



(12) **DEMANDE DE BREVET CANADIEN**  
**CANADIAN PATENT APPLICATION**

(13) **A1**

(86) **Date de dépôt PCT/PCT Filing Date:** 2022/03/22  
(87) **Date publication PCT/PCT Publication Date:** 2022/11/17  
(85) **Entrée phase nationale/National Entry:** 2023/10/23  
(86) **N° demande PCT/PCT Application No.:** US 2022/021272  
(87) **N° publication PCT/PCT Publication No.:** 2022/240484  
(30) **Priorité/Priority:** 2021/05/11 (US17/317,262)

(51) **Cl.Int./Int.Cl.** **G06Q 20/34** (2012.01),  
**G06F 16/955** (2019.01), **G06F 21/35** (2013.01),  
**G06Q 20/10** (2012.01), **G06Q 20/32** (2012.01)  
(71) **Demandeur/Applicant:**  
CAPITAL ONE SERVICES, LLC, US  
(72) **Inventeurs/Inventors:**  
RULE, JEFFREY, US;  
BERGERON, GEORGE, US;  
NEWMAN, KAITLIN, US;  
HART, COLIN, US  
(74) **Agent:** ROBIC

(54) **Titre : SYSTEMES ET TECHNIQUES POUR UTILISER UN LIEN ACTIF DANS UN LOCALISATEUR UNIFORME DE RESSOURCE EN VUE D'EFFECTUER UN ECHANGE D'ARGENT**  
(54) **Title: SYSTEMS AND TECHNIQUES TO UTILIZE AN ACTIVE LINK IN A UNIFORM RESOURCE LOCATOR TO PERFORM A MONEY EXCHANGE**

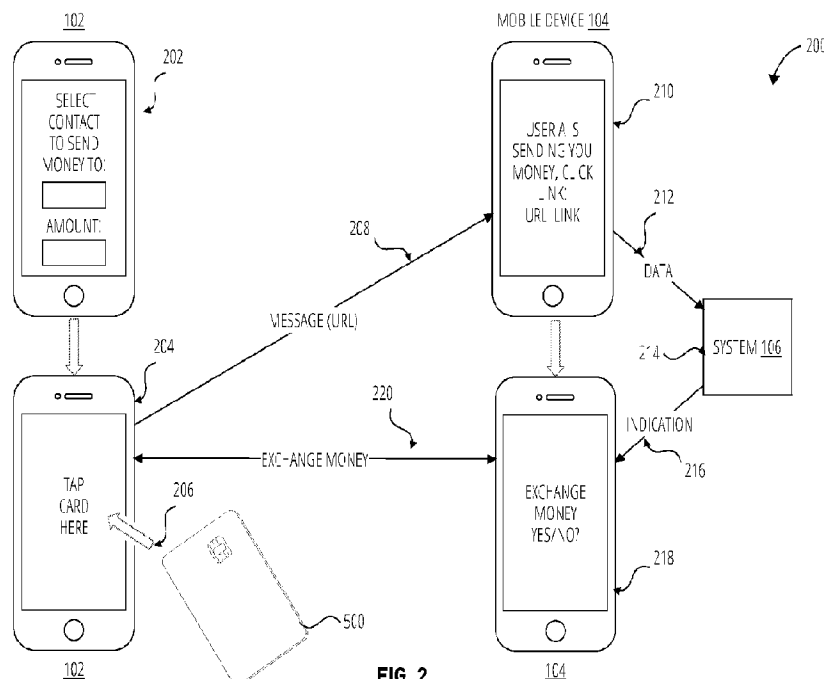


**FIG. 2**

(57) **Abrégé/Abstract:**

A system in which a payment card tapped onto a mobile device is used to authenticate a user sending money to another user or requesting money from another user. When the first user taps a payment card onto their mobile device, a URL with encrypted string is sent to the mobile device of the user receiving the money or the money request. The mobile device of the other user launches a banking app as a result of opening the URL. That app uses the encrypted string to authenticate the first user sending or requesting the money.



**Date Submitted:** 2023/10/23

**CA App. No.:** 3216450

**Abstract:**

A system in which a payment card tapped onto a mobile device is used to authenticate a user sending money to another user or requesting money from another user. When the first user taps a payment card onto their mobile device, a URL with encrypted string is sent to the mobile device of the user receiving the money or the money request. The mobile device of the other user launches a banking app as a result of opening the URL. That app uses the encrypted string to authenticate the first user sending or requesting the money.

**SYSTEMS AND TECHNIQUES TO UTILIZE AN ACTIVE LINK IN A UNIFORM  
RESOURCE LOCATOR TO PERFORM A MONEY EXCHANGE**

**RELATED APPLICATIONS**

[0001] This application claims priority to U.S. Patent Application Serial No. 17/317,262, entitled “SYSTEMS AND TECHNIQUES TO UTILIZE AN ACTIVE LINK IN A UNIFORM RESOURCE LOCATOR TO PERFORM A MONEY EXCHANGE” filed on May 11, 2021. The contents of the aforementioned application are incorporated herein by reference in their entirety.

**BACKGROUND**

[0002] There are many types of electronic payment systems that allow users to exchange money with other users (i.e., peer-to-peer payment transactions). In theory, conventional electronic payment systems provide a convenient method for transferring money between users. Conventional electronic payment systems, however, have several drawbacks that often cause users frustration, confusion and result in an unsatisfactory exchange. One such drawback of conventional electronic payment systems is that they are typically standalone systems to which both the sender and the recipient must subscribe and require a user to go through a number of steps to complete a transaction. In some instances, the sender, recipient, or both may have to go through a time-consuming process of setting up an account and providing sensitive financial information to use a system to send or receive a payment through another banking system. Often times the users may rarely use the electronic payment system after an initial transaction.

[0003] The limited nature of conventional electronic payment systems also adds inconvenience. In particular, the standalone nature of conventional electronic payment systems typically requires that users open a separate application dedicated just to payment transactions in order to send or receive a payment. The inconvenience of the standalone nature of conventional electronic payment systems can discourage users of using such systems. Accordingly, there are a number of disadvantages with conventional electronic payment systems and methods.

**BRIEF SUMMARY**

[0004] Embodiments may include systems, devices, techniques, and methods including a mobile device including a display device, a processor, and memory. The memory may include instructions, that when executed by the processor, cause the processor to receive an

input to perform an electronic exchange of money with a second mobile device of a first user, cause the display device to present an indication to provide a contactless card on the mobile device, receive, from the contactless card, first data, the first data encrypted with a private key of a second user, generate a message comprising a uniform resource locator (URL) comprising the first data to communicate to the second mobile device, the URL configured to cause the second mobile device to perform authentication of the second user with the first data, communicate the message to the second mobile device, and cause the electronic exchange of money based on the authentication of the second user.

**[0005]** Embodiments also include systems, devices, techniques, and methods including a mobile device including an input device, a processor, and memory. The memory may include instructions that when executed by the processor, cause the processor to receive a message comprising a uniform resource locator (URL) comprising first data from a second mobile device, the URL configured to cause the mobile device to perform authentication of a first user with the first data, detect, via the input device, an input to activate the URL, launch an application in response to the input, wherein the application is configured to communicate with a server to perform the authentication of the first user with the first data, receive an indication that the first user is authenticated by the server, and cause performance of an exchange of money based on the first data and the indication that the first user is authenticated.

**[0006]** Embodiments also include systems, devices, techniques, and methods include a computer-implemented method including detecting, by a processor of a first mobile device associated with a first user, a request to exchange money with a second user of a second mobile device, causing, on a display device, a prompt to request the first user associated with the first user device to tap a contactless card on the first user device, receiving, from the contactless card, first data, the first data encrypted with a private key associated with the first user, generating a text message comprising a uniform resource locator (URL) further comprising the first data to communicate to second user device, the URL configured to cause the second mobile device to perform authentication of the first user with the first data, communicating the message to the second mobile device, receiving an indication from the second mobile device indicating the first user is authenticated, and exchanging information with the second mobile device to exchange the money.

#### BRIEF DESCRIPTION OF THE SEVERAL VIEWS OF THE DRAWINGS

**[0007]** FIG. 1 illustrates an example of a system 100 in accordance with embodiments discussed herein.

- [0008] FIG. 2 illustrates an aspect of the subject matter in accordance with one embodiment.
- [0009] FIG. 3 illustrates a routine 300 in accordance with embodiments discussed herein.
- [0010] FIG. 4 illustrates a routine 400 in accordance with embodiments discussed herein.
- [0011] FIG. 5 illustrates a contactless card 500 in accordance with embodiments.
- [0012] FIG. 6 illustrates a transaction card component 600 in accordance with embodiments.
- [0013] FIG. 7 illustrates a sequence flow 700 in accordance with embodiments.
- [0014] FIG. 8 is a diagram of a key system 800 in accordance with embodiments.
- [0015] FIG. 9 is example of a flowchart 900 for generating a cryptogram according to embodiments.
- [0016] FIG. 10 illustrates an example of a flowchart 1000 in accordance with embodiments.
- [0017] FIG. 11 illustrates an example of a 1100 in accordance with embodiments.
- [0018] FIG. 12 illustrates a computer architecture 1200 in accordance with one embodiment.
- [0019] FIG. 13 illustrates a communications architecture 1300 in accordance with one embodiment.

#### DETAILED DESCRIPTION

[0020] Embodiments discussed herein may be generally directed to devices, systems, and techniques to enable users to exchange money in a secure streamlined manner. For example, a user using their mobile device may send money to another user via their mobile device. The exchange may be performed by the sender interacting with a banking application on the mobile device. Specifically, the banking application may be configured to enable a user to enter a person or contact, enter an amount of money, and send the money to the specific person or contact. The sender's banking application may generate a message that includes information to identify the sender and an amount of money. The message may be sent to the recipient's mobile device to initiate the exchange. In some instances, the message may include a Uniform resource locator (URL) embedded with the information. The URL may be invoked by the recipient to automatically launch a banking application on the recipient's mobile device to authenticate the sender and perform the exchange.

[0021] As will be discussed in more detail in the following description, the techniques are more secure and streamlined than previous solutions because they utilize a contactless card

including secure data associated with the sender to authenticate the sender. The secure data may be embedded in the URL and communicated to the recipient's mobile device in the message. The recipient's mobile device may authenticate the sender, either locally or remotely via a server, and then proceed with performing the exchange based on the results of the authentication process.

**[0022]** Additionally, the exchange may occur with a minimal number of actions performed by the sender and recipient. For example, the sender may select or enter a contact, enter an amount of money, and tap their contactless card on the mobile device to perform the exchange. The recipient may read the message and activate the URL, and confirm the exchange of the money. The exchange may occur based on the operations performed by the sender's mobile device, the recipient's mobile device, and a banking system or third-party system. Details are discussed in the following description.

**[0023]** FIG. 1 illustrates an example of a system 100 to perform the operations discussed herein including enabling users to exchange money via mobile devices. The illustrated system 100 includes a limited number of components and devices for discussion purposes. In embodiments, the system 100 includes any number of servers, devices, networking equipment, etc., configured to enable any number of users to send money to other users via devices.

**[0024]** In FIG. 1, the system 100 includes mobile device 102, mobile device 104, and system 106, which all may be coupled via network 108, e.g., the Internet. Examples of a mobile device may include a cellular telephone, a personal digital assistant, a tablet computer, etc., and are configured to communicate in accordance with cellular protocols, wireless protocols, and so forth. In one example, a mobile device may include a housing made of a material, such as metal or plastic, and include components, such as one or more processors, memory, storage, input devices (touchscreen interface, buttons, sensors, microphone, etc.), output devices (display, speaker, vibration device, haptic device, etc.), short-range communication radios, antenna(s), etc.

**[0025]** In the illustrated example, mobile device 102 may communicate with mobile device 104 via network 108, and vice versa. In some instances, the mobile devices may communicate with each other over other networks, such as a local area network, a wireless local area network, a wide area network, etc. The mobile devices may also be configured to establish a secure point-to-point (P2P) connection or a direct connection between each other. The P2P or direct connection may be established utilizing a protocol, such as

Bluetooth Low Energy (BLE), Bluetooth, Wireless Fidelity (WiFi), HaLow (IEEE 802.11ah), Z-wave, or Zigbee.

**[0026]** In embodiments, a mobile device includes software, such as an operating system and applications or apps. The operating system may be any type of operating, such as ANDRIOD®, IOS®, or WINDOWS®, and configured to manage the mobile device's hardware, software resources, and provide services for the other applications. The applications may be software applications design to operate on the mobile device and provide services and experiences for users. Examples of applications include productivity applications, e.g., email applications, messaging applications, phone applications, contact databases, calendar applications, etc. Other applications may include gaming applications, map applications, location-based services, etc.

**[0027]** In embodiments discussed herein, a mobile device may also include a banking application associated with one or more banking accounts and is configured to provide banking services for users including enabling users to send and receive money between each other. The banking application may also be configured to provide additional banking services, e.g., manage accounts, pay bills, review accounts, etc. At a high level, the banking application may be configured to present information to users in one or more graphical user interface (GUI) displays, process user inputs and selections, communicate with other applications, communicate with one or more backend systems (banking server(s)) and services, and so forth to provide the banking services.

**[0028]** The mobile devices 102 and 104 may also be coupled with system 106 via network 108. System 106 may include one or more servers or computing devices, and provide one or more services for mobile applications to send/receive money. In one example, the system 106 may be a banking system configured to perform authentication operations to authenticate a user requesting to send money to another user or requesting money from another user. System 106 may also provide additional services, such as general banking services, account management, payment services, and so forth. Embodiments are not limited in this manner. In some instances, the system 106 may be configured to process the exchange of money between the mobile devices 102 and 104. For example, the system 106 may electronically deduct the money from the sender's account and electronically add the money to the recipient's account. In some instances, the sender may be associated with a first banking system, and the recipient may be associated with a second banking system. The banking systems may communicate data between each other to coordinate the exchange of money. In the following description, FIG. 2 illustrates additional details for performing a transfer of money utilizing system 100.

**[0029]** FIG. 2 is an example of a sequence diagram 200 that may be performed in system 100 to send money from a first mobile device 102 to a second mobile device 104. Note that FIG. 2 illustrates one possible sequence, and in some embodiments, one or more operations may occur before, after, and simultaneously as other operations. Also, as will be discussed in more detail below, a user may request money from another user.

