode in step 404, where the passcode is used to authenticate the financial transaction. For example, the electronic device can display the passcode entry interface on a touchscreen. The passcode entry interface may include interactive elements labeled with symbols (*e.g.,* characters, digits, icons, etc.) for composing the passcode.

[0063] Through the passcode entry interface, the electronic device receives an indication from the user identifying at least a portion of the passcode in step 406. For example, the indication can be a key entry corresponding to a symbol that is part of the passcode. Specifically, the indication can be a touch event on the touchscreen, where a location of the touch event corresponds to a label of the symbol on the passcode entry interface.

[0064] The electronic device can encode the identified portion as a plaintext value in step 408, prior to step 410 of encrypting the encoded portion. The identified portion, such as a symbol, can have a numeric representation ($\mathbf{d}_i$), such as a number represented by binary bits. Here, "i" denotes the position of the symbol in the passcode. The encoding in step 408 may include selecting a prime number ($\mathbf{p}_i$) from a set of small prime bases and calculating the prime number raised to the power equal to the numeric representation of the identified portion, *i.e.,* $\mathbf{p}_i^{d\_i}$. The prime number used to encode the identified portion can be selected based at least in part on the position of the portion in the passcode relative to other portions.

[0065] This set of small prime bases may be part of the cryptography parameters stored in the electronic device. The cryptography parameters may include an encryption key (*e.g.,* a public encryption key), a generator g, a modulus N, the set of prime bases, or any combination thereof. The process 400 may include step 412 of receiving such cryptography parameters from an external source. The electronic device may receive at least a portion of the cryptography parameters directly or indirectly from the card reader. The electronic

device may instead receive at least a portion of the cryptography parameters directly or indirectly from an external server system. For example, the electronic device may first retrieve a hardware ID of the card reader, and then query the external server system for cryptography parameters associated with the hardware ID. In some embodiments, the cryptography parameters may be retrieved from both the card reader and the external server system. This way, the electronic device has an opportunity to verify the parameters against each other. The cryptography parameters may also be encrypted and/or otherwise encoded with a verifiable signature. Upon receiving the cryptography parameters, the electronic device can verify these cryptography parameters using the verifiable signature, such as via a certificate authority. In alternative embodiments, at least a portion of the cryptography parameters is downloaded onto the electronic device when a mobile application implementing the process 400 is downloaded and installed onto the electronic device.

[0066] The prime numbers used for encoding in step 408 may be based on one or more of the cryptography parameters retrieved in step 412. For example, when encrypting using the ElGamal encryption mechanism, the prime numbers may satisfy the requirement that $\mathbf{p}_i^q \equiv 1 \pmod{N}$. Here, N is the modulus chosen to perform the ElGamal encryption and q is a Sophie Germain prime associated with N such that $N = 2q + 1$. N and/or q may be part of the cryptography parameters accessible to the electronic device.

[0067] Step 410 includes the electronic device encrypting the encoded portion from step 408 or the identified portion from step 406 into ciphertext using a homomorphic encryption system, such as the encryption mechanism of the ElGamal cryptosystem or a variant thereof. The encryption may utilize the received cryptography parameters in step 412, including the encryption key.

[0068] Certain encryption mechanisms, including the El Gamal encryption mechanism, generate and use a random variable as part of the encryption to protect the confidentiality of the ciphertexts. It is noted that in step 410, the electronic device may generate the random variable by reading a pseudorandom stream using a non-buffering I/O call. A non-buffering I/O call can minimize exposure of the random variable in the memory buffer of the electronic device to avoid a third party from influencing or reading the random variable in order to extract information related to the passcode.

[0069] The electronic device updates a cumulative encryption string when the encryption finishes in step 414. If the encoded portion corresponds to the first portion of the passcode, step 414 sets the cumulative encryption string as the ciphertext for that portion. Otherwise, the cumulative encryption string is updated by executing a cumulative operation

on the ciphertext based on the cumulative encryption string computed for a previous portion of the passcode. The homomorphic property of the homomorphic encryption system dictates which cumulative operation to execute. For example, where ElGamal encryption mechanism is used, the cumulative operation is a cumulative product (*i.e.,* multiplying the ciphertexts together). That is, the cumulative encryption string can be set as a cumulative product of the ciphertext and all previous ciphertexts corresponding to all previously encrypted portions of the passcode.

[0070] Step 406, step 408, step 410, and step 414 are repeated until the cumulative encryption string "accumulates" every portion of the passcode. In step 416, the electronic device transmits a message containing the cumulative encryption string to a destination device to verify the passcode. For example, the destination device can be the card reader or an external server system.

[0071] **FIG. 5** is a flow chart of a process 500 to authenticate a passcode entry based at least partly on an encrypted message. The process 500 may be consistent with various other embodiments described in this disclosure. The process 500 may be implemented by a passcode verification system, such as the passcode verification system 110 of **FIG. 1** or the authentication device 204 of **FIG. 2**. Step 502 includes receiving ciphertext from a passcode entry device, such as the electronic device 102 of **FIG. 1** or the passcode entry device 202 of **FIG. 2**. The passcode verification system can decrypt the ciphertext into plaintext in step 504. The decryption uses a decryption mechanism of a homomorphic cryptosystem, such as the ElGamal cryptosystem.

[0072] The plaintext is indicative of a product or a sum of encoded portions of the passcode entry. Each encoded portion represents a portion of the passcode entry and a position of the portion relative to other portions of the passcode entry. For example, the encoded portion may be indicative of a symbol that is part of the passcode. Subsequently, in step 506, the passcode verification system verifies the passcode entry based at least in part on the product or the sum of the encoded portions.

[0073] As a specific example, step 506 can include sub-step 508 to retrieve an authentic passcode. Where the passcode verification system is a card reader device, sub-step 508 includes retrieving the authentic passcode from a financial transaction card via a read head. In sub-step 510, the passcode verification system can encode portions of the authentic passcode in the same manner as how the encoded portions of the passcode entry are calculated on the passcode entry device. In sub-step 512, the passcode verification system can calculate a cumulative value through a cumulative operation on the encoded portions of

the authentic passcode. The cumulative operation is dictated by the homomorphic property of the homomorphic cryptosystem. For example, where the homomorphic property preserves multiplication, the cumulative value is calculated as a cumulative product of the encoded portions of the authentic passcode. In sub-step 514, the passcode verification system can authenticate the passcode entry by comparing the cumulative value against the product or the sum of the encoded portions of the passcode entry.

[0074] As another specific example, after retrieving the authentic passcode in sub-step 508, the passcode verification system can determine a sequence of symbols representative of the passcode entry by factoring the product of the encoded portions in sub-step 516. The product may be factored using a set of prime numbers used to encode each portion. The passcode verification system can then compare the sequence against the authentic passcode in sub-step 518. In other embodiments, the passcode verification system sends the sequence to the financial transaction card for the financial transaction card to verify the sequence against the authentic passcode.