**[0030]** At 202, the mobile device 102 may present one or more GUI displays for a user to enter information to send (or request) money. Specifically, the mobile device 102, including a mobile application, such as the banking application or the web browser, presents one or more GUI displays. The displays may be configured for a user to interact with, e.g., make selections, navigate through options and menus, and perform operations. A GUI display may be configured with any type of input element, e.g., entry fields, drop-down menus, touch-based menus or buttons, etc., and the mobile device 102 is configured to process a user input detected by an input device, e.g., button, scroll wheel, touchscreen interface, etc. In the illustrated example, the banking application presents an area for a user to enter a contact and an amount of money. The text field may include features such as auto-complete, suggestions, and be configured to accept the contact in a number of formats, e.g., email address, phone number, username, etc. Embodiments are not limited to the illustrated specific GUI display, and the display may be another format. For example, the banking application may provide a list of contacts from the contact database on the mobile device 102. In some instances, the list may be ordered based on criteria, such as the frequency of use, closest contacts, friends vs. family, etc. In some instances, mobile applications may present more than one GUI display, e.g., a first GUI display (or page) to enter a contact and a second GUI display to enter an amount. As mentioned above, in some instances, the user may request money from another user. In these instances, the user may be presented with similar GUI displays; however, a GUI display provides an indication that the user is requesting money, not sending money.

**[0031]** At 202, the user may select or enter a contact and provide an amount of money to send (or request). The mobile device 102 may process the inputs to determine a device, e.g., mobile device 104, associated with the selected contact. The mobile device 102 is configured to perform an authentication operation to authenticate the user to send or request money. In the illustrated example, the mobile device 102 performs an authentication utilizing a contactless card associated with the user.

**[0032]** At 204, the mobile device 102, including the banking application, may present a GUI display instructing the user to tap the card on the mobile device 102. In some instances, the GUI display may instruct the user to tap the card on a specific surface, e.g., the display

or the back of the mobile device 102. Tapping the card on the mobile device 102 initiates an exchange of data between mobile device 102 and the card. Both the mobile device 102 and contactless card 500 are configured to communicate with each other utilizing a short-range communication protocol. For example, the contactless card and the mobile device 102 may perform one or more near field communication (NFC) exchanges. As part of the exchange, the contactless card 500 may be configured to communicate information and data to the 102 that can be used to authenticate the user. For example, at 206, the 500 may send a string, e.g., an alphanumeric sequence of characters, that is stored on the 500 and uniquely identifies the user. In some instances, the contactless card 500 may send the string based on a request or read performed by the mobile device 102. Further, the contactless card 500 may encrypt the string using a private key stored on the contactless card 500 and send the string as an encrypted string to the mobile device 102. The private key may be part of a key pair and associated with a public key that may be used to authenticate the user. In embodiments, the private key may be a proprietary key generated by and associated with the card provider. In other instances, the private key may be associated with performing operations in accordance with the Europay, Mastercard, Visa (EMV) standard.

**[0033]** At 206, the mobile device 102 may process the information received from the contactless card 500 and generate a message to send to a device (mobile device 104) associated with the selected contact. In embodiments, the mobile device 102 may determine to send a message to a mobile device using an identifier or attribute for the contact, e.g., a phone number, username, an electronic mail (email) address, etc. In the illustrated example, the mobile device 102 generates a text message in accordance with a text messaging protocol, e.g., SMS, MMS, RCS, etc. However, in embodiments, other formats may be utilized, e.g., e-mail, real-time messaging, etc., and use protocols such as user datagram protocol (UDP) and transmission control protocol (TCP).

**[0034]** The message may include information to indicate to the recipient that another user wishes to send (or request) money, information received from the contactless card 500, and information based on the user inputs, e.g., selected contact and amount entered. In embodiments, the message may also include an active link or active URL. The active URL may be used by the recipient to facilitate the exchange of money. For example, the recipient may select or activate the URL, which in-turn causes a mobile application to launch of the recipient's device. The active URL may be a deep link to launch a banking app or webpage on the mobile device 104, for example. In embodiments, the URL may be embedded with the information including one or more of the encrypted string generated by the contactless card 500, an identifier of the recipient, an identifier of the amount of money, and an

identifier of the sender. The URL may be in the format in accordance with one or more operating systems and APIs. For example, the message may include a URL in the format in accordance with standards and APIs for the operating systems IOS, ANDROID, and/or MICROSOFT. Example formats may include "appam://<string> <recipient identifier> <amount identifier>," or "https://site.com/<string>/<recipient identifier>/<amount identifier>," but embodiments are not limited this manner.

**[0035]** At 208, the mobile device 102 may send the message to the mobile device 104, i.e., the device associated with the contact. The message may be sent in accordance with one or more communication protocols based on the type of message. At 210, the mobile device 104 may receive the message and present the message on a display device. As mentioned, the message may include an active URL selectable by the user of mobile device 104. The mobile device 104 may present the message in a GUI display of a messaging application. A user may interact with the message and select or click on the URL. The mobile device 104 is configured to detect the user selecting and/or activating the URL and to launch the application or webpage based on the information in the URL. The mobile device 104 may launch an application, such as a banking application or a webpage in a web browser. In embodiments, the mobile device 104, including the application, is also configured to process the information embedded in the URL. For example, the application may process the encrypted string and pass the encrypted string through an API to a system 106 to authenticate the sender at 212. In embodiments, the application may also send the other data to the system 106 including the identifier of the sender, the identifier of the recipient, and the amount.

**[0036]** At 214, the system 106 may authenticate the sender based on the information received from the mobile device 104. For example, the system 106 may utilize the identifier of the sender to perform a lookup to determine a stored authenticated string from a database or a data store. The system 106 may also determine a public key associated with the sender and attempt to decrypt the string. The decrypted string may then be compared to the authenticate string to determine if they match and to authenticate the sender. If the system 106 determines that the strings match, the sender is authenticated, and if they do not match, the sender is not authenticated.

**[0037]** At 216, the system 106 may return a result of the comparison or indication as to whether the sender is authenticated. The indication may be communicated as a result to the API call to perform the authentication operation. At 218, the mobile device 104 may present the indication to the user in a GUI display. In some instances, the mobile device 104 may display an option for the user to accept or decline the money. The mobile device 104 may

process a user input to decline the exchange and cease operations or not perform additional operations to exchange the money. Alternatively, the mobile device 104 may process a user input to accept or proceed with the exchange of money and perform additional operations to perform the exchange at 220. For example, the mobile device 104 may provide an indication or communication to mobile device 102 to proceed with sending the money. The mobile device 102 may communicate with one or more backend servers, such as a banking server, an indication to electronically transfer the money to an account associated with the recipient, and mobile device 104. In other instances, the application may communicate with a banking server an indication to proceed with the transfer of money.

**[0038]** In some instances, a secure connection may be established between the mobile device 102 and mobile device 104 to exchange money. The secure connection may be a P2P localized connection between the mobile devices 102 and 104. The mobile device 102 may send the money as an electronic token to the mobile device 104. The mobile device 104 may process the token and communicate with a banking server to provide an indication of the transfer. The token may include information that may be used by the banking server to complete an exchange. The information may include an identifier of the sender, an identifier of the recipient, an amount of money, and so forth. Embodiments are not limited in this manner.

**[0039]** FIG. 3 illustrates an example routine 300 that may be performed by system 100. Specifically, routine 300 may be performed by a mobile device configured to send (or request) money to another user via their mobile device.

**[0040]** In block 302, the routine 300 includes detecting, by a processor of a first mobile device associated with a first user, a request to exchange money with a second user of a second mobile device. For example, the mobile device may include a banking application that may present one or more options for the user to send or request money from another person via a P2P exchange. The user, via the banking application, may provide an input to select whether to send or request money. The user may also provide additional information for the exchange. For example, the application may enable a user to enter a contact or person to send or request the money from and an amount of money for the exchange. The mobile device may receive an input selection via a touchscreen interface to send money to another user, for example. In one example, the input selection may include a selection of a contact from a contact list stored on the mobile device. In other instances, the user may type the contact in a field using a keyboard, such as a physical keyboard or a virtual keyboard. In other instances, the user may provide other identifying information, such as a phone

number, an email address, a username, etc. Embodiments may not be limited to these examples.

**[0041]** To perform the exchange of money, the user must be authenticated to the second mobile device and second user. Typical authentication methods may be utilized including providing a password, performing a biometric analysis, sending a code in a text message, and so forth. However, embodiments discussed herein may provide additional security by utilizing a contactless card. In the illustrated routine 300, the first user is authenticated utilizing a contactless card associated with the first user. Using the contactless card to authenticate the user provides advantages over other technologies by ensuring that the user has possession of the card and the contactless card includes a token or string that is unique to the user. As will be discussed in more detail, the token or string may be used to authenticate the user via a third-party, e.g., a banking system or other third-party authentication system. In some instances, the second user may authenticate the first user utilizing a local authenticated string and a public key associated with the first user.

**[0042]** The authentication operation performed with the contactless card includes the mobile device exchanging information with the contactless card, which may be initiated by the user bringing the contactless card within a communication range (NFC communication range is ~10cm) of the mobile device. At block 304, the routine 300 includes causing, on a display device, a prompt to request the first user associated with the first mobile device to tap the contactless card on the first mobile device. When the contactless card comes within a sufficient range of the mobile device, a connection may be established between the device and the card, the mobile device may be configured to exchange data with the card, and the card may be configured to exchange data with mobile device. Instructing the user to tap the card on the mobile device ensures that the card is brought within the communication range. However, other embodiments may include instructing the user to bring the contactless card close or near the mobile device, and the mobile device may provide an indication, e.g., displaying a message or icon, lighting an LED, etc., when the card is within communication range.

**[0043]** At block 306, the routine 300 includes receiving data from the contactless card. The data may be encrypted by the contactless card with a private key, e.g., a proprietary private key or EMV based private key, that is associated with the first user and securely stored on the contactless card. In some instances, the contactless card may communicate the data to the mobile device in one or more NFC Data Exchange Format (NDEF) message(s). The encrypted data may include information to identify the first user, such as a token or string, that is also securely stored in the contactless card. The mobile device may provide an

indication to the user when the exchange is complete and that a message is to be sent to the second mobile device.

**[0044]** In embodiments, the mobile device may generate and send an indication of the exchange of money to the second mobile device associated with the second user. For example, at block 308, the routine 300 includes generating a text message comprising text including information of the exchange, a uniform resource locator (URL), and the data from the contactless card. The text may identify the sender and an amount of money. The URL may be an active URL or deep link, that when activated by the recipient, causes the receiving mobile device to launch an application and perform an authentication of the first user using the data from the contactless card. In embodiments, the mobile device may generate the message and embed the data in the message and/or in the URL. For example, the data including the encrypted string, may be incorporated into the URL as a parameter. Thus, when the URL is activated, the encrypted string is passed to the launched application. In some instances, the mobile device may include additional information in the URL including an identifier to identify the sender, an identifier to identify the recipient, and a specified amount of money to exchange. The identifiers and specified amount may also be passed to the launched application as parameters and may be used by the application to process the exchange.

**[0045]** At block 310, the routine 300 includes communicating the message to the second mobile device. The message may be a text message and communicated to the second mobile device based on an associated phone number. The text message may be sent in accordance with any type of messaging protocol, such as SMS, MMS, and RCS. In some instances, other mediums may be used to send the message, email, chat applications, and so forth. Embodiments are not limited in this manner.

**[0046]** In embodiments, the second mobile device may receive the message and process the message. For example, the second mobile device may present the message to the second user on a display device in GUI display. When the URL is activated by the second user, the second mobile device may perform additional operations including authenticating the first user. In some instances, the authentication may be performed locally by the second mobile device; and in other instances, the second mobile device may utilize a banking system or other third-party system to authenticate the first user. The second mobile device may determine a result of the authentication operation and provide the result to the first mobile device.

**[0047]** At block 312, the routine 300 includes receiving an indication from the second mobile device indicating whether the first user is authenticated or not authenticated. The indication may be received from the second mobile device in a text message or other communication medium. The indication may include a result of the authentication and an indication as to whether the second user accepts or denies the exchange of money. If the first user cannot be authenticated or the second user denies the exchange, the first and second mobile device may cease operations to exchange the money. Alternatively, if the first user is authenticated and the second user accepts the exchanges, the devices may continue to perform operations to exchange the money. At block 314, the routine 300 includes exchanging information with the second mobile device to exchange the money.

**[0048]** In some instances, e.g., when the first and second mobile devices are remote from each other, e.g., cannot form a localized P2P connection, the mobile devices may perform the exchange of money via a banking system. In an example, the sender and the recipient may have accounts with the same bank and banking system. The banking applications on their respective devices may also be configured to communicate directly with the banking system. In this example, the second mobile device, if the first user is authenticated and the second user accepts the exchange, may communicate an indication to the banking system indicating that the exchange is accepted. The indication may be sent via an API communication from the banking application on the second mobile device to the banking system. The banking system may then electronically perform the transaction by deducting the money from the sender's account and adding the money to the recipient's account. In some instances, the first mobile device may initiate and cause the exchange of money upon reception of the indication that the first user is authenticated and the second user approves of the exchange. Similarly, the first mobile device may communicate with a banking system via the banking application on the first mobile to cause the exchange. The banking system may then perform the exchange as discussed above, e.g., electronically deducting the money from the sender's account and adding the money to the recipient's account.

**[0049]** In embodiments, the first and second users may utilize different banks having different banking systems, and the banking systems may communicate with each other to perform the exchange of money. For example, the first user may provide an identifier to identify the first user's banking system to the second mobile device. Once the exchange is approved, the second mobile device may communicate information to a banking system associated with the second user. The second user's banking system may then communicate with the first user's banking system to perform the exchange. In the other instances, the second mobile device may communicate information to identify the second user's banking

system to the first mobile device, and the first mobile device may initiate the exchange by communicating with the first user's banking system.

**[0050]** In some embodiments, the banking application may be a third-party banking application, e.g., an application not operated directly by a banking system. The third-party banking application may be configured on each of the mobile devices including having one or more banking accounts configured to operate with the third-party banking application. Once the exchange is approved, the third-party applications on each of the sender's and recipient's devices may communicate with their respective banks to perform the exchange. In some instances, the third-party application including a third-party system may communicate with the sender's banking system to deduct the money from the sender's account. The third-party system may hold the money for an amount of time to earn a fee or interest and then send the money to the recipient's account. Embodiments are not limited in this manner.

**[0051]** In embodiments, the first and second mobile device may communicate data over a P2P secure connection to exchange the money. For example, in response to the first user being authenticated and the second user approving the exchange, the second mobile device may communicate with the first mobile device to establish a P2P connection. The connection may be established in accordance with a particular P2P protocol, such as BLE, Bluetooth, ZigBee, Wi-Fi, etc. Once the connection is established, the first and/or the second mobile device may initiate an exchange of money by communicating a message. For example, the second mobile device may communicate a request for data or token to exchange the money to the first mobile device. The first mobile device may send data, which may include token or secure information having an indication of the amount of money to exchange, to the second mobile device. The second mobile device may communicate with a banking system to perform the exchange between the banking accounts of the first user and the second user. In some instances, the data may also include an identifier of the sender, an identifier of the recipient, bank account information for the sender, etc. The data may be sent securely and be encrypted.

**[0052]** FIG. 4 illustrates an example routine 400 that may be performed by system 100. Specifically, routine 400 may be performed by a mobile device receiving a communication from another mobile device, the communication indicates that another user wishes to send (or request) money.

**[0053]** At block 402, the routine 400 includes receiving, by a first mobile device associated with a first user, a message comprising text indicating that a user wants to send

(or request) money, and a uniform resource locator (URL) comprising data. The text may identify the sender of the message and the amount of money. In embodiments, the message may be received from a second mobile device associated with a second user via a communication protocol, such as SMS, MMS, RCS, or another protocol.

**[0054]** In embodiments, the URL may be an active URL or deep-link, that when activated by the recipient, causes the second mobile device to launch an application and perform an authentication of the second user using the data received from the first mobile device and a contactless card. In embodiments, the data may include a token or encrypted string and be embedded in the URL. The data may be passed to the application, when launched, as a parameter that can be used by the application to perform the authentication of the second user. Other information may also be embedded in the URL, such as identifiers of the first and second users and an amount of money to exchange.

**[0055]** At block 404, the routine 400 includes detecting, by the first mobile device, an input to activate the URL. For example, the first mobile device may include an input device, such as a touchscreen interface and the user may select or “click” on the URL.

Embodiments are not limited to this example, and the input may be provided via another input device, e.g., a button, a microphone (voice instruction), and so forth.

**[0056]** In block 406, the routine 400 includes launching, by the first mobile device, an application in response to the input, wherein the application is configured to communicate with a server to perform the authentication of the second user with the data. As mentioned, the URL may be a deep link, which may point to a location or provides instructions that are processed by the operating system to launch the application. For example, the URL may indicate a specific application to launch or point to a website to launch in a web browser. In some instances, the URL may include an indication or specify a landing spot or location to launch the application. Thus, the application or web browser may open to a GUI display displaying information for the exchange of money, e.g., sender/request's name, contact information, an amount of money, etc. In embodiments, the URL may include the data, such as the token or encrypted string, an identifier to identify the sender, an identifier to identify the recipient, and an indication of the amount of money that may be passed to the application as parameters when the application is launched.

**[0057]** In embodiments, the first mobile device, including the launched application, may initiate an authentication operation to authenticate the second user when the URL is activated. Specifically, the launched application may utilize the information embedded in the URL, e.g., the token or encrypted string, to authenticate the user. In some instances, the

application may perform the authentication operation locally by decrypting the encrypted string and comparing the decrypted string to an authenticated string. In some instances, the first mobile device may first obtain the authenticated string from a banking system in a secure communication.

**[0058]** In the illustrated example, the second mobile device including the application, may send the encrypted string to a server of the banking system or a third-party system to perform the authentication operation. The application may send the data including the encrypted string to the server via API communication(s). The server may decrypt the encrypted string based on a public key associated with the second user, and compare the decrypted string to an authenticated string stored on the banking system. If the strings match, the second user may be authenticated, and vice versa.

**[0059]** At block 408, the routine 400 includes receiving, by the first mobile device, an indication of whether the second user is authenticated by the server. The indication may be returned to the application in response to the API communication(s). In some instances, the GUI display may display an indication that an authentication operation is being performed to authenticate the second user and the results of the authentication operation once they are determined. If the user cannot be authenticated, the operations to exchange money may cease. If the user is authenticated, the application may request the first user to confirm that they want to proceed with the exchange of money.

**[0060]** In response to the second user being authenticated and the first user confirming the transaction, at block 410, the routine 400 includes causing, by the first mobile device, performance of an exchange of money. As discussed above in FIG. 3, the first mobile device may communicate with one or more banking systems to perform the exchange of money. In other instances, the first mobile device may send an indication that authentication was successful and that the first user approves the exchange to the second mobile device (sender's device). The second mobile device may communicate with the one or more banking systems to perform the exchange. In even other instances, a P2P connection may be established between the first and second mobile devices and data may be exchanged between the devices to perform the exchange.

**[0061]** FIG. 5 illustrates an example configuration of a contactless card 500, which may include a contactless card, a payment card, such as a credit card, debit card, or gift card, issued by a service provider as displayed as service provider indicia 502 on the front or back of the contactless card 500. In some examples, the contactless card 500 is not related to a payment card, and may include, without limitation, an identification card. In some

examples, the transaction card may include a dual interface contactless payment card, a rewards card, and so forth. The contactless card 500 may include a substrate 508, which may include a single layer or one or more laminated layers composed of plastics, metals, and other materials. Exemplary substrate materials include polyvinyl chloride, polyvinyl chloride acetate, acrylonitrile butadiene styrene, polycarbonate, polyesters, anodized titanium, palladium, gold, carbon, paper, and biodegradable materials. In some examples, the contactless card 500 may have physical characteristics compliant with the ID-1 format of the ISO/IEC 7816 standard, and the transaction card may otherwise be compliant with the ISO/IEC 14443 standard. However, it is understood that the contactless card 500 according to the present disclosure may have different characteristics, and the present disclosure does not require a transaction card to be implemented in a payment card.

**[0062]** The contactless card 500 may also include identification information 506 displayed on the front and/or back of the card, and a contact pad 504. The contact pad 504 may include one or more pads and be configured to establish contact with another client device, such as an ATM, a user device, smartphone, laptop, desktop, or tablet computer via transaction cards. The contact pad may be designed in accordance with one or more standards, such as ISO/IEC 7816 standard, and enable communication in accordance with the EMV protocol. The contactless card 500 may also include processing circuitry, antenna and other components as will be further discussed in FIG. 6. These components may be located behind the contact pad 504 or elsewhere on the substrate 508, e.g. within a different layer of the substrate 508, and may electrically and physically coupled with the contact pad 504. The contactless card 500 may also include a magnetic strip or tape, which may be located on the back of the card (not shown in FIG. 5). The contactless card 500 may also include a Near-Field Communication (NFC) device coupled with an antenna capable of communicating via the NFC protocol. Embodiments are not limited in this manner.

**[0063]** As illustrated in FIG. 5, the contact pad 504 of contactless card 500 may include processing circuitry 616 for storing, processing, and communicating information, including a processor 602, a memory 604, and one or more interface(s) 606. It is understood that the processing circuitry 616 may contain additional components, including processors, memories, error and parity/CRC checkers, data encoders, anticollision algorithms, controllers, command decoders, security primitives and tamper proofing hardware, as necessary to perform the functions described herein.

**[0064]** The memory 604 may be a read-only memory, write-once read-multiple memory or read/write memory, e.g., RAM, ROM, and EEPROM, and the contactless card 500 may include one or more of these memories. A read-only memory may be factory programmable

as read-only or one-time programmable. One-time programmability provides the opportunity to write once then read many times. A write once/read-multiple memory may be programmed at a point in time after the memory chip has left the factory. Once the memory is programmed, it may not be rewritten, but it may be read many times. A read/write memory may be programmed and re-programmed many times after leaving the factory. A read/write memory may also be read many times after leaving the factory. In some instances, the memory 604 may be encrypted memory utilizing an encryption algorithm executed by the processor 602 to encrypted data.

**[0065]** The memory 604 may be configured to store one or more applet(s) 608, one or more counter(s) 610, a customer identifier 614, such as a token or string, and the account number(s) 612, which may be virtual account numbers. The one or more applet(s) 608 may comprise one or more software applications configured to execute on one or more contactless cards, such as a Java® Card applet. However, it is understood that applet(s) 608 are not limited to Java Card applets, and instead may be any software application operable on contactless cards or other devices having limited memory. The one or more counter(s) 610 may comprise a numeric counter sufficient to store an integer. The customer identifier 614 may comprise a unique alphanumeric identifier, token, or string assigned to a user of the contactless card 500, and the identifier may distinguish the user of the contactless card from other contactless card users. In some examples, the customer identifier 614 may identify both a customer and an account assigned to that customer and may further identify the contactless card 500 associated with the customer's account. In some instances, the memory 604 may include the customer identifier 614 and one or more tokens or additional strings that may be used to identify the customer. As stated, the account number(s) 612 may include thousands of one-time use virtual account numbers associated with the contactless card 500. An applet(s) 608 of the contactless card 500 may be configured to manage the account number(s) 612 (e.g., to select an account number(s) 612, mark the selected account number(s) 612 as used, and transmit the account number(s) 612 to a mobile device for auto filling by an auto filling service.

**[0066]** The processor 602 and memory elements of the foregoing exemplary embodiments are described with reference to the contact pad 504, but the present disclosure is not limited thereto. It is understood that these elements may be implemented outside of the contact pad 504 or entirely separate from it, or as further elements in addition to processor 602 and memory 604 elements located within the contact pad 504.

**[0067]** In some examples, the contactless card 500 may comprise one or more antenna(s) 618. The one or more antenna(s) 618 may be placed within the contactless card 500 and

around the processing circuitry 616 of the contact pad 504. For example, the one or more antenna(s) 618 may be integral with the processing circuitry 616 and the one or more antenna(s) 618 may be used with an external booster coil. As another example, the one or more antenna(s) 618 may be external to the contact pad 504 and the processing circuitry 616.

**[0068]** In an embodiment, the coil of contactless card 500 may act as the secondary of an air core transformer. The terminal may communicate with the contactless card 500 by cutting power or amplitude modulation. The contactless card 101 may infer the data transmitted from the terminal using the gaps in the contactless card's power connection, which may be functionally maintained through one or more capacitors. The contactless card 500 may communicate back by switching a load on the contactless card's coil or load modulation. Load modulation may be detected in the terminal's coil through interference. More generally, using the antenna(s) 618, processor 602, and/or the memory 604, the contactless card 101 provides a communications interface to communicate via NFC, Bluetooth, and/or Wi-Fi communications.

**[0069]** As explained above, contactless card 500 may be built on a software platform operable on smart cards or other devices having limited memory, such as JavaCard, and one or more or more applications or applets may be securely executed. Applet(s) 608 may be added to contactless cards to provide a one-time password (OTP) for multifactor authentication (MFA) in various mobile application-based use cases. Applet(s) 608 may be configured to respond to one or more requests, such as near field data exchange requests, from a reader, such as a mobile NFC reader (e.g., of a mobile device or point-of-sale terminal), and produce an NDEF message that comprises a cryptographically secure OTP encoded as an NDEF text tag.

**[0070]** One example of an NDEF OTP is an NDEF short-record layout (SR=1). In such an example, one or more applet(s) 608 may be configured to encode the OTP as an NDEF type 4 well known type text tag. In some examples, NDEF messages may comprise one or more records. The applet(s) 608 may be configured to add one or more static tag records in addition to the OTP record.

**[0071]** In some examples, the one or more applet(s) 608 may be configured to emulate an RFID tag. The RFID tag may include one or more polymorphic tags. In some examples, each time the tag is read, different cryptographic data is presented that may indicate the authenticity of the contactless card. Based on the one or more applet(s) 608, an NFC read

of the tag may be processed, the data may be transmitted to a server, such as a server of a banking system, and the data may be validated at the server.

**[0072]** In some examples, the contactless card 500 and server may include certain data such that the card may be properly identified. The contactless card 500 may include one or more unique identifiers, e.g., tokens or strings (not pictured). Each time a read operation takes place, the counter(s) 610 may be configured to increment. In some examples, each time data from the contactless card 500 is read (e.g., by a mobile device), the counter(s) 610 is transmitted to the server for validation and determines whether the counter(s) 610 are equal (as part of the validation) to a counter of the server.

**[0073]** The one or more counter(s) 610 may be configured to prevent a replay attack. For example, if a cryptogram has been obtained and replayed, that cryptogram is immediately rejected if the counter(s) 610 has been read or used or otherwise passed over. If the counter(s) 610 has not been used, it may be replayed. In some examples, the counter that is incremented on the card is different from the counter that is incremented for transactions. The contactless card 101 is unable to determine the application transaction counter(s) 610 since there is no communication between applet(s) 608 on the contactless card 500.

**[0074]** In some examples, the counter(s) 610 may get out of sync. In some examples, to account for accidental reads that initiate transactions, such as reading at an angle, the counter(s) 610 may increment but the application does not process the counter(s) 610. In some examples, when the mobile device 10 is woken up, NFC may be enabled and the device 110 may be configured to read available tags, but no action is taken responsive to the reads.

**[0075]** To keep the counter(s) 610 in sync, an application, such as a background application, may be executed that would be configured to detect when the mobile device 110 wakes up and synchronize with the server of a banking system indicating that a read that occurred due to detection to then move the counter(s) 610 forward. In other examples, Hashed One Time Password may be utilized such that a window of mis-synchronization may be accepted. For example, if within a threshold of 10, the counter(s) 610 may be configured to move forward. But if within a different threshold number, for example within 10 or 1200, a request for performing re-synchronization may be processed which requests via one or more applications that the user tap, gesture, or otherwise indicate one or more times via the user's device. If the counter(s) 610 increases in the appropriate sequence, then it possible to know that the user has done so.

**[0076]** The key diversification technique described herein with reference to the counter(s) 610, master key, and diversified key, is one example of encryption and/or decryption a key diversification technique. This example key diversification technique should not be considered limiting of the disclosure, as the disclosure is equally applicable to other types of key diversification techniques.

**[0077]** During the creation process of the contactless card 500, two cryptographic keys may be assigned uniquely per card. The cryptographic keys may comprise symmetric keys which may be used in both encryption and decryption of data. Triple DES (3DES) algorithm may be used by EMV and it is implemented by hardware in the contactless card 500. By using the key diversification process, one or more keys may be derived from a master key based upon uniquely identifiable information for each entity that requires a key.