[0075] While processes or blocks are presented in a given order in **FIGs. 4** and **5**, alternative embodiments may perform routines having steps, or employ systems having blocks, in a different order, and some processes or blocks may be deleted, moved, added, subdivided, combined, and/or modified to provide alternative or subcombinations. Each of these processes or blocks may be implemented in a variety of different ways. In addition, while processes or blocks are at times shown as being performed in series, these processes or blocks may instead be performed in parallel, or may be performed at different times.

[0076] **FIG. 6** is a block diagram of an example of a financial transaction system 600 including an electronic device 602 (*e.g.,* the electronic device 102 of **FIG. 1** or the passcode entry device 202 of **FIG. 2**) and a card reader 604 (*e.g.,* the passcode verification system 110 of **FIG. 1** or the authentication device 204 of **FIG. 2**). Note that the architecture shown in **FIG. 6** is only one example architecture of the financial transaction system 600, and that the electronic device 602 can have more or fewer components than shown, or a different configuration of components. The various components shown in **FIG. 6** can be implemented by using hardware, software, firmware or a combination thereof, including one or more signal processing and/or application specific integrated circuits.

[0077] The electronic device 602 that can include one or more computer-readable mediums 610, processing system 620, touch subsystem 630, display/graphics subsystem 640, communications circuitry 650, storage 660, and audio circuitry 670. These components may be coupled by one or more communication buses or other signal lines.

[0078] The communications circuitry 650 can include RF circuitry 652 and/or port 654 for sending and receiving information. The RF circuitry 652 permits transmission of information over a wireless link or network to one or more other devices and includes well-known circuitry for performing this function. The port 654 permits transmission of information over a wired link. The communications circuitry 650 can communicate, for example, with the card reader 604. Alternatively, the card reader 604 may be connected through the audio circuitry 670. The communications circuitry 650 can be coupled to the processing system 620 via a peripherals interface 624. The peripherals interface 624 can include various known components for establishing and maintaining communication between peripherals and the processing system 620.

[0079] The audio circuitry 670 can be coupled to an audio speaker (not shown), a microphone (not shown), the card reader 604, or any combination thereof and includes known circuitry for processing signals received from the peripherals interface 624 to enable a user to communicate in real-time with other users or system(s). In some embodiments, the audio circuitry 670 includes a headphone jack (not shown).

[0080] The peripherals interface 624 can couple with various peripherals, such as the card reader 604, of the system to one or more processors 626 and the computer-readable medium 610. The one or more processors 626 can communicate with one or more computer-readable mediums 610 via a controller 622. The computer-readable medium 610 can be any device or medium that can store code and/or data for use by the one or more processors 626. The medium 610 can include a memory hierarchy, including but not limited to cache, main memory, and secondary memory. The memory hierarchy can be implemented using any combination of RAM (*e.g.*, SRAM, DRAM, DDRAM), ROM, FLASH, magnetic and/or optical storage devices, such as disk drives, magnetic tape, CDs (compact disks) and DVDs (digital video discs). The medium 610 may also include a transmission medium for carrying information-bearing signals indicative of computer instructions or data (with or without a carrier wave upon which the signals are modulated). For example, the transmission medium may include a communications network, including but not limited to the Internet, intranet(s), Local Area Networks (LANs), Wide Local Area Networks (WLANs), Storage Area Networks (SANs), Metropolitan Area Networks (MAN) and the like.

[0081] The one or more processors 626 can run various software components stored in the medium 610 to perform various functions for the electronic device 602. Note that the order of the modules in the medium 610 does not denote the order of a software stack as implemented in the medium 610. In some embodiments, the software components include an

operating system 611, a communication module (or set of instructions) 612, a touch processing module (or set of instructions) 613, a passcode interface module (or set of instructions) 615, such as the passcode interface module 226 of **FIG. 2**, and one or more applications (or set of instructions) 618, for example, including one or more of the modules described in the electronic device 102 and/or the passcode entry device 202. Each of these modules and above noted applications correspond to a set of instructions for performing one or more functions described above and the methods described in this application (*e.g.*, the computer-implemented methods and other information processing methods described herein). These modules (*e.g.*, sets of instructions) need not be implemented as separate software programs, procedures, or modules, and thus various subsets of these modules may be combined or otherwise rearranged in various embodiments. In some embodiments, the medium 610 may store a subset of the modules and data structures identified above. Furthermore, the medium 610 may store additional modules and data structures not described above.

[0082] The operating system 611 can include various procedures, sets of instructions, software components, and/or drivers for controlling and managing general system tasks (*e.g.*, memory management, storage device control, power management, etc.) and facilitates communication between various hardware and software components.

[0083] The communication module 612 facilitates communication with other devices using the communications circuitry 650 and includes various software components for handling data received from the RF circuitry 652 and/or the port 654.

[0084] The touch-processing module 613 includes various software components for performing various tasks associated with touch hardware 634 including but not limited to receiving and processing touch input received from the I/O device 630 via a touch I/O device controller 632. For example, the touch-processing module 613 can also include software components for performing tasks associated with other I/O devices (not shown).

[0085] The passcode interface module 615 is configured to present and maintain a passcode interface for a user to enter a passcode to authenticate the user's identity. The passcode interface module 615 can include various known software components for rendering, animating and displaying graphical objects on a display surface. In embodiments, in which the touch hardware 634 is a touch sensitive display (*e.g.*, touch screen), the passcode interface module 615 includes components for rendering, displaying, and animating objects on the touch sensitive display. The passcode interface module 615 can provide graphics instructions (*e.g.*, animation or still image) to graphics I/O controller 644, so that the graphics

I/O controller 644 can display the graphics on display 646. The passcode interface module 615 can further control the audio circuitry 670 to provide an auditory component to the passcode interface.

[0086] One or more applications 618 can include any applications installed on the electronic device 602, including without limitation, modules of the electronic device 102 and/or the passcode entry device 202, a browser, keyboard emulation, widgets, JAVA-enabled applications, encryption, digital rights management, voice recognition, voice replication, location determination capability (such as that provided by the global positioning system (GPS)), etc.

[0087] The touch I/O controller 632 is coupled to the touch hardware 634 for controlling or performing various functions. The touch hardware 634 communicates with the processing system 620 via the touch I/O device controller 632, which includes various components for processing user touch input (*e.g.,* scanning hardware). One or more other input controllers (not shown) receives/sends electrical signals from/to other I/O devices (not shown). Other I/O devices may include physical buttons, dials, slider switches, sticks, keyboards, touch pads, additional display screens, or any combination thereof.

[0088] If embodied as a touch screen, the touch hardware 634 displays visual output to the user in a GUI. The visual output may include text, graphics, video, and any combination thereof. Some or all of the visual output may correspond to user-interface objects. The touch hardware 634 forms a touch-sensitive surface that accepts touch input from the user. The touch hardware 634 and the touch controller 632 (along with any associated modules and/or sets of instructions in the medium 610) detects and tracks touches or near touches (and any movement or release of the touch) on the touch hardware 634 and converts the detected touch input into interaction with graphical objects, such as one or more user-interface objects. In the case in which the touch hardware 634 and the display 646 are embodied as a touch screen, the user can directly interact with graphical objects that are displayed on the touch screen. Alternatively, in the case in which hardware 634 is embodied as a touch device other than a touch screen (*e.g.,* a touch pad), the user may indirectly interact with graphical objects that are displayed on a separate display screen. In some embodiments, the touch controller 632 may be configured such that it is disabled when a passcode interface is being displayed by the display 646.

[0089] Embodiments in which the touch hardware 634 is a touch screen, the touch screen may use LCD (liquid crystal display) technology, LPD (light emitting polymer

display) technology, OLED (organic light emitting diode), or OEL (organic electro luminescence), although other display technologies may be used in other embodiments.

[0090] In some embodiments, the peripherals interface 624, the one or more processors 626, and the memory controller 622 may be implemented on a single chip. In some other embodiments, they may be implemented on separate chips. The storage 660 can be any suitable medium for storing data, including, for example, volatile memory (*e.g.,* cache, RAM), non-volatile memory (*e.g.,* Flash, hard-disk drive), or a combination of both for storing data.

EXAMPLES

To summarize, therefore, the above disclosure includes the following examples.

1. A method comprising:

initializing, at a mobile device, a financial transaction and receiving card data from a card reader attached to the mobile device, wherein the initializing is in response to detecting a swipe of a financial transaction card at the card reader;

displaying a passcode entry interface on a touchscreen of the mobile device to enable a user to enter a passcode;

determining a sequence of digits corresponding to the passcode entered by the user;

for each digit of the sequence, encoding the digit into an encoded digit, and encrypting the encoded digit into ciphertext using homomorphic cryptography; and

maintaining a cumulative product based at least partly on the ciphertexts corresponding to the digits of the sequence; and

initiating a process to authenticate the financial transaction by transmitting a message indicative of at least the cumulative product, wherein the cumulative product represents the passcode entered by the user.

2. The method of example 1, wherein maintaining the cumulative product includes multiplying the ciphertext corresponding to a newly entered digit with the cumulative product calculated for all previous digits in the sequence.

3. The method of example 1, wherein encoding each digit and encrypting each encoded digit are performed in response to determining the digit entered by the user; and wherein maintaining the cumulative product includes updating the cumulative product in response to encrypting each encoded digit.

4. The method of example 1, wherein encoding the digit includes calculating the encoded digit as a prime number raised to the power of the digit.

5. The method of example 1, further comprising:

receiving a public key for said encrypting using the homomorphic cryptography from an external server system; and

verifying a signature of the public key using a certificate authority.

6. A method of encrypting a passcode, the method comprising:

receiving an indication of a portion of the passcode;

calculating, via a processor of a computing device, a plaintext value based at least in part on the indication, wherein the plaintext value represents an encoded portion of the passcode;

encrypting, via the processor, the plaintext value into ciphertext using a homomorphic encryption system; and

updating, via the processor, a cumulative encryption string by executing a cumulative operation to aggregate the ciphertext corresponding to the encoded portion into the cumulative encryption string computed for a previous portion of the passcode, wherein the cumulative operation is in accordance with a homomorphic property of the homomorphic encryption system.

7. The method of example 6, wherein receiving the indication includes receiving a key entry corresponding to a symbol that is part of the passcode.

8. The method of example 7, wherein calculating the plaintext value is based at least in part on a numeric representation ("d_i") of the symbol.

9. The method of example 8, wherein the symbol is a numeric digit and the numeric representation of the symbol is the numeric digit.

10. The method of example 8, wherein calculating the plaintext value comprises:

selecting, based on a sequential position ("i") of the portion corresponding to the indication received, a prime number ("p_i") from a set of prime number bases; and

calculating the selected prime number raised to the power of the numeric representation of the symbol.

11. The method of example 10, further comprising selecting the prime number from a set of prime bases based at least partly on the position of the digit in the sequence of digits.

12. The method of example 11, wherein each prime number (**p**) in the set of prime bases satisfies $\mathbf{p}^q \equiv 1 \pmod{N}$, wherein the homomorphic cryptography is based on ElGamal asymmetric encryption and N is the modulus chosen to perform the ElGamal encryption and q is a Sophie Germain prime associated with N such that $N = 2q + 1$.

13. The method of example 10, further comprising receiving at least a portion of cryptography parameters, including the set of prime bases, for the homomorphic cryptography from an external server system, wherein the cryptography parameters are associated with a hardware ID of the card reader.

14. The method of example 10, further comprising receiving at least a portion of cryptography parameters, including the set of prime bases, for the homomorphic cryptography from the card reader.

15. The method of example 7, further comprising displaying a passcode entry interface on a touchscreen, the passcode entry interface includes interactive elements labeled with symbols for composing passcodes; wherein receiving the key entry includes receiving a touch event on the touchscreen; and wherein the touch event includes a location on the passcode entry interface where the touch event occurs.

16. The method of example 6, wherein encrypting using the homomorphic encryption system includes encrypting using an ElGamal encryption system or a variant thereof.

17. The method of example 6, further comprising transmitting a message indicative of at least the cumulative encryption string to a destination device to verify the passcode.

18. The method of example 17, further comprising requesting a public key associated with a hardware ID of the destination device from an external server; and wherein encrypting the plaintext value includes encrypting the plaintext value using the public key.

19. The method of example 17, further comprising requesting a public key from the destination device; and wherein encrypting the plaintext value includes encrypting the plaintext value using the public key.

20. The method of example 6, wherein executing the cumulative operation includes computing a product of the ciphertext and a cumulative product of all previous ciphertexts corresponding to all previously encrypted portions of the passcode.

21. The method of example 6, wherein encrypting the plaintext value includes generating a random variable by reading a pseudorandom stream using a non-buffering I/O

call; and wherein encrypting the plaintext value is based at least in part on the random variable.

22. A method of verifying a passcode entry by a user, the method comprising:

receiving ciphertext from a passcode entry device;

decrypting the ciphertext into plaintext, wherein the plaintext is indicative of a product of encoded portions of the passcode entry, wherein each encoded portion represents a portion in the passcode entry and a position of the portion in the passcode entry; and

verifying the passcode entry based at least partly on the product of the encoded portions.

23. The method of example 22, wherein each encoded portion is indicative of a symbol that is part of the passcode entry and the position of the symbol in the passcode entry.

24. The method of example 22, wherein decrypting the ciphertext includes decrypting the ciphertext using a decryption mechanism of a homomorphic cryptosystem.

25. The method of example 24, wherein decrypting the ciphertext includes decrypting using an ElGamal encryption system or a variant thereof.

26. The method of example 22, wherein verifying the passcode entry includes:

calculating a hash string based at least partly on an authentic passcode; and

verifying the passcode entry based at least partly on the hash string and the product of the encoded portions.