**[0078]** In some examples, to overcome deficiencies of 3DES algorithms, which may be susceptible to vulnerabilities, a session key may be derived (such as a unique key per session) but rather than using the master key, the unique card-derived keys and the counter may be used as diversification data. For example, each time the contactless card 101 is used in operation, a different key may be used for creating the message authentication code (MAC) and for performing the encryption. This results in a triple layer of cryptography. The session keys may be generated by the one or more applets and derived by using the application transaction counter with one or more algorithms (as defined in EMV 4.3 Book 2 A1.3.1 Common Session Key Derivation).

**[0079]** Further, the increment for each card may be unique, and assigned either by personalization, or algorithmically assigned by some identifying information. For example, odd numbered cards may increment by 2 and even numbered cards may increment by 5. In some examples, the increment may also vary in sequential reads, such that one card may increment in sequence by 1, 3, 5, 2, 2, ... repeating. The specific sequence or algorithmic sequence may be defined at personalization time, or from one or more processes derived from unique identifiers. This can make it harder for a replay attacker to generalize from a small number of card instances.

**[0080]** The authentication message may be delivered as the content of a text NDEF record in hexadecimal ASCII format. In another example, the NDEF record may be encoded in hexadecimal format.

**[0081]** FIG. 7 is a timing diagram illustrating an example sequence for providing an exchange of data according to one or more embodiments of the present disclosure. Sequence

flow 700 may include contactless card 500 and a mobile device, which may include an application 702 and processor 704.

**[0082]** At line 708, the application 702 communicates with the contactless card 500 (e.g., after being brought near the contactless card 500). Communication between the application 702 and the contactless card 500 may involve the contactless card 500 being sufficiently close to a card reader (not shown) of the mobile device to enable NFC data transfer between the application 702 and the contactless card 500.

**[0083]** At line 706, after communication has been established between mobile device and contactless card 500, contactless card 500 generates a message authentication code (MAC) cryptogram. In some examples, this may occur when the contactless card 500 is read by the application 702. In particular, this may occur upon a read, such as an NFC read, of a near field data exchange (NDEF) tag, which may be created in accordance with the NFC Data Exchange Format. For example, a reader application, such as application 702, may transmit a message, such as an applet select message, with the applet ID of an NDEF producing applet. Upon confirmation of the selection, a sequence of select file messages followed by read file messages may be transmitted. For example, the sequence may include “Select Capabilities file”, “Read Capabilities file”, and “Select NDEF file”. At this point, a counter value maintained by the contactless card 500 may be updated or incremented, which may be followed by “Read NDEF file.” At this point, the message may be generated which may include a header and a shared secret, such as a token or string. Session keys may then be generated. The MAC cryptogram may be created from the message, which may include the header and the shared secret. The MAC cryptogram may then be concatenated with one or more blocks of random data, and the MAC cryptogram and a random number (RND) may be encrypted with the session key. Thereafter, the cryptogram and the header may be concatenated, and encoded as ASCII hex and returned in NDEF message format (responsive to the “Read NDEF file” message).

**[0084]** In some examples, the MAC cryptogram may be transmitted as an NDEF tag, and in other examples the MAC cryptogram may be included with a uniform resource indicator (e.g., as a formatted string). In some examples, application 702 may be configured to transmit a request to contactless card 500, the request comprising an instruction to generate a MAC cryptogram.

**[0085]** At line 710, the contactless card 500 sends the MAC cryptogram to the application 702. In some examples, the transmission of the MAC cryptogram occurs via NFC, however, the present disclosure is not limited thereto. In other examples, this communication may

occur via Bluetooth, Wi-Fi, or other means of wireless data communication. At line 712, the application 702 communicates the MAC cryptogram to the processor 704.

**[0086]** At line 714, the processor 704 verifies the MAC cryptogram pursuant to an instruction from the application 122. For example, the MAC cryptogram may be verified, as explained below. In some examples, verifying the MAC cryptogram may be performed by a device other than mobile device, such as a server of a banking system in data communication with the mobile device. For example, processor 704 may output the MAC cryptogram for transmission to the server of the banking system, which may verify the MAC cryptogram. In some examples, the MAC cryptogram may function as a digital signature for purposes of verification. Other digital signature algorithms, such as public key asymmetric algorithms, e.g., the Digital Signature Algorithm and the RSA algorithm, or zero knowledge protocols, may be used to perform this verification.

**[0087]** FIG. 8 illustrates a diagram of a system 800 configured to implement one or more embodiments of the present disclosure. As explained below, during the contactless card creation process, two cryptographic keys may be assigned uniquely for each card. The cryptographic keys may comprise symmetric keys which may be used in both encryption and decryption of data. Triple DES (3DES) algorithm may be used by EMV and it is implemented by hardware in the contactless card. By using a key diversification process, one or more keys may be derived from a master key based upon uniquely identifiable information for each entity that requires a key.

**[0088]** Regarding master key management, two issuer master keys 802, 826 may be required for each part of the portfolio on which the one or more applets is issued. For example, the first master key 802 may comprise an Issuer Cryptogram Generation/Authentication Key (Iss-Key-Auth) and the second master key 826 may comprise an Issuer Data Encryption Key (Iss-Key-DEK). As further explained herein, two issuer master keys 802, 826 are diversified into card master keys 808, 820, which are unique for each card. In some examples, a network profile record ID (pNPR) 522 and derivation key index (pDKI) 824, as back office data, may be used to identify which Issuer Master Keys 802, 826 to use in the cryptographic processes for authentication. The system performing the authentication may be configured to retrieve values of pNPR 822 and pDKI 824 for a contactless card at the time of authentication.

**[0089]** In some examples, to increase the security of the solution, a session key may be derived (such as a unique key per session) but rather than using the master key, the unique card-derived keys and the counter may be used as diversification data, as explained above.

For example, each time the card is used in operation, a different key may be used for creating the message authentication code (MAC) and for performing the encryption. Regarding session key generation, the keys used to generate the cryptogram and encipher the data in the one or more applets may comprise session keys based on the card unique keys (Card-Key-Auth 808 and Card-Key-Dek 820). The session keys (Aut-Session-Key 832 and DEK-Session-Key 810) may be generated by the one or more applets and derived by using the application transaction counter (pATC) 804 with one or more algorithms. To fit data into the one or more algorithms, only the 2 low order bytes of the 4-byte pATC 804 is used. In some examples, the four byte session key derivation method may comprise:  $F1 := \text{PATC}(\text{lower 2 bytes}) \parallel 'F0' \parallel '00' \parallel \text{PATC (four bytes)}$   $F1 := \text{PATC}(\text{lower 2 bytes}) \parallel '0F' \parallel '00' \parallel \text{PATC (four bytes)}$   $SK := \{(\text{ALG (MK) [F1]}) \parallel \text{ALG (MK) [F2]}\}$ , where ALG may include 3DES ECB and MK may include the card unique derived master key.

**[0090]** As described herein, one or more MAC session keys may be derived using the lower two bytes of pATC 804 counter. At each tap of the contactless card, pATC 804 is configured to be updated, and the card master keys Card-Key-AUTH 704 and Card-Key-DEK 820 are further diversified into the session keys Aut-Session-Key 832 and DEK-Session-KEY 810. pATC 804 may be initialized to zero at personalization or applet initialization time. In some examples, the pATC counter 804 may be initialized at or before personalization, and may be configured to increment by one at each NDEF read.

**[0091]** Further, the update for each card may be unique, and assigned either by personalization, or algorithmically assigned by pUID or other identifying information. For example, odd numbered cards may increment or decrement by 2 and even numbered cards may increment or decrement by 5. In some examples, the update may also vary in sequential reads, such that one card may increment in sequence by 1, 3, 5, 2, 2, ... repeating. The specific sequence or algorithmic sequence may be defined at personalization time, or from one or more processes derived from unique identifiers. This can make it harder for a replay attacker to generalize from a small number of card instances.

**[0092]** The authentication message may be delivered as the content of a text NDEF record in hexadecimal ASCII format. In some examples, only the authentication data and an 8-byte random number followed by MAC of the authentication data may be included. In some examples, the random number may precede cryptogram A and may be one block long. In other examples, there may be no restriction on the length of the random number. In further examples, the total data (i.e., the random number plus the cryptogram) may be a multiple of the block size. In these examples, an additional 8-byte block may be added to match the block produced by the MAC algorithm. As another example, if the algorithms employed

used 16-byte blocks, even multiples of that block size may be used, or the output may be automatically, or manually, padded to a multiple of that block size.

**[0093]** The MAC may be performed by a function key (AUT-Session-Key) 832. The data specified in cryptogram may be processed with javacard.signature method:

ALG\_DES\_MAC8\_ISO9797\_1\_M2\_ALG3 to correlate to EMV ARQC verification methods. The key used for this computation may comprise a session key AUT-Session-Key 832, as explained above. As explained above, the low order two bytes of the counter may be used to diversify for the one or more MAC session keys. As explained below, AUT-Session-Key 832 may be used to MAC data 806, and the resulting data or cryptogram An 814 and random number RND may be encrypted using DEK-Session-Key 810 to create cryptogram B or output 818 sent in the message.

**[0094]** In some examples, one or more HSM commands may be processed for decrypting such that the final 16 (binary, 32 hex) bytes may comprise a 3DES symmetric encrypting using CBC mode with a zero IV of the random number followed by MAC authentication data. The key used for this encryption may comprise a session key DEK-Session-Key 810 derived from the Card-Key-DEK 820. In this case, the ATC value for the session key derivation is the least significant byte of the counter pATC 804.

**[0095]** The format below represents a binary version example embodiment. Further, in some examples, the first byte may be set to ASCII 'A'.

| Message Format          |         |      |               |                     |  |
|-------------------------|---------|------|---------------|---------------------|--|
| 1                       | 2       | 4    | 8             | 8                   |  |
| 0x43 (Message Type ‘A’) | Version | pATC | RND           | Cryptogram A (MAC)  |  |
|                         |         |      |               |                     |  |
| Cryptogram A (MAC)      | 8 bytes |      |               |                     |  |
| MAC of                  |         |      |               |                     |  |
| 2                       | 8       | 4    | 4             | 18 bytes input data |  |
| Version                 | pUID    | pATC | Shared Secret |                     |  |

| Message Format |   |   |    |
|----------------|---|---|----|
| 1              | 2 | 4 | 16 |

|                           |                |      |               |                     |  |
|---------------------------|----------------|------|---------------|---------------------|--|
| 0x43 (Message Type ‘A’)   | Version        | pATC |               | Cryptogram B        |  |
|                           |                |      |               |                     |  |
| <b>Cryptogram A (MAC)</b> | <b>8 bytes</b> |      |               |                     |  |
| MAC of                    |                |      |               |                     |  |
| 2                         | 8              | 4    | 4             | 18 bytes input data |  |
| Version                   | pUID           | pATC | Shared Secret |                     |  |
|                           |                |      |               |                     |  |
| <b>Cryptogram B</b>       | <b>16</b>      |      |               |                     |  |
| Sym Encryption of         |                |      |               |                     |  |
| 8                         | 8              |      |               |                     |  |
| RND                       | Cryptogram A   |      |               |                     |  |

[0096] Another exemplary format is shown below. In this example, the tag may be encoded in hexadecimal format.

|                       |      |      |               |                     |
|-----------------------|------|------|---------------|---------------------|
| <b>Message Format</b> |      |      |               |                     |
| 2                     | 8    | 4    | 8             | 8                   |
| Version               | pUID | pATC | RND           | Cryptogram A (MAC)  |
|                       |      |      |               |                     |
| <b>8 bytes</b>        |      |      |               |                     |
| 8                     | 8    | 4    | 4             | 18 bytes input data |
| pUID                  | pUID | pATC | Shared Secret |                     |

|                       |      |      |              |
|-----------------------|------|------|--------------|
| <b>Message Format</b> |      |      |              |
| 2                     | 8    | 4    | 16           |
| Version               | pUID | pATC | Cryptogram B |
|                       |      |      |              |

|                      |                 |      |               |
|----------------------|-----------------|------|---------------|
| 8 bytes              |                 |      |               |
| 8                    |                 | 4    | 4             |
| pUID                 | pUID            | pATC | Shared Secret |
|                      |                 |      |               |
| <b>Cryptogram B</b>  | <b>16</b>       |      |               |
| Sym Encryption<br>of |                 |      |               |
| 8                    | 8               |      |               |
| RND                  | Cryptogram<br>A |      |               |

[0097] The UID field of the received message may be extracted to derive, from master keys Iss-Key-AUTH 500 and Iss-Key-DEK 826, the card master keys (Card-Key-Auth 808 and Card-Key-DEK 820) for that particular card. Using the card master keys (Card-Key-Auth 704 and Card-Key-DEK 820), the counter (pATC) field of the received message may be used to derive the session keys (Aut-Session-Key 832 and DEK-Session-Key 810) for that particular card. Cryptogram B 818 may be decrypted using the DEK-Session-KEY, which yields cryptogram An 814 and RND, and RND may be discarded. The UID field may be used to look up the shared secret of the contactless card which, along with the Ver, UID, and pATC fields of the message, may be processed through the cryptographic MAC using the re-created Aut-Session-Key to create a MAC output, such as MAC'. If MAC' is the same as cryptogram An 814, then this indicates that the message decryption and MAC checking have all passed. Then the pATC may be read to determine if it is valid.

[0098] During an authentication session, one or more cryptograms may be generated by the one or more applications. For example, the one or more cryptograms may be generated as a 3DES MAC using ISO 9797-1 Algorithm 3 with Method 2 padding via one or more session keys, such as Aut-Session-Key 832. The input data 806 may take the following form: Version (2), pUID (8), pATC (4), Shared Secret (4). In some examples, the numbers in the brackets may comprise length in bytes. In some examples, the shared secret may be generated by one or more random number generators which may be configured to ensure, through one or more secure processes, that the random number is unpredictable. In some examples, the shared secret may comprise a random 4-byte binary number injected into the card at personalization time that is known by the authentication service. During an

authentication session, the shared secret may not be provided from the one or more applets to the mobile application. Method 2 padding may include adding a mandatory 0x'80' byte to the end of input data and 0x'00' bytes that may be added to the end of the resulting data up to the 8-byte boundary. The resulting cryptogram may comprise 8 bytes in length.

**[0099]** In some examples, one benefit of encrypting an unshared random number as the first block with the MAC cryptogram, is that it acts as an initialization vector while using CBC (Block chaining) mode of the symmetric encryption algorithm. This allows the “scrambling” from block to block without having to pre-establish either a fixed or dynamic IV.