27. The method of example 22, wherein verifying the passcode entry includes:

retrieving an authentic passcode; and

determining a sequence of symbols representative of the passcode entry by factoring the product of the encoded portions using a set of prime numbers used to encode each portion; and

sending the sequence of symbols to be compared against the authentic passcode in a financial transaction card.

28. An apparatus comprising:

a memory device storing executable instructions, that, when executed by a processor, is operable to:

receive an indication of a portion of a passcode;

calculate a plaintext value based at least in part on the indication, wherein the plaintext value represents an encoded portion of the passcode;

encrypt the plaintext value into ciphertext using a homomorphic encryption system; and

update a cumulative encryption string by executing a cumulative operation to aggregate the ciphertext corresponding to the encoded portion into the cumulative encryption string computed for a previous portion of the passcode, wherein the cumulative operation is in accordance with a homomorphic property of the homomorphic encryption system.

29. An apparatus comprising:

an interface to receive a message including ciphertext representing a passcode entry of a user;

a read head to access card information stored in a payment card; and

a logic circuitry configured to:

decrypt the ciphertext into plaintext, wherein the plaintext is indicative of a product of encoded portions of the passcode entry, wherein each encoded portion represents a portion in the passcode entry and a position of the portion in the passcode entry; and

verify the passcode entry based at least partly on the product of the encoded portions and an authentic passcode accessible via the read head.

What is claimed is:

1. A method comprising:

initializing, at a mobile device, a financial transaction and receiving card data from a card reader attached to the mobile device, wherein the initializing is in response to detecting a swipe of a financial transaction card at the card reader;

displaying a passcode entry interface on a touchscreen of the mobile device to enable a user to enter a passcode;

determining a sequence of digits corresponding to the passcode entered by the user;

for each digit of the sequence, encoding the digit into an encoded digit, and encrypting the encoded digit into ciphertext using homomorphic cryptography; and

maintaining a cumulative product based at least partly on the ciphertexts corresponding to the digits of the sequence; and

initiating a process to authenticate the financial transaction by transmitting a message indicative of at least the cumulative product, wherein the cumulative product represents the passcode entered by the user.

2. The method of claim 1, wherein maintaining the cumulative product includes multiplying the ciphertext corresponding to a newly entered digit with the cumulative product calculated for all previous digits in the sequence.

3. The method of claim 1, wherein encoding each digit and encrypting each encoded digit are performed in response to determining the digit entered by the user; and wherein maintaining the cumulative product includes updating the cumulative product in response to encrypting each encoded digit.

4. The method of claim 1, wherein encoding the digit includes calculating the encoded digit as a prime number raised to the power of the digit.

5. The method of claim 1, further comprising:

receiving a public key for said encrypting using the homomorphic cryptography from an external server system; and

verifying a signature of the public key using a certificate authority.

6. A method of encrypting a passcode, the method comprising:

receiving an indication of a portion of the passcode;

calculating, via a processor of a computing device, a plaintext value based at least in part on the indication, wherein the plaintext value represents an encoded portion of the passcode;

encrypting, via the processor, the plaintext value into ciphertext using a homomorphic encryption system; and

updating, via the processor, a cumulative encryption string by executing a cumulative operation to aggregate the ciphertext corresponding to the encoded portion into the cumulative encryption string computed for a previous portion of the passcode, wherein the cumulative operation is in accordance with a homomorphic property of the homomorphic encryption system.

7. The method of claim 6, wherein receiving the indication includes receiving a key entry corresponding to a symbol that is part of the passcode.

8. The method of claim 7, wherein calculating the plaintext value is based at least in part on a numeric representation ("d_i") of the symbol.

9. The method of claim 8, wherein the symbol is a numeric digit and the numeric representation of the symbol is the numeric digit.

10. The method of claim 8, wherein calculating the plaintext value comprises:

selecting, based on a sequential position ("i") of the portion corresponding to the indication received, a prime number ("p_i") from a set of prime number bases; and

calculating the selected prime number raised to the power of the numeric representation of the symbol.

11. The method of claim 10, further comprising selecting the prime number from a set of prime bases based at least partly on the position of the digit in the sequence of digits.

12. The method of claim 11, wherein each prime number (**p**) in the set of prime bases satisfies $\mathbf{p}^q \equiv 1 \pmod{N}$, wherein the homomorphic cryptography is based on ElGamal asymmetric encryption and N is the modulus chosen to perform the ElGamal encryption and q is a Sophie Germain prime associated with N such that $N = 2q + 1$.

13. The method of claim 10, further comprising receiving at least a portion of cryptography parameters, including the set of prime bases, for the homomorphic cryptography from an external server system, wherein the cryptography parameters are associated with a hardware ID of the card reader.

14. The method of claim 10, further comprising receiving at least a portion of cryptography parameters, including the set of prime bases, for the homomorphic cryptography from the card reader.

15. The method of claim 7, further comprising displaying a passcode entry interface on a touchscreen, the passcode entry interface includes interactive elements labeled with symbols for composing passcodes; wherein receiving the key entry includes receiving a touch event on the touchscreen; and wherein the touch event includes a location on the passcode entry interface where the touch event occurs.

16. The method of claim 6, wherein encrypting using the homomorphic encryption system includes encrypting using an ElGamal encryption system or a variant thereof.

17. The method of claim 6, further comprising transmitting a message indicative of at least the cumulative encryption string to a destination device to verify the passcode.

18. The method of claim 17, further comprising requesting a public key associated with a hardware ID of the destination device from an external server; and wherein encrypting the plaintext value includes encrypting the plaintext value using the public key.

19. The method of claim 17, further comprising requesting a public key from the destination device; and wherein encrypting the plaintext value includes encrypting the plaintext value using the public key.

20. The method of claim 6, wherein executing the cumulative operation includes computing a product of the ciphertext and a cumulative product of all previous ciphertexts corresponding to all previously encrypted portions of the passcode.

21. The method of claim 6, wherein encrypting the plaintext value includes generating a random variable by reading a pseudorandom stream using a non-buffering I/O call; and wherein encrypting the plaintext value is based at least in part on the random variable.

22. A method of verifying a passcode entry by a user, the method comprising:

receiving ciphertext from a passcode entry device;

decrypting the ciphertext into plaintext, wherein the plaintext is indicative of a product of encoded portions of the passcode entry, wherein each encoded portion represents a portion in the passcode entry and a position of the portion in the passcode entry; and

verifying the passcode entry based at least partly on the product of the encoded portions.

23. The method of claim 22, wherein each encoded portion is indicative of a symbol that is part of the passcode entry and the position of the symbol in the passcode entry.

24. The method of claim 22, wherein decrypting the ciphertext includes decrypting the ciphertext using a decryption mechanism of a homomorphic cryptosystem.

25. The method of claim 24, wherein decrypting the ciphertext includes decrypting using an ElGamal encryption system or a variant thereof.

26. The method of claim 22, wherein verifying the passcode entry includes:

calculating a hash string based at least partly on an authentic passcode; and

verifying the passcode entry based at least partly on the hash string and the product of the encoded portions.

27. The method of claim 22, wherein verifying the passcode entry includes:

retrieving an authentic passcode; and

determining a sequence of symbols representative of the passcode entry by factoring the product of the encoded portions using a set of prime numbers used to encode each portion; and

sending the sequence of symbols to be compared against the authentic passcode in a financial transaction card.

28. An apparatus comprising:

a memory device storing executable instructions, that, when executed by a processor, is operable to:

receive an indication of a portion of a passcode;

calculate a plaintext value based at least in part on the indication, wherein the plaintext value represents an encoded portion of the passcode;

encrypt the plaintext value into ciphertext using a homomorphic encryption system; and

update a cumulative encryption string by executing a cumulative operation to aggregate the ciphertext corresponding to the encoded portion into the cumulative encryption string computed for a previous portion of the passcode, wherein the cumulative operation is in accordance with a homomorphic property of the homomorphic encryption system.

29. An apparatus comprising:

an interface to receive a message including ciphertext representing a passcode entry of a user;

a read head to access card information stored in a payment card; and

a logic circuitry configured to:

decrypt the ciphertext into plaintext, wherein the plaintext is indicative of a product of encoded portions of the passcode entry, wherein each encoded portion represents a portion in the passcode entry and a position of the portion in the passcode entry; and

verify the passcode entry based at least partly on the product of the encoded portions and an authentic passcode accessible via the read head.

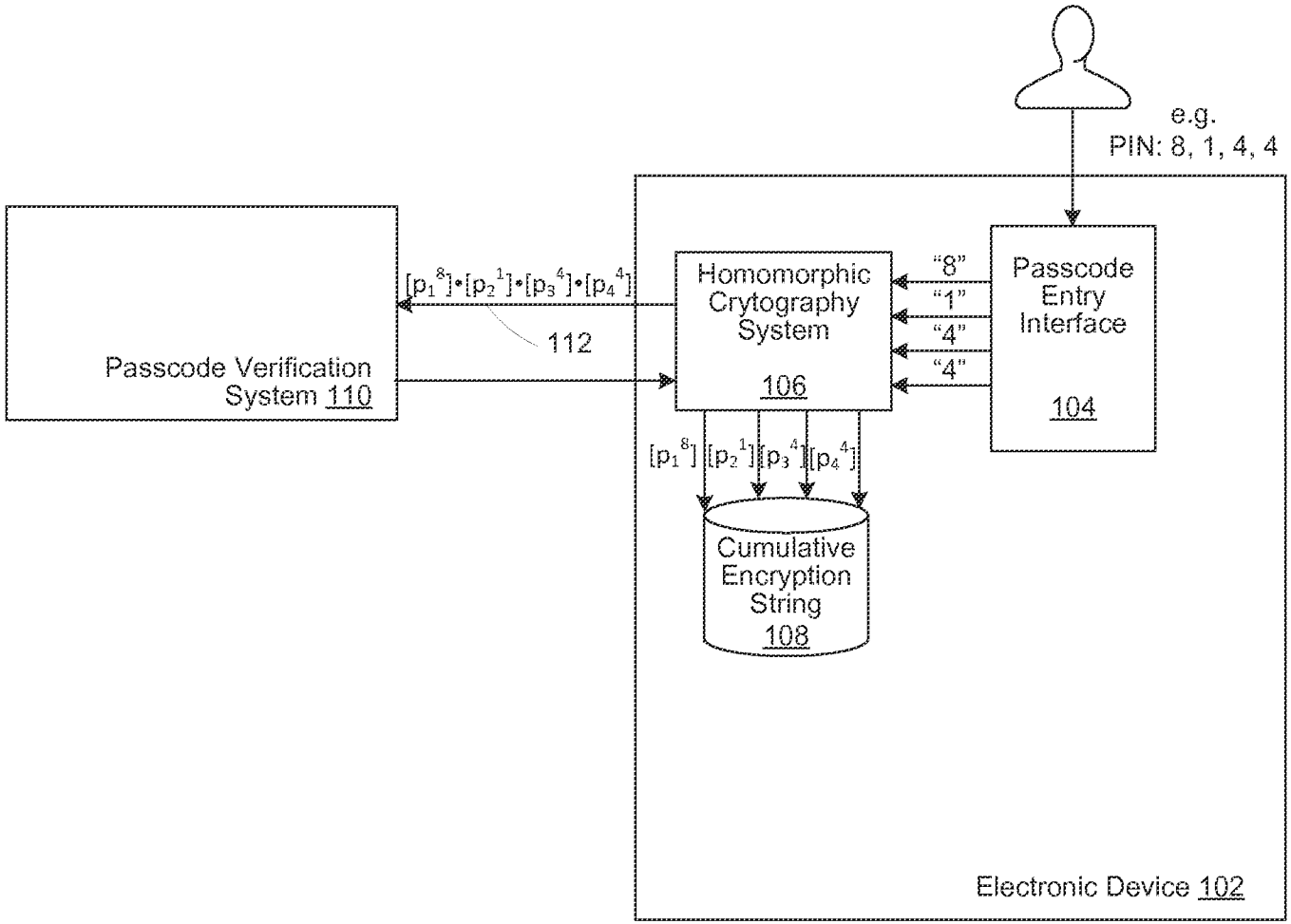
e.g.
PIN: 8, 1, 4, 4
Passcode Verification System 110
$[p_1^8]*[p_2^1]*[p_3^4]*[p_4^4]$
112
Homomorphic Crytography System
106
"8"
"1"
"4"
"4"
Passcode Entry Interface
104
$[p_1^8]$ $[p_2^1]$ $[p_3^4]$ $[p_4^4]$
Cumulative Encryption String
108
Electronic Device 102