**[0100]** By including the application transaction counter (pATC) as part of the data included in the MAC cryptogram, the authentication service may be configured to determine if the value conveyed in the clear data has been tampered with. Moreover, by including the version in the one or more cryptograms, it is difficult for an attacker to purposefully misrepresent the application version in an attempt to downgrade the strength of the cryptographic solution. In some examples, the pATC may start at zero and be updated by 1 each time the one or more applications generates authentication data. The authentication service may be configured to track the pATCs used during authentication sessions. In some examples, when the authentication data uses a pATC equal to or lower than the previous value received by the authentication service, this may be interpreted as an attempt to replay an old message, and the authenticated may be rejected. In some examples, where the pATC is greater than the previous value received, this may be evaluated to determine if it is within an acceptable range or threshold, and if it exceeds or is outside the range or threshold, verification may be deemed to have failed or be unreliable. In the MAC operation 812, data 806 is processed through the MAC using Aut-Session-Key 832 to produce MAC output (cryptogram A) 814, which is encrypted.

**[0101]** In order to provide additional protection against brute force attacks exposing the keys on the card, it is desirable that the MAC cryptogram 814 be enciphered. In some examples, data or cryptogram An 814 to be included in the ciphertext may comprise: Random number (8), cryptogram (8). In some examples, the numbers in the brackets may comprise length in bytes. In some examples, the random number may be generated by one or more random number generators which may be configured to ensure, through one or more secure processes, that the random number is unpredictable. The key used to encipher this data may comprise a session key. For example, the session key may comprise DEK-Session-Key 810. In the encryption operation 816, data or cryptogram An 814 and RND are processed using DEK-Session-Key 706 to produce encrypted data, cryptogram B 818. The

data 814 may be enciphered using 3DES in cipher block chaining mode to ensure that an attacker must run any attacks over all of the ciphertext. As a non-limiting example, other algorithms, such as Advanced Encryption Standard (AES), may be used. In some examples, an initialization vector of 0x'0000000000000000' may be used. Any attacker seeking to brute force the key used for enciphering this data will be unable to determine when the correct key has been used, as correctly decrypted data will be indistinguishable from incorrectly decrypted data due to its random appearance.

**[0102]** In order for the authentication service to validate the one or more cryptograms provided by the one or more applets, the following data must be conveyed from the one or more applets to the mobile device in the clear during an authentication session: version number to determine the cryptographic approach used and message format for validation of the cryptogram, which enables the approach to change in the future; pUID to retrieve cryptographic assets, and derive the card keys; and pATC to derive the session key used for the cryptogram.

**[0103]** FIG. 9 illustrates a method 900 for generating a cryptogram. For example, at block 902, a network profile record ID (pNPR) and derivation key index (pDKI) may be used to identify which Issuer Master Keys to use in the cryptographic processes for authentication. In some examples, the method may include performing the authentication to retrieve values of pNPR and pDKI for a contactless card at the time of authentication.

**[0104]** At block 904, Issuer Master Keys may be diversified by combining them with the card's unique ID number (pUID) and the PAN sequence number (PSN) of one or more applets, for example, a payment applet.

**[0105]** At block 906, Card-Key-Auth and Card-Key-DEK (unique card keys) may be created by diversifying the Issuer Master Keys to generate session keys which may be used to generate a MAC cryptogram.

**[0106]** At block 908, the keys used to generate the cryptogram and encipher the data in the one or more applets may comprise the session keys of block 1230 based on the card unique keys (Card-Key-Auth and Card-Key-DEK). In some examples, these session keys may be generated by the one or more applets and derived by using pATC, resulting in session keys Aut-Session-Key and DEK-Session-Key.

**[0107]** FIG. 10 depicts an exemplary process 1000 illustrating key diversification according to one example. Initially, a sender and the recipient may be provisioned with two different master keys. For example, a first master key may comprise the data encryption master key, and a second master key may comprise the data integrity master key. The sender

has a counter value, which may be updated at block 1002, and other data, such as data to be protected, which it may secure share with the recipient.

**[0108]** At block 1004, the counter value may be encrypted by the sender using the data encryption master key to produce the data encryption derived session key, and the counter value may also be encrypted by the sender using the data integrity master key to produce the data integrity derived session key. In some examples, a whole counter value or a portion of the counter value may be used during both encryptions.

**[0109]** In some examples, the counter value may not be encrypted. In these examples, the counter may be transmitted between the sender and the recipient in the clear, i.e., without encryption.

**[0110]** At block 1006, the data to be protected is processed with a cryptographic MAC operation by the sender using the data integrity session key and a cryptographic MAC algorithm. The protected data, including plaintext and shared secret, may be used to produce a MAC using one of the session keys (AUT-Session-Key).

**[0111]** At block 1008, the data to be protected may be encrypted by the sender using the data encryption derived session key in conjunction with a symmetric encryption algorithm. In some examples, the MAC is combined with an equal amount of random data, for example each 8 bytes long, and then encrypted using the second session key (DEK-Session-Key).

**[0112]** At block 1010, the encrypted MAC is transmitted, from the sender to the recipient, with sufficient information to identify additional secret information (such as shared secret, master keys, etc.), for verification of the cryptogram.

**[0113]** At block 1012, the recipient uses the received counter value to independently derive the two derived session keys from the two master keys as explained above.

**[0114]** At block 1014, the data encryption derived session key is used in conjunction with the symmetric decryption operation to decrypt the protected data. Additional processing on the exchanged data will then occur. In some examples, after the MAC is extracted, it is desirable to reproduce and match the MAC. For example, when verifying the cryptogram, it may be decrypted using appropriately generated session keys. The protected data may be reconstructed for verification. A MAC operation may be performed using an appropriately generated session key to determine if it matches the decrypted MAC. As the MAC operation is an irreversible process, the only way to verify is to attempt to recreate it from source data.

**[0115]** At block 1016, the data integrity derived session key is used in conjunction with the cryptographic MAC operation to verify that the protected data has not been modified.

**[0116]** Some examples of the methods described herein may advantageously confirm when a successful authentication is determined when the following conditions are met. First, the ability to verify the MAC shows that the derived session key was proper. The MAC may only be correct if the decryption was successful and yielded the proper MAC value. The successful decryption may show that the correctly derived encryption key was used to decrypt the encrypted MAC. Since the derived session keys are created using the master keys known only to the sender (e.g., the transmitting device) and recipient (e.g., the receiving device), it may be trusted that the contactless card which originally created the MAC and encrypted the MAC is indeed authentic. Moreover, the counter value used to derive the first and second session keys may be shown to be valid and may be used to perform authentication operations.

**[0117]** Thereafter, the two derived session keys may be discarded, and the next iteration of data exchange will update the counter value (returning to block 1002) and a new set of session keys may be created (at block 1010). In some examples, the combined random data may be discarded.

**[0118]** FIG. 11 illustrates a method 1000 for card activation according to an example embodiment. For example, card activation may be completed by a system including a card, a device, and one or more servers. The contactless card, device, and one or more servers may reference same or similar components that were previously explained a, such as contactless card 500, mobile device, and a server.

**[0119]** In block , the card may be configured to dynamically generate data. In some examples, this data may include information such as an account number, card identifier, card verification value, or phone number, which may be transmitted from the card to the device. In some examples, one or more portions of the data may be encrypted via the systems and methods disclosed herein.

**[0120]** In block 1104, one or more portions of the dynamically generated data may be communicated to an application of the device via NFC or other wireless communication. For example, a tap of the card proximate to the device may allow the application of the device to read the one or more portions of the data associated with the contactless card. In some examples, if the device does not comprise an application to assist in activation of the card, the tap of the card may direct the device or prompt the customer to a software application store to download an associated application to activate the card. In some examples, the user may be prompted to sufficiently gesture, place, or orient the card towards a surface of the device, such as either at an angle or flatly placed on, near, or proximate the

surface of the device. Responsive to a sufficient gesture, placement and/or orientation of the card, the device may proceed to transmit the one or more encrypted portions of data received from the card to the one or more servers.

**[0121]** In block 1106, the one or more portions of the data may be communicated to one or more servers, such as a card issuer server. For example, one or more encrypted portions of the data may be transmitted from the device to the card issuer server for activation of the card.

**[0122]** In block 1108, the one or more servers may decrypt the one or more encrypted portions of the data via the systems and methods disclosed herein. For example, the one or more servers may receive the encrypted data from the device and may decrypt it in order to compare the received data to record data accessible to the one or more servers. If a resulting comparison of the one or more decrypted portions of the data by the one or more servers yields a successful match, the card may be activated. If the resulting comparison of the one or more decrypted portions of the data by the one or more servers yields an unsuccessful match, one or more processes may take place. For example, responsive to the determination of the unsuccessful match, the user may be prompted to tap, swipe, or wave gesture the card again. In this case, there may be a predetermined threshold comprising a number of attempts that the user is permitted to activate the card. Alternatively, the user may receive a notification, such as a message on his or her device indicative of the unsuccessful attempt of card verification and to call, email or text an associated service for assistance to activate the card, or another notification, such as a phone call on his or her device indicative of the unsuccessful attempt of card verification and to call, email or text an associated service for assistance to activate the card, or another notification, such as an email indicative of the unsuccessful attempt of card verification and to call, email or text an associated service for assistance to activate the card.

**[0123]** In block 1110, the one or more servers may transmit a return message based on the successful activation of the card. For example, the device may be configured to receive output from the one or more servers indicative of a successful activation of the card by the one or more servers. The device may be configured to display a message indicating successful activation of the card. Once the card has been activated, the card may be configured to discontinue dynamically generating data so as to avoid fraudulent use. In this manner, the card may not be activated thereafter, and the one or more servers are notified that the card has already been activated.

**[0124]** FIG. 12 illustrates an embodiment of an exemplary computer architecture 1200 suitable for implementing various embodiments as previously described. In one embodiment, the computer architecture 1200 may include or be implemented as part of one or more systems or devices discussed herein.

**[0125]** As used in this application, the terms “system” and “component” are intended to refer to a computer-related entity, either hardware, a combination of hardware and software, software, or software in execution, examples of which are provided by the exemplary computing computer architecture 1200. For example, a component can be, but is not limited to being, a process running on a processor, a processor, a hard disk drive, multiple storage drives (of optical and/or magnetic storage medium), an object, an executable, a thread of execution, a program, and/or a computer. By way of illustration, both an application running on a server and the server can be a component. One or more components can reside within a process and/or thread of execution, and a component can be localized on one computer and/or distributed between two or more computers. Further, components may be communicatively coupled to each other by various types of communications media to coordinate operations. The coordination may involve the uni-directional or bi-directional exchange of information. For instance, the components may communicate information in the form of signals communicated over the communications media. The information can be implemented as signals allocated to various signal lines. In such allocations, each message is a signal. Further embodiments, however, may alternatively employ data messages. Such data messages may be sent across various connections. Exemplary connections include parallel interfaces, serial interfaces, and bus interfaces.

**[0126]** The computing computer architecture 1200 includes various common computing elements, such as one or more processors, multi-core processors, co-processors, memory units, chipsets, controllers, peripherals, interfaces, oscillators, timing devices, video cards, audio cards, multimedia input/output (I/O) components, power supplies, and so forth. The embodiments, however, are not limited to implementation by the computing computer architecture 1200.

**[0127]** As shown in FIG. 12, the computing computer architecture 1200 includes a processor 1212, a system memory 1204 and a system bus 1206. The processor 1212 can be any of various commercially available processors.

**[0128]** The system bus 1206 provides an interface for system components including, but not limited to, the system memory 1204 to the processor 1212. The system bus 1206 can be any of several types of bus structure that may further interconnect to a memory bus (with or

without a memory controller), a peripheral bus, and a local bus using any of a variety of commercially available bus architectures. Interface adapters may connect to the system bus 1206 via slot architecture. Example slot architectures may include without limitation Accelerated Graphics Port (AGP), Card Bus, (Extended) Industry Standard Architecture ((E)ISA), Micro Channel Architecture (MCA), NuBus, Peripheral Component Interconnect (Extended) (PCI(X)), PCI Express, Personal Computer Memory Card International Association (PCMCIA), and the like.

**[0129]** The computer architecture 1200 may include or implement various articles of manufacture. An article of manufacture may include a computer-readable storage medium to store logic. Examples of a computer-readable storage medium may include any tangible media capable of storing electronic data, including volatile memory or non-volatile memory, removable or non-removable memory, erasable or non-erasable memory, writeable or re-writable memory, and so forth. Examples of logic may include executable computer program instructions implemented using any suitable type of code, such as source code, compiled code, interpreted code, executable code, static code, dynamic code, object-oriented code, visual code, and the like. Embodiments may also be at least partly implemented as instructions contained in or on a non-transitory computer-readable medium, which may be read and executed by one or more processors to enable performance of the operations described herein.

**[0130]** The system memory 1204 may include various types of computer-readable storage media in the form of one or more higher speed memory units, such as read-only memory (ROM), random-access memory (RAM), dynamic RAM (DRAM), Double-Data-Rate DRAM (DDRAM), synchronous DRAM (SDRAM), static RAM (SRAM), programmable ROM (PROM), erasable programmable ROM (EPROM), electrically erasable programmable ROM (EEPROM), flash memory, polymer memory such as ferroelectric polymer memory, ovonic memory, phase change or ferroelectric memory, silicon-oxide-nitride-oxide-silicon (SONOS) memory, magnetic or optical cards, an array of devices such as Redundant Array of Independent Disks (RAID) drives, solid state memory devices (e.g., USB memory, solid state drives (SSD) and any other type of storage media suitable for storing information. In the illustrated embodiment shown in FIG. 12, the system memory 1204 can include non-volatile 1208 and/or volatile 1210. A basic input/output system (BIOS) can be stored in the non-volatile 1208.

**[0131]** The computer 1202 may include various types of computer-readable storage media in the form of one or more lower speed memory units, including an internal (or external) hard disk drive 1230, a magnetic disk drive 1216 to read from or write to a removable

magnetic disk 1220, and an optical disk drive 1228 to read from or write to a removable optical disk 1232 (e.g., a CD-ROM or DVD). The hard disk drive 1230, magnetic disk drive 1216 and optical disk drive 1228 can be connected to system bus 1206 the by an HDD interface 1214, and FDD interface 1218 and an optical disk drive interface 1234, respectively. The HDD interface 1214 for external drive implementations can include at least one or both of Universal Serial Bus (USB) and IEEE 1394 interface technologies.