FIG. 1

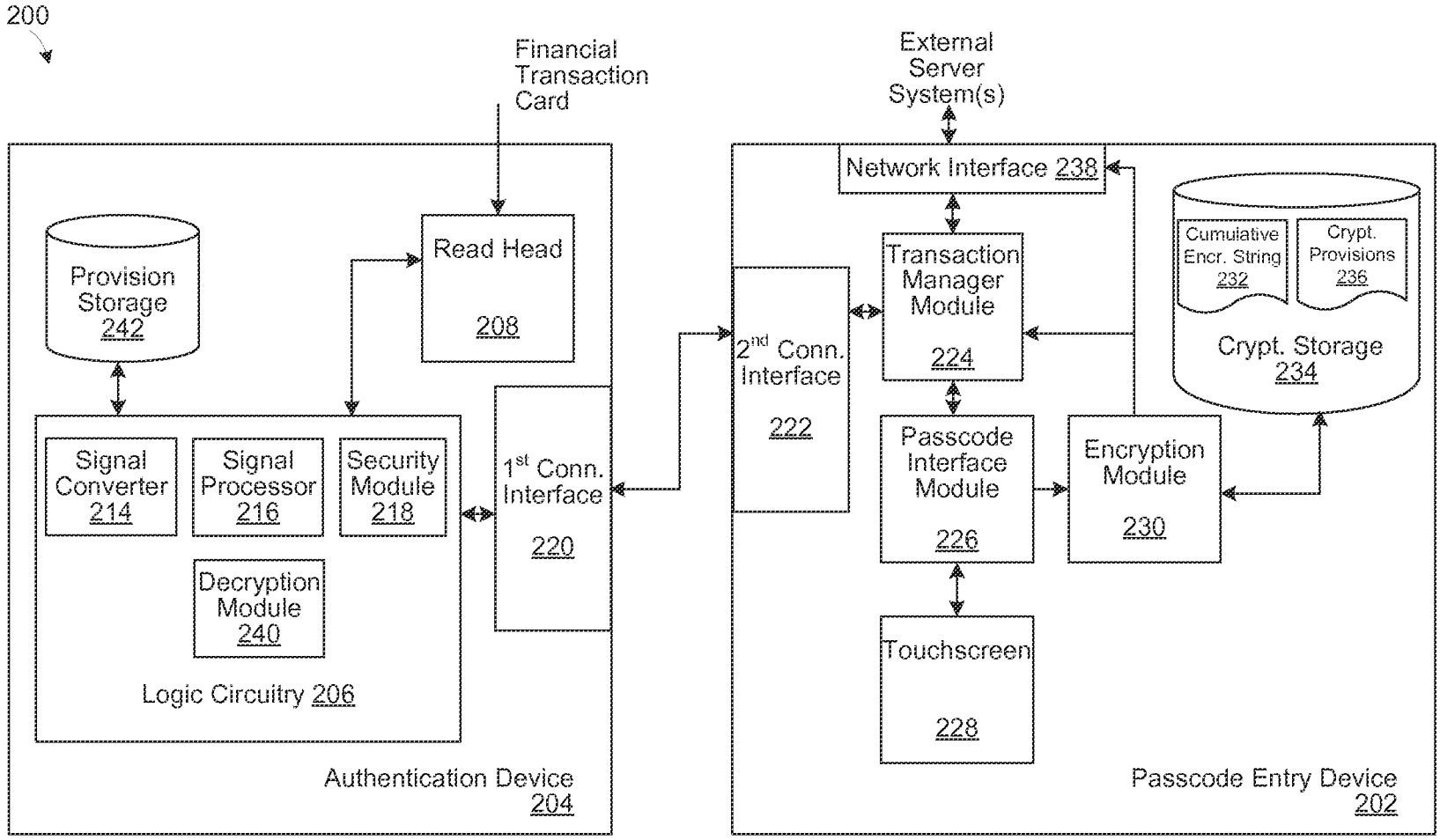
200
Financial Transaction Card
External Server System(s)
Network Interface 238
Provision Storage 242
Read Head 208
Transaction Manager Module 224
Cumulative Encr. String 232
Crypt. Provisions 236
Crypt. Storage 234
2nd Conn. Interface 222
Signal Converter 214
Signal Processor 216
Security Module 218
1st Conn. Interface 220
Passcode Interface Module 226
Encryption Module 230
Decryption Module 240
Touchscreen 228
Logic Circuitry 206
Authentication Device 204
Passcode Entry Device 202

FIG. 2

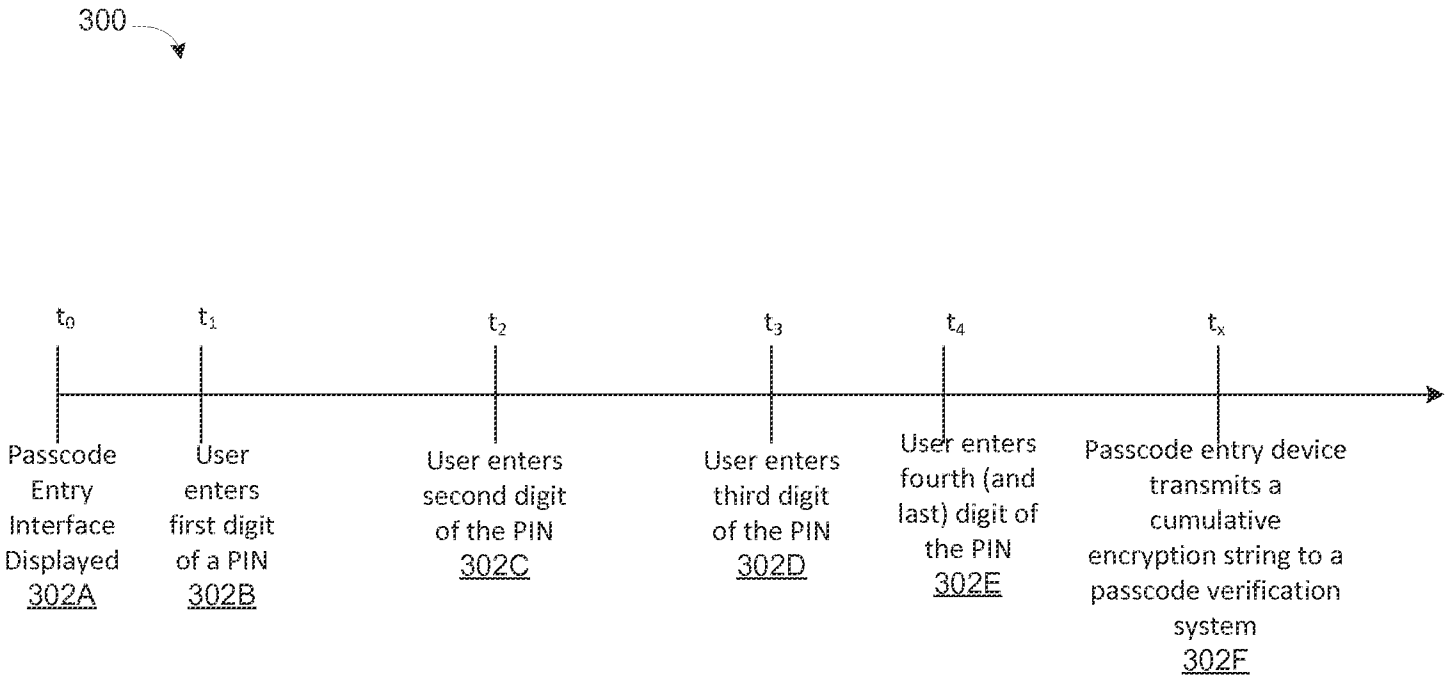
300
t_0
t_1
t_2
t_3
t_4
t_x
Passcode Entry Interface Displayed 302A
User enters first digit of a PIN 302B
User enters second digit of the PIN 302C
User enters third digit of the PIN 302D
User enters fourth (and last) digit of the PIN 302E
Passcode entry device transmits a cumulative encryption string to a passcode verification system 302F

FIG. 3

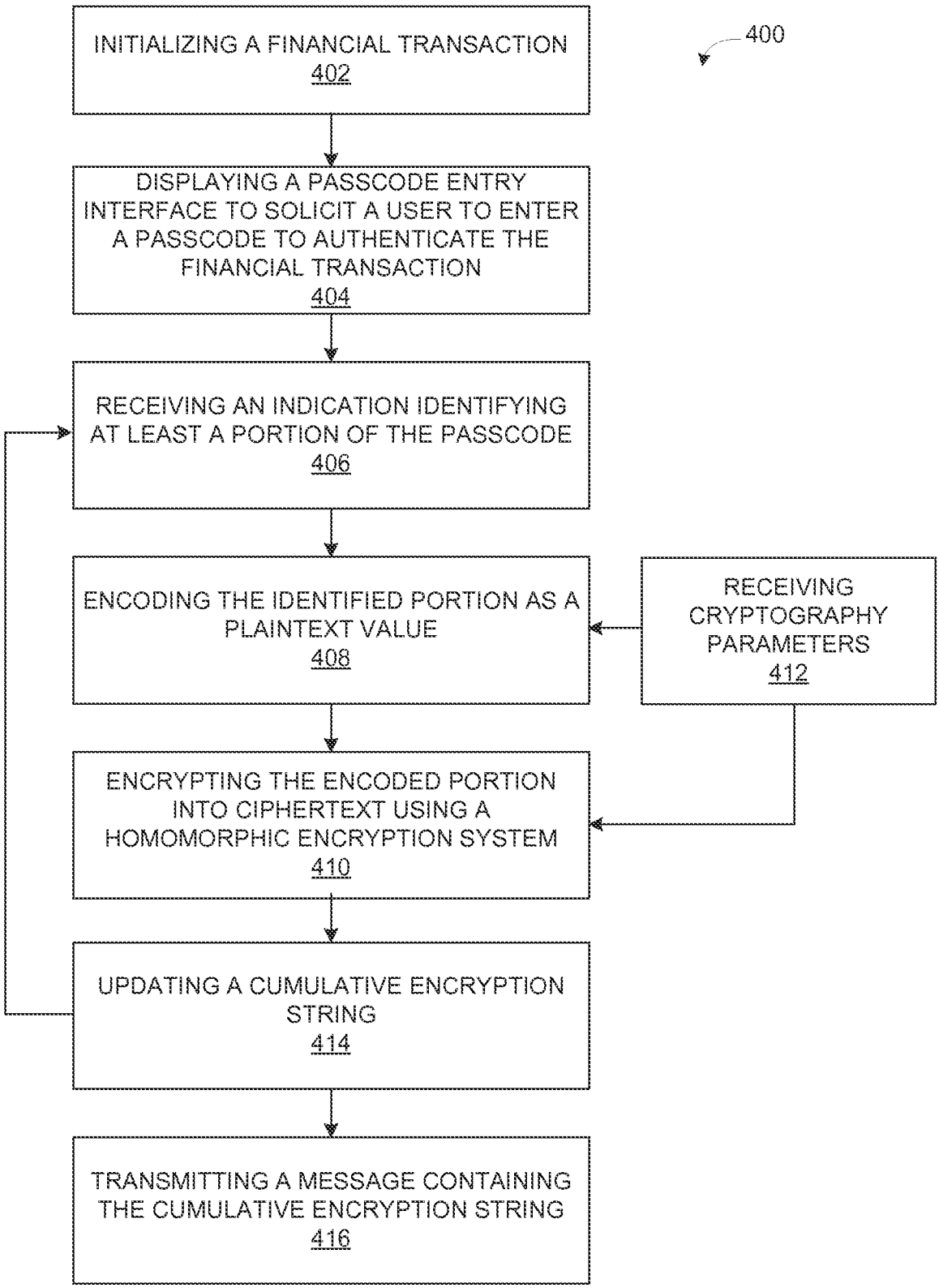
400
INITIALIZING A FINANCIAL TRANSACTION
402
DISPLAYING A PASSCODE ENTRY INTERFACE TO SOLICIT A USER TO ENTER A PASSCODE TO AUTHENTICATE THE FINANCIAL TRANSACTION
404
RECEIVING AN INDICATION IDENTIFYING AT LEAST A PORTION OF THE PASSCODE
406
ENCODING THE IDENTIFIED PORTION AS A PLAINTEXT VALUE
408
RECEIVING CRYPTOGRAPHY PARAMETERS
412
ENCRYPTING THE ENCODED PORTION INTO CIPHERTEXT USING A HOMOMORPHIC ENCRYPTION SYSTEM
410
UPDATING A CUMULATIVE ENCRYPTION STRING
414
TRANSMITTING A MESSAGE CONTAINING THE CUMULATIVE ENCRYPTION STRING
416