**[0132]** The drives and associated computer-readable media provide volatile and/or nonvolatile storage of data, data structures, computer-executable instructions, and so forth. For example, a number of program modules can be stored in the drives and non-volatile 1208, and volatile 1210, including an operating system 1222, one or more applications 1242, other program modules 1224, and program data 1226. In one embodiment, the one or more applications 1242, other program modules 1224, and program data 1226 can include, for example, the various applications and/or components of the systems discussed herein.

**[0133]** A user can enter commands and information into the computer 1202 through one or more wire/wireless input devices, for example, a keyboard 1250 and a pointing device, such as a mouse 1252. Other input devices may include microphones, infra-red (IR) remote controls, radio-frequency (RF) remote controls, game pads, stylus pens, card readers, dongles, finger print readers, gloves, graphics tablets, joysticks, keyboards, retina readers, touch screens (e.g., capacitive, resistive, etc.), trackballs, track pads, sensors, styluses, and the like. These and other input devices are often connected to the processor 1212 through an input device interface 1236 that is coupled to the system bus 1206 but can be connected by other interfaces such as a parallel port, IEEE 1394 serial port, a game port, a USB port, an IR interface, and so forth.

**[0134]** A monitor 1244 or other type of display device is also connected to the system bus 1206 via an interface, such as a video adapter 1246. The monitor 1244 may be internal or external to the computer 1202. In addition to the monitor 1244, a computer typically includes other peripheral output devices, such as speakers, printers, and so forth.

**[0135]** The computer 1202 may operate in a networked environment using logical connections via wire and/or wireless communications to one or more remote computers, such as a remote computer(s) 1248. The remote computer(s) 1248 can be a workstation, a server computer, a router, a personal computer, portable computer, microprocessor-based entertainment appliance, a peer device or other common network node, and typically includes many or all the elements described relative to the computer 1202, although, for purposes of brevity, only a memory and/or storage device 1258 is illustrated. The logical

connections depicted include wire/wireless connectivity to a local area network 1256 and/or larger networks, for example, a wide area network 1254. Such LAN and WAN networking environments are commonplace in offices and companies, and facilitate enterprise-wide computer networks, such as intranets, all of which may connect to a global communications network, for example, the Internet.

**[0136]** When used in a local area network 1256 networking environment, the computer 1202 is connected to the local area network 1256 through a wire and/or wireless communication network interface or network adapter 1238. The network adapter 1238 can facilitate wire and/or wireless communications to the local area network 1256, which may also include a wireless access point disposed thereon for communicating with the wireless functionality of the network adapter 1238.

**[0137]** When used in a wide area network 1254 networking environment, the computer 1202 can include a modem 1240, or is connected to a communications server on the wide area network 1254 or has other means for establishing communications over the wide area network 1254, such as by way of the Internet. The modem 1240, which can be internal or external and a wire and/or wireless device, connects to the system bus 1206 via the input device interface 1236. In a networked environment, program modules depicted relative to the computer 1202, or portions thereof, can be stored in the remote memory and/or storage device 1258. It will be appreciated that the network connections shown are exemplary and other means of establishing a communications link between the computers can be used.

**[0138]** The computer 1202 is operable to communicate with wire and wireless devices or entities using the IEEE 802 family of standards, such as wireless devices operatively disposed in wireless communication (e.g., IEEE 802.11 over-the-air modulation techniques). This includes at least Wi-Fi (or Wireless Fidelity), WiMax, and Bluetooth™ wireless technologies, among others. Thus, the communication can be a predefined structure as with a conventional network or simply an ad hoc communication between at least two devices. Wi-Fi networks use radio technologies called IEEE 802.11 (a, b, g, n, etc.) to provide secure, reliable, fast wireless connectivity. A Wi-Fi network can be used to connect computers to each other, to the Internet, and to wire networks (which use IEEE 802.3-related media and functions).

**[0139]** The various elements of the devices as previously described herein may include various hardware elements, software elements, or a combination of both. Examples of hardware elements may include devices, logic devices, components, processors, microprocessors, circuits, processors, circuit elements (e.g., transistors, resistors, capacitors,

inductors, and so forth), integrated circuits, application specific integrated circuits (ASIC), programmable logic devices (PLD), digital signal processors (DSP), field programmable gate array (FPGA), memory units, logic gates, registers, semiconductor device, chips, microchips, chip sets, and so forth. Examples of software elements may include software components, programs, applications, computer programs, application programs, system programs, software development programs, machine programs, operating system software, middleware, firmware, software modules, routines, subroutines, functions, methods, procedures, software interfaces, application program interfaces (API), instruction sets, computing code, computer code, code segments, computer code segments, words, values, symbols, or any combination thereof. However, determining whether an embodiment is implemented using hardware elements and/or software elements may vary in accordance with any number of factors, such as desired computational rate, power levels, heat tolerances, processing cycle budget, input data rates, output data rates, memory resources, data bus speeds and other design or performance constraints, as desired for a given implementation.

**[0140]** The components and features of the devices described above may be implemented using any combination of discrete circuitry, application specific integrated circuits (ASICs), logic gates and/or single chip architectures. Further, the features of the devices may be implemented using microcontrollers, programmable logic arrays and/or microprocessors or any combination of the foregoing where suitably appropriate. It is noted that hardware, firmware and/or software elements may be collectively or individually referred to herein as “logic” or “circuit.”

**[0141]** FIG. 13 is a block diagram depicting an exemplary communications architecture 1300 suitable for implementing various embodiments as previously described. The communications architecture 1300 includes various common communications elements, such as a transmitter, receiver, transceiver, radio, network interface, baseband processor, antenna, amplifiers, filters, power supplies, and so forth. The embodiments, however, are not limited to implementation by the communications architecture 1300, which may be consistent with systems and devices discussed herein.

**[0142]** As shown in FIG. 13, the communications architecture 1300 includes one or more client(s) 1302 and server(s) 1304. The server(s) 1304 may implement one or more functions and embodiments discussed herein. The client(s) 1302 and the server(s) 1304 are operatively connected to one or more respective client data store 1306 and server data store 1308 that can be employed to store information local to the respective client(s) 1302 and server(s) 1304, such as cookies and/or associated contextual information.

**[0143]** The client(s) 1302 and the server(s) 1304 may communicate information between each other using a communication framework 1310. The communication framework 1310 may implement any well-known communications techniques and protocols. The communication framework 1310 may be implemented as a packet-switched network (e.g., public networks such as the Internet, private networks such as an enterprise intranet, and so forth), a circuit-switched network (e.g., the public switched telephone network), or a combination of a packet-switched network and a circuit-switched network (with suitable gateways and translators).

**[0144]** The communication framework 1310 may implement various network interfaces arranged to accept, communicate, and connect to a communications network. A network interface may be regarded as a specialized form of an input/output (I/O) interface. Network interfaces may employ connection protocols including without limitation direct connect, Ethernet (e.g., thick, thin, twisted pair 10/100/1200 Base T, and the like), token ring, wireless network interfaces, cellular network interfaces, IEEE 802.11a-x network interfaces, IEEE 802.16 network interfaces, IEEE 802.20 network interfaces, and the like. Further, multiple network interfaces may be used to engage with various communications network types. For example, multiple network interfaces may be employed to allow for the communication over broadcast, multicast, and unicast networks. Should processing requirements dictate a greater amount speed and capacity, distributed network controller architectures may similarly be employed to pool, load balance, and otherwise increase the communicative bandwidth required by client(s) 1302 and the server(s) 1304. A communications network may be any one and the combination of wired and/or wireless networks including without limitation a direct interconnection, a secured custom connection, a private network (e.g., an enterprise intranet), a public network (e.g., the Internet), a Personal Area Network (PAN), a Local Area Network (LAN), a Metropolitan Area Network (MAN), an Operating Missions as Nodes on the Internet (OMNI), a Wide Area Network (WAN), a wireless network, a cellular network, and other communications networks.

## CLAIMS

1. A mobile device, comprising:

a display device;

a processor; and

memory comprising instructions that when executed by the processor, cause the processor to:

receive an input to perform an electronic exchange of money with a second mobile device of a first user;

cause the display device to present an indication to provide a contactless card on the mobile device;

receive, from the contactless card, first data, the first data encrypted with a private key of a second user;

generate a message comprising a uniform resource locator (URL) comprising the first data to communicate to the second mobile device, the URL configured to cause the second mobile device to perform authentication of the second user with the first data;

communicate the message to the second mobile device; and

cause the electronic exchange of money based on the authentication of the second user.

2. The mobile device of claim 1, comprising an input device coupled with the memory and the processor, the input device configured to receive the input comprising a selection of a contact, an indicated amount of money to electronically exchange, or a combination thereof.

3. The mobile device of claim 2, wherein the URL further comprises a first identifier to identify the amount of money, and a second identifier to identify the first user.

4. The mobile device of claim 1, the processor to:

detect, via a near-field communication (NFC) signal, the contactless card;

initiate an NFC exchange with the contactless card based on the contactless card coming within a communication range of the mobile device; and

receive the first data from the contactless card via the NFC exchange.

5. The mobile device of claim 1, wherein the URL comprises a link, that when activated, is configured to launch a webpage on the second mobile device to perform the authentication of the second user with the first data.

6. The mobile device of claim 1, wherein the first data comprises a string encrypted with the private key stored on the contactless card.
7. The mobile device of claim 1, wherein the private key is part of a key pair, and the key pair further comprising a public key associated with the contactless card, the first user, or a combination thereof.
8. The mobile device of claim 1, the processor to receive a result indicating that the second user is successfully authenticated from the second mobile device, and to cause the electronic exchange based on the result.
9. The mobile device of claim 1, the processor to communicate the message to the second mobile device as a Short Message Service (SMS) message, a Multimedia Message Service (MMS) message, or a Rich Communication Service (RCS) message.
10. A mobile device, comprising:
  - an input device;
  - a processor; and
  - memory comprising instructions that when executed by the processor, cause the processor to:
    - receive a message comprising a uniform resource locator (URL) comprising first data from a second mobile device, the URL configured to cause the mobile device to perform authentication of a first user with the first data;
    - detect, via the input device, an input to activate the URL;
    - launch an application in response to the input, wherein the application is configured to communicate with a server to perform the authentication of the first user with the first data;
    - receive an indication that the first user is authenticated by the server; and
    - cause performance of an exchange of money based on the first data and the indication that the first user is authenticated.
11. The mobile device of claim 10, wherein the input device comprises a touchscreen interface, the touchscreen interface to detect a selection of the URL on a display device.
12. The mobile device of claim 11, wherein the URL comprises a link, that when selected, launches the application.

13. The mobile device of claim 10, comprising a display device configured to present the message in a graphical user interface (GUI) display.

14. The mobile device of claim 10, the processor to communicate the first data to the server to perform the authentication of the first user.

15. The mobile device of claim 14, wherein the URL further comprises a first identifier of an amount of money to exchange, and a second identifier to identify the first user, and the processor to communicate at least one of the first identifier and the second identifier to the server.

16. The mobile device of claim 15, wherein the processor to cause the performance of the exchange based the amount of money and the first user being authenticated.

17. The mobile device of claim 10, wherein first data comprises an encrypted string associated with the first user, and the indication comprising information to indicate that the encrypted string was successful decrypted with a private key associated with the first user.

18. The mobile device of claim 17, wherein encrypted string is encrypted with a private key stored on a contactless card associated with the first user.

19. The mobile device of claim 10, the processor to receive the message as a Short Message Service (SMS) message, a Multimedia Message Service (MMS) message, or a Rich Communication Service (RCS) message.

20. A computer-implemented method, comprising:

- detecting, by a processor of a first mobile device associated with a first user, a request to exchange money with a second user of a second mobile device;

- causing, on a display device, a prompt to request the first user associated with the first user device to tap a contactless card on the first user device;

- receiving, from the contactless card, first data, the first data encrypted with a private key associated with the first user;

- generating a text message comprising a uniform resource locator (URL) further comprising the first data to communicate to second user device, the URL configured to cause the second mobile device to perform authentication of the first user with the first data;

- communicating the message to the second mobile device;

- receiving an indication from the second mobile device indicating the first user is authenticated; and

exchanging information with the second mobile device to exchange the money.

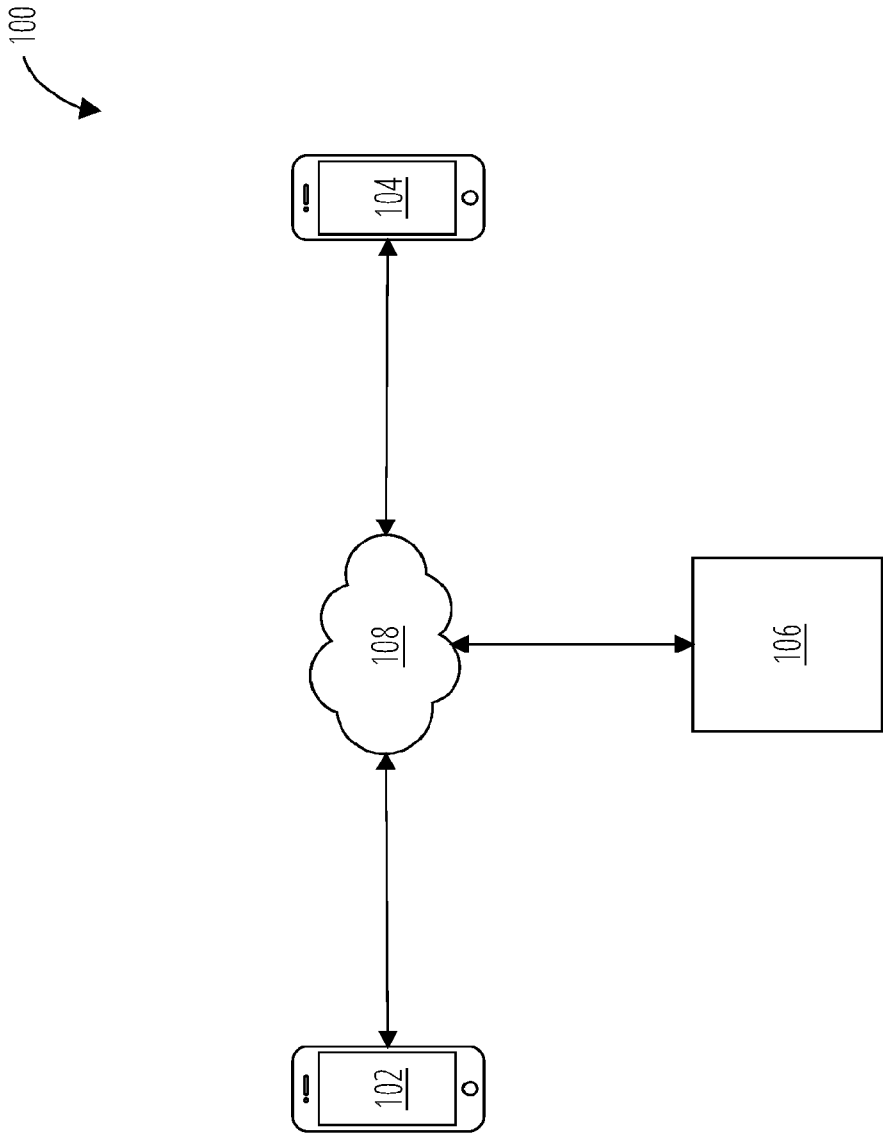


FIG. 1

2/13

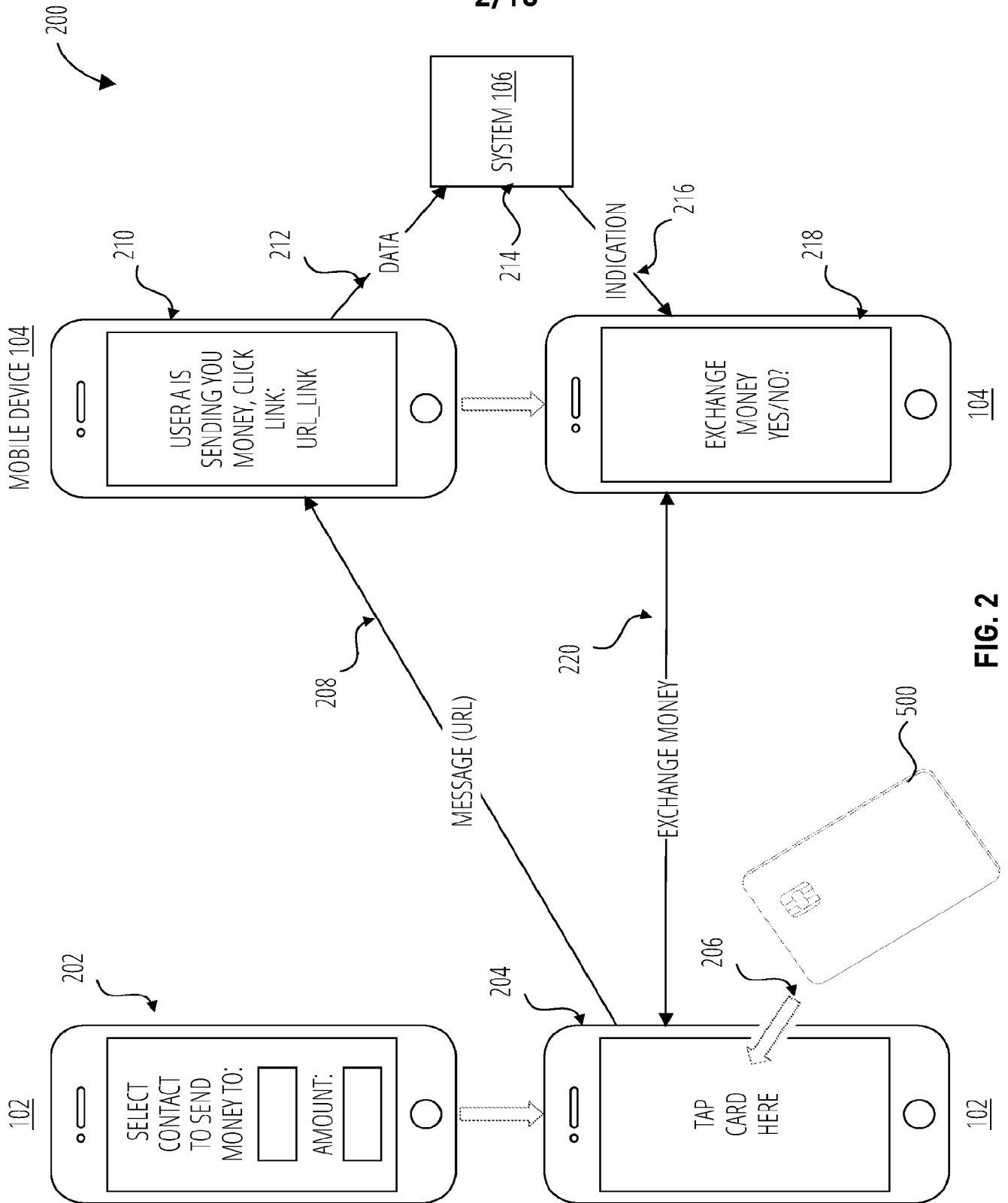


FIG. 2

3/13

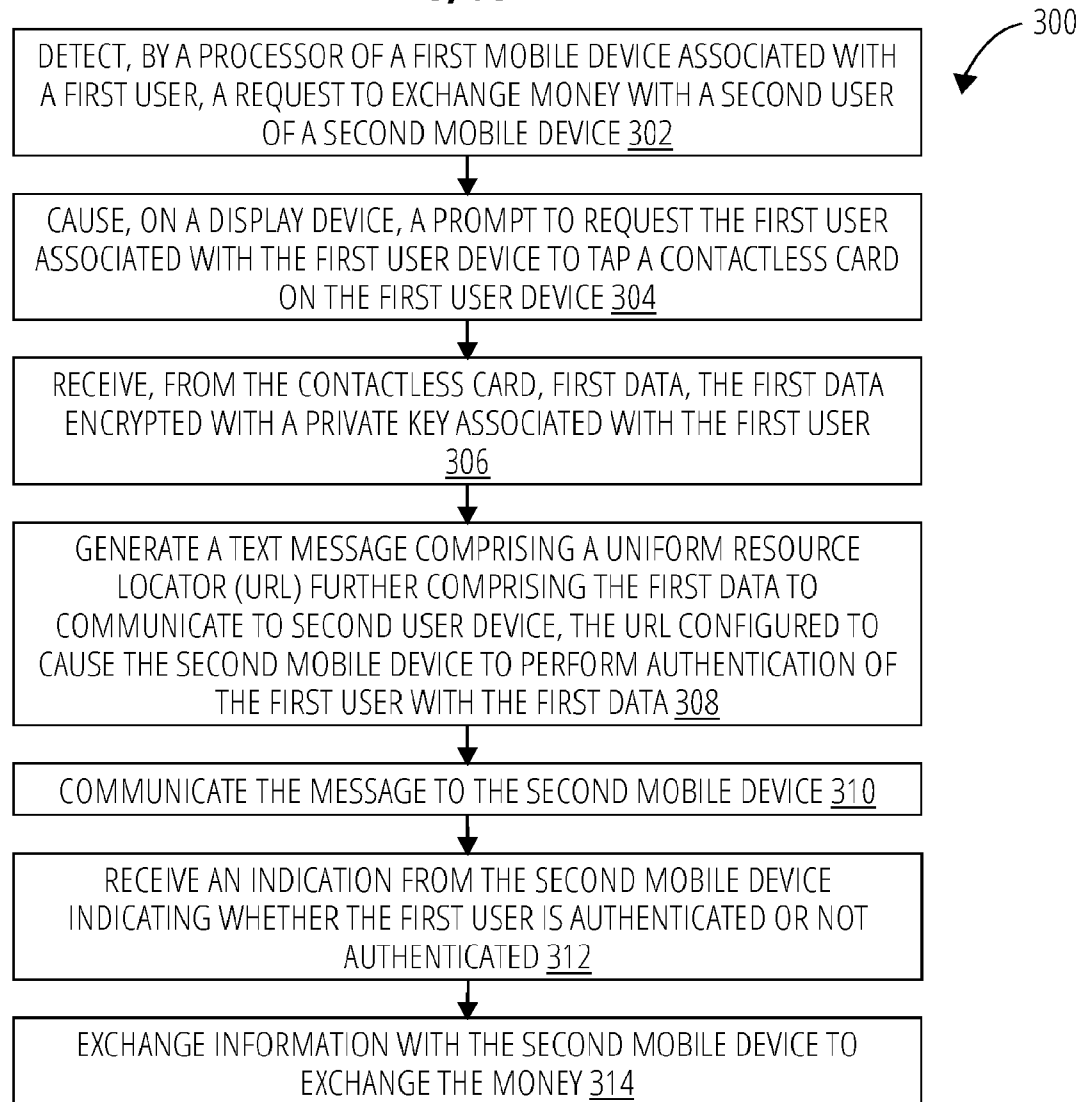


FIG. 3

4/13

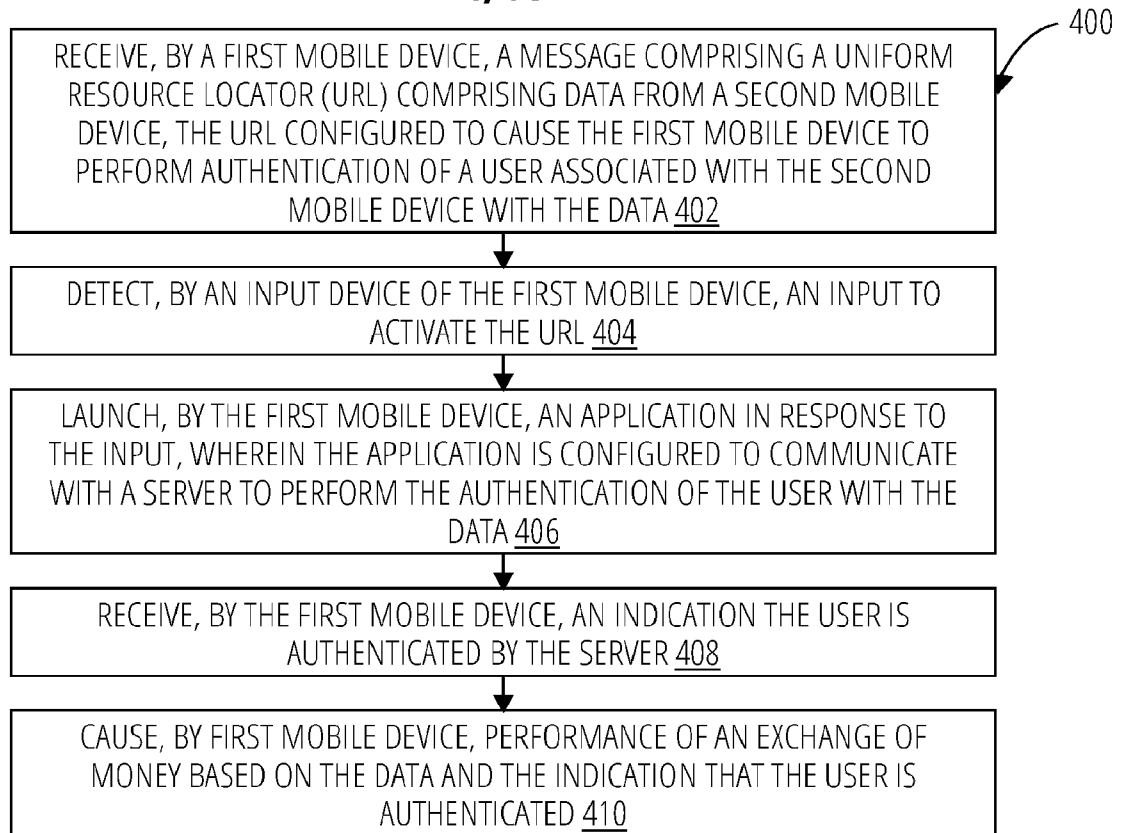


FIG. 4

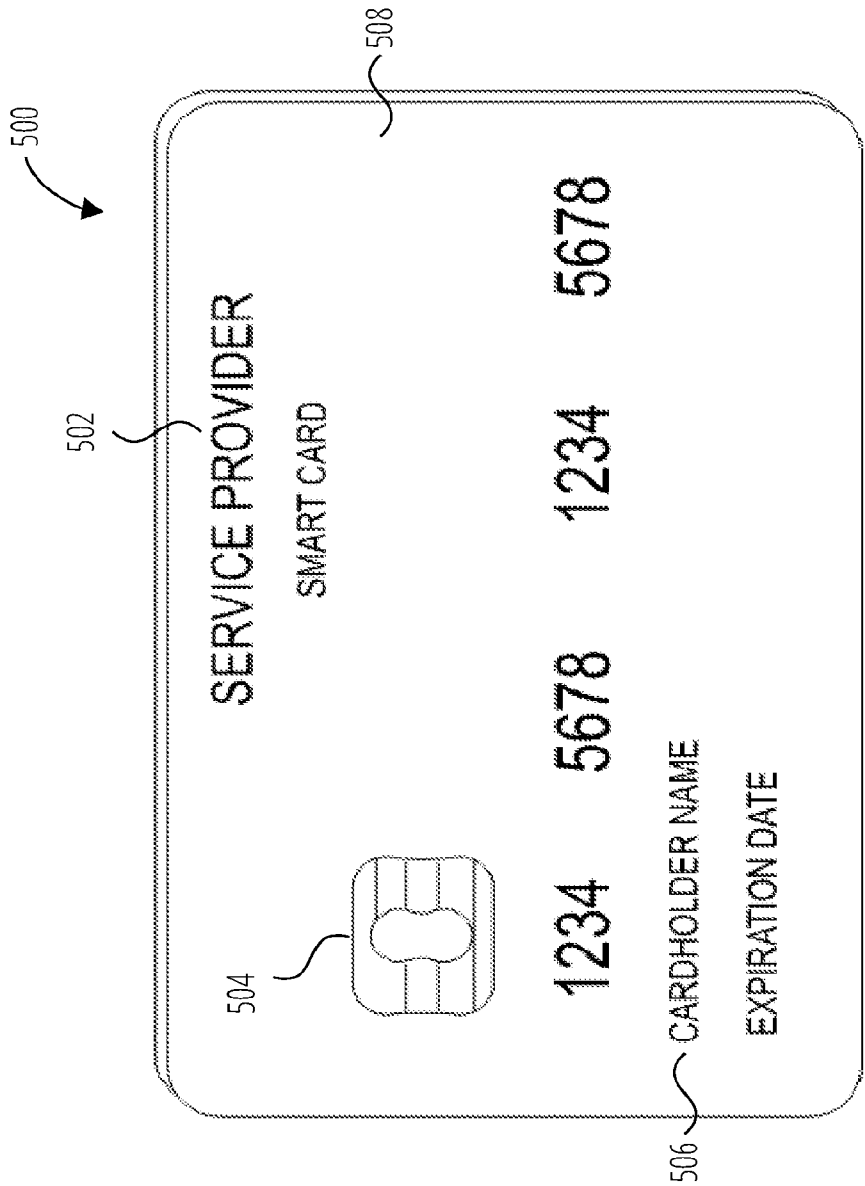