FIG. 4

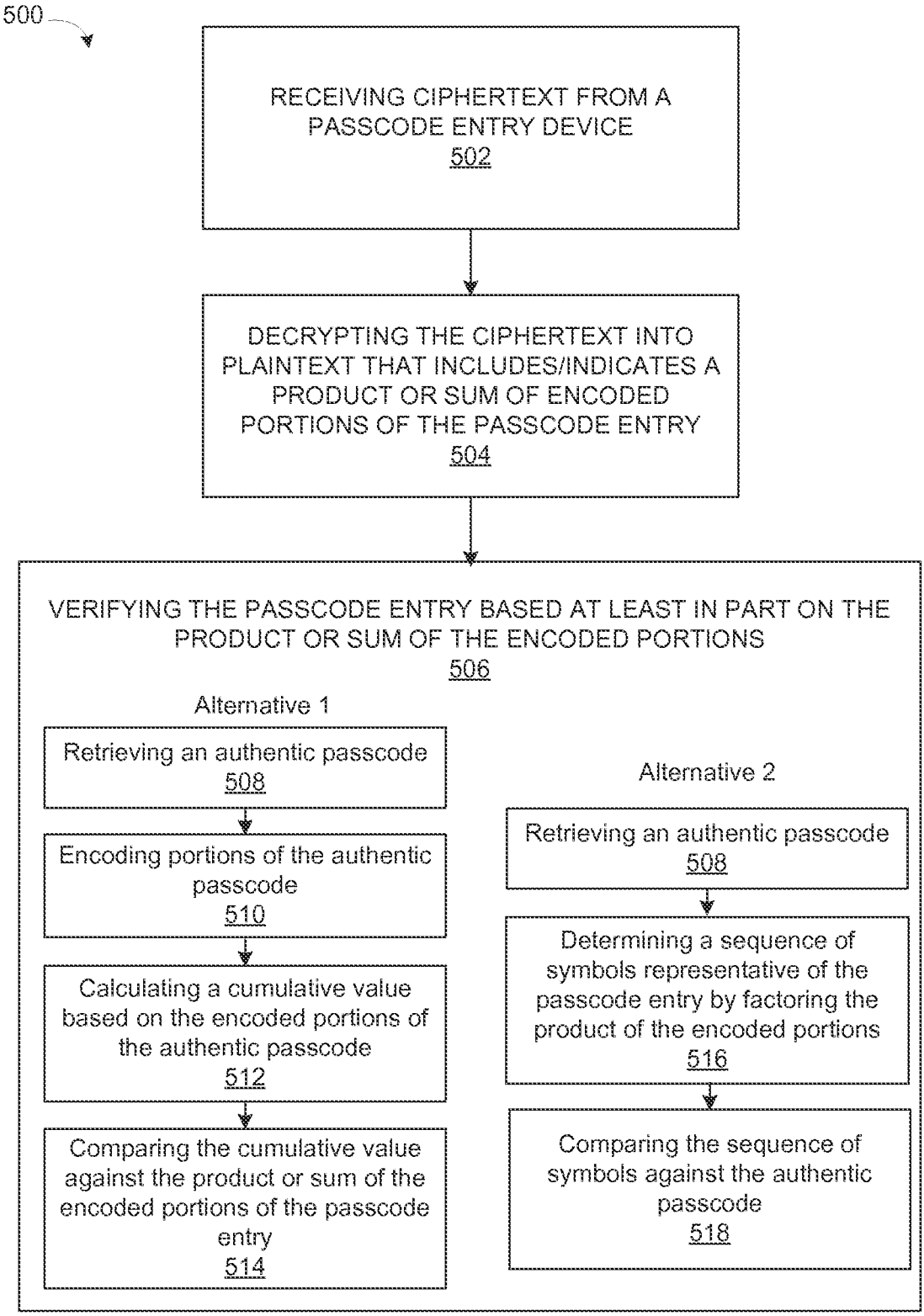
500
RECEIVING CIPHERTEXT FROM A PASSCODE ENTRY DEVICE
502
DECRYPTING THE CIPHERTEXT INTO PLAINTEXT THAT INCLUDES/INDICATES A PRODUCT OR SUM OF ENCODED PORTIONS OF THE PASSCODE ENTRY
504
VERIFYING THE PASSCODE ENTRY BASED AT LEAST IN PART ON THE PRODUCT OR SUM OF THE ENCODED PORTIONS
506
Alternative 1
Retrieving an authentic passcode
508
Encoding portions of the authentic passcode
510
Calculating a cumulative value based on the encoded portions of the authentic passcode
512
Comparing the cumulative value against the product or sum of the encoded portions of the passcode entry
514
Alternative 2
Retrieving an authentic passcode
508
Determining a sequence of symbols representative of the passcode entry by factoring the product of the encoded portions
516
Comparing the sequence of symbols against the authentic passcode
518

FIG. 5

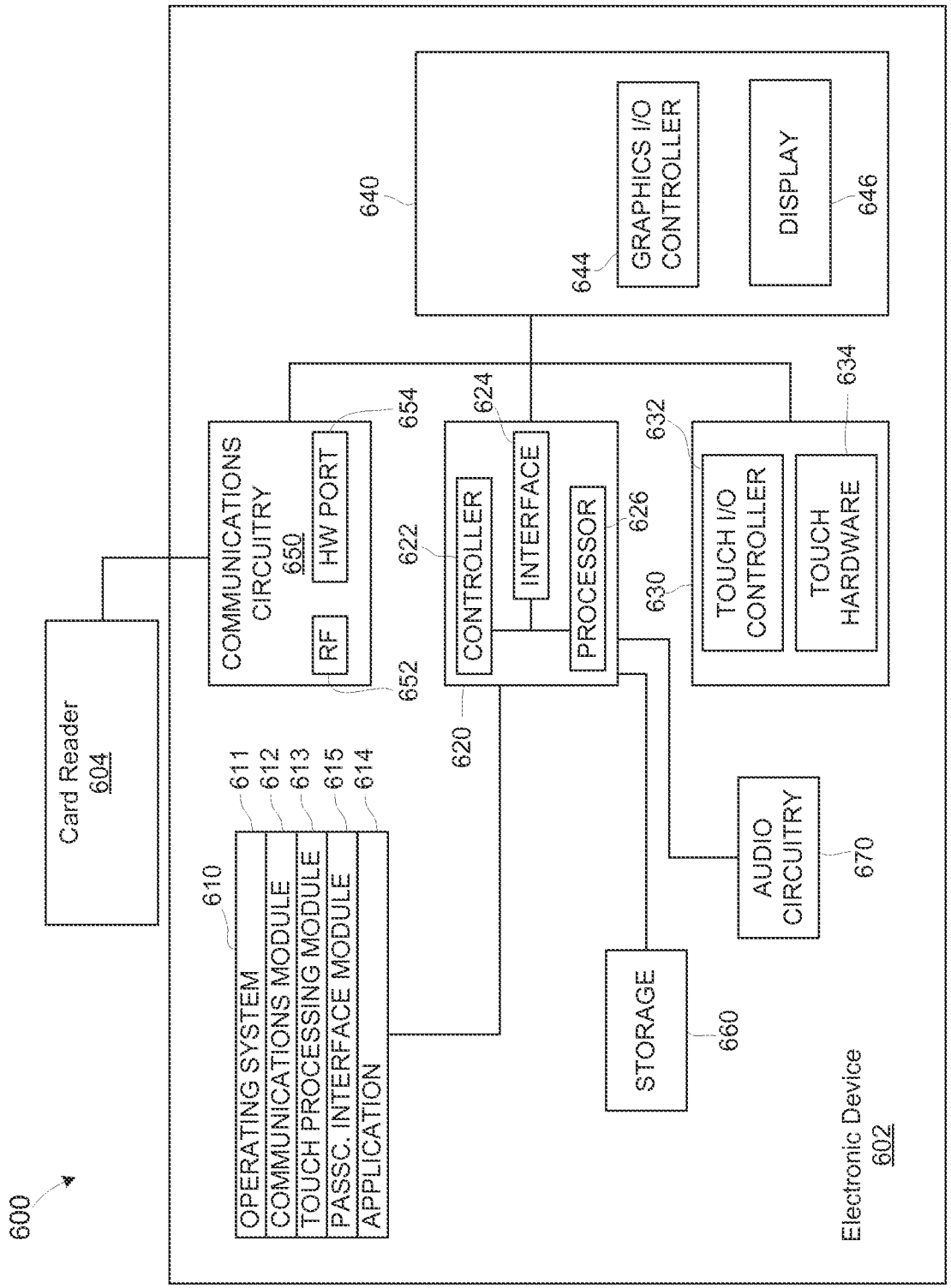
600
Card Reader
604
COMMUNICATIONS CIRCUITRY
650
RF
HW PORT
652
654
620
CONTROLLER
622
INTERFACE
624
PROCESSOR
626
640
GRAPHICS I/O CONTROLLER
644
DISPLAY
646
TOUCH I/O CONTROLLER
632
TOUCH HARDWARE
634
630
AUDIO CIRCUITRY
670
STORAGE
660
610
OPERATING SYSTEM
611
COMMUNICATIONS MODULE
612
TOUCH PROCESSING MODULE
613
PASSC. INTERFACE MODULE
615
APPLICATION
614
Electronic Device
602

FIG. 6