FIG. 5

6/13

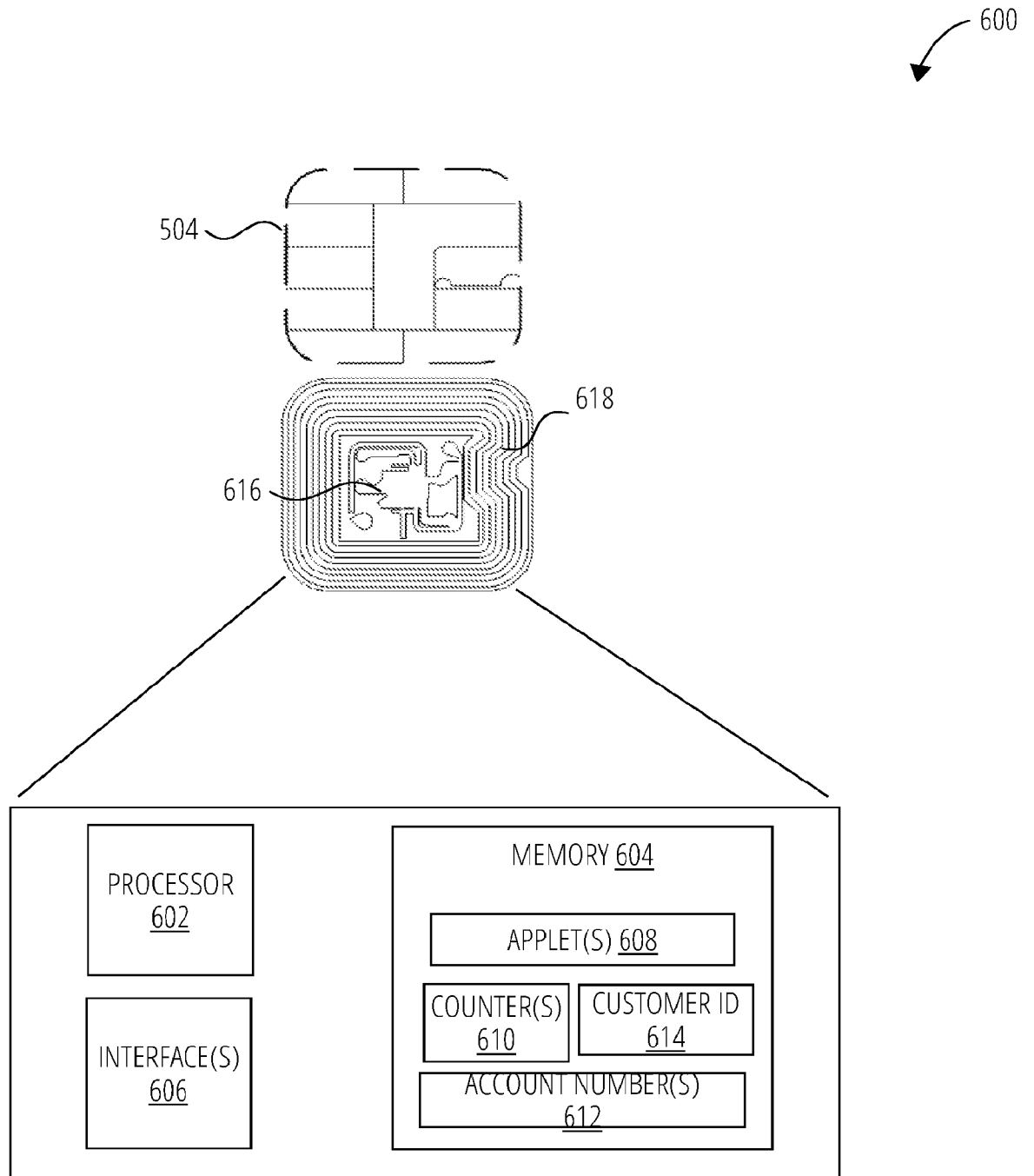


FIG. 6

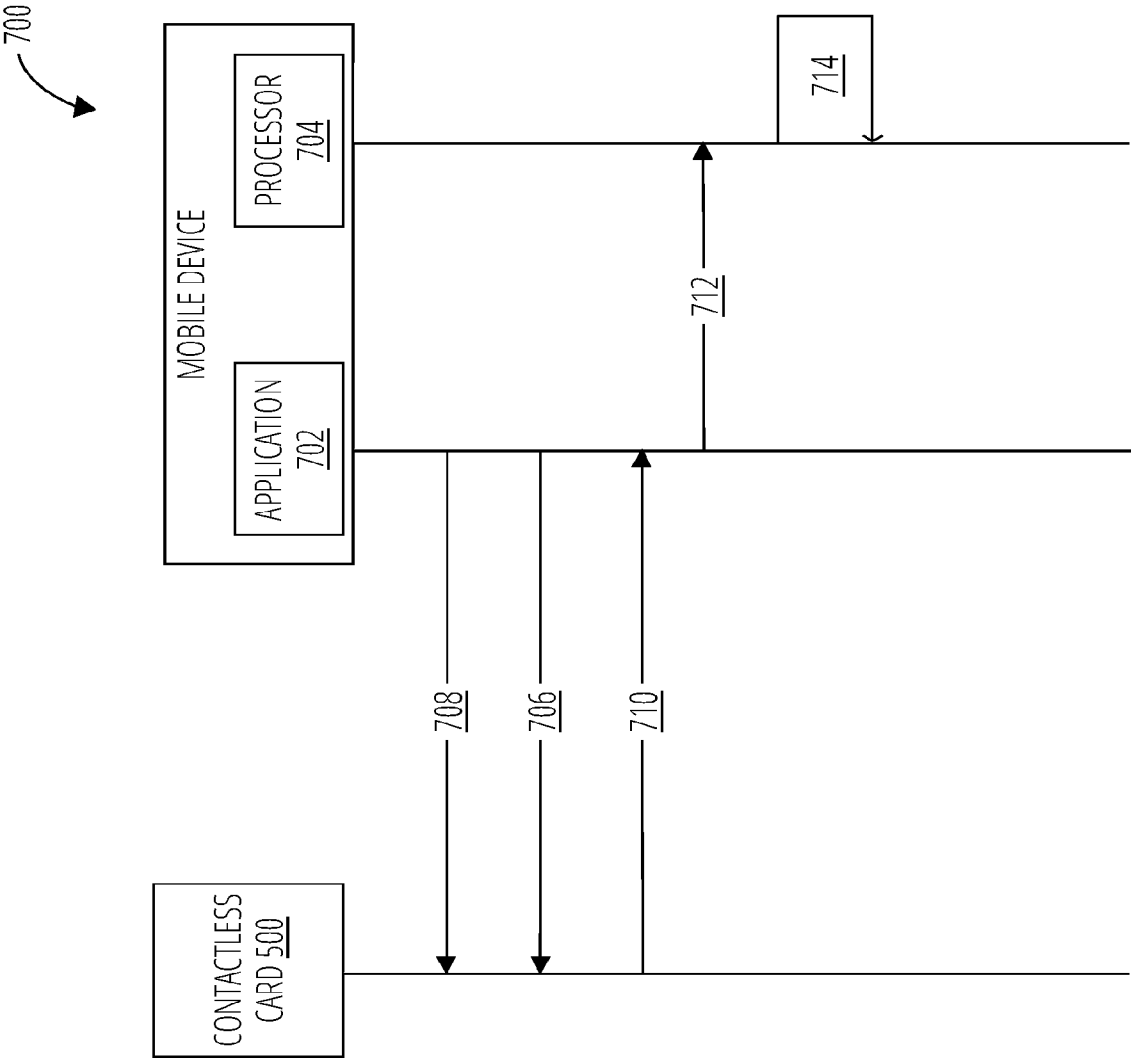


FIG. 7

8/13

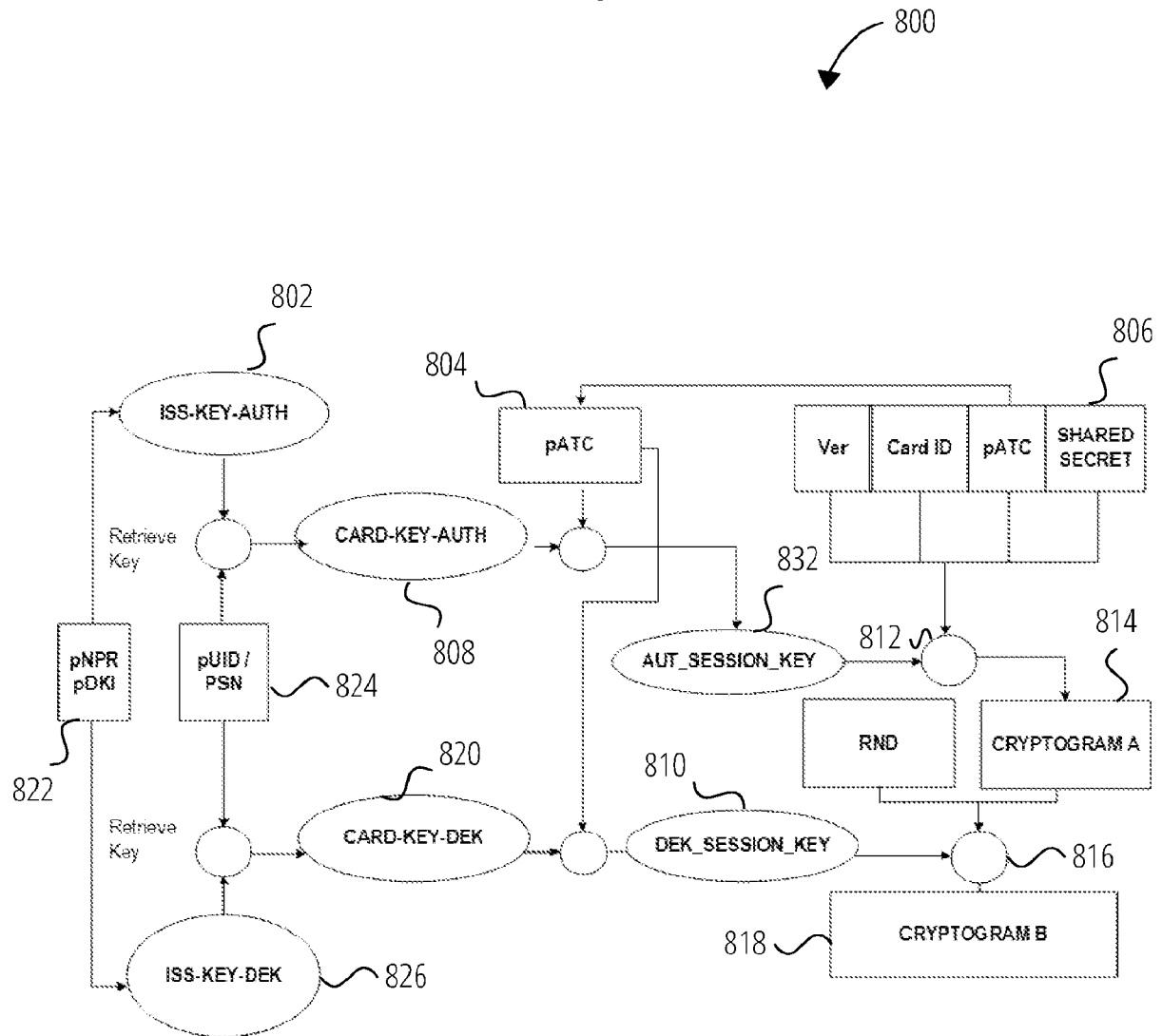


FIG. 8

9/13

900

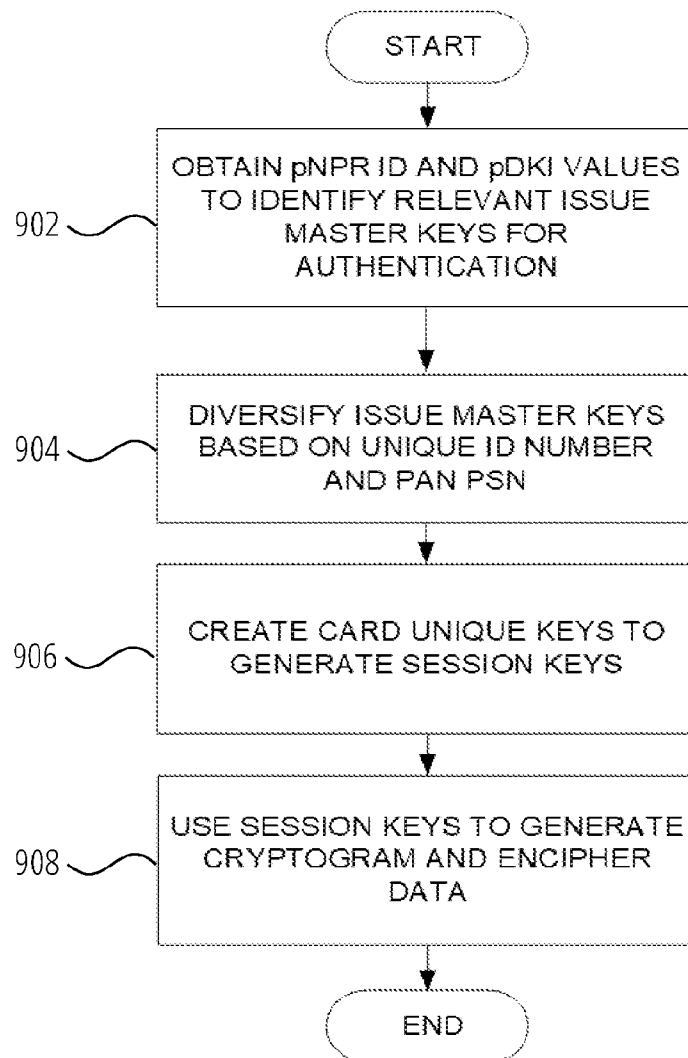


FIG. 9

10/13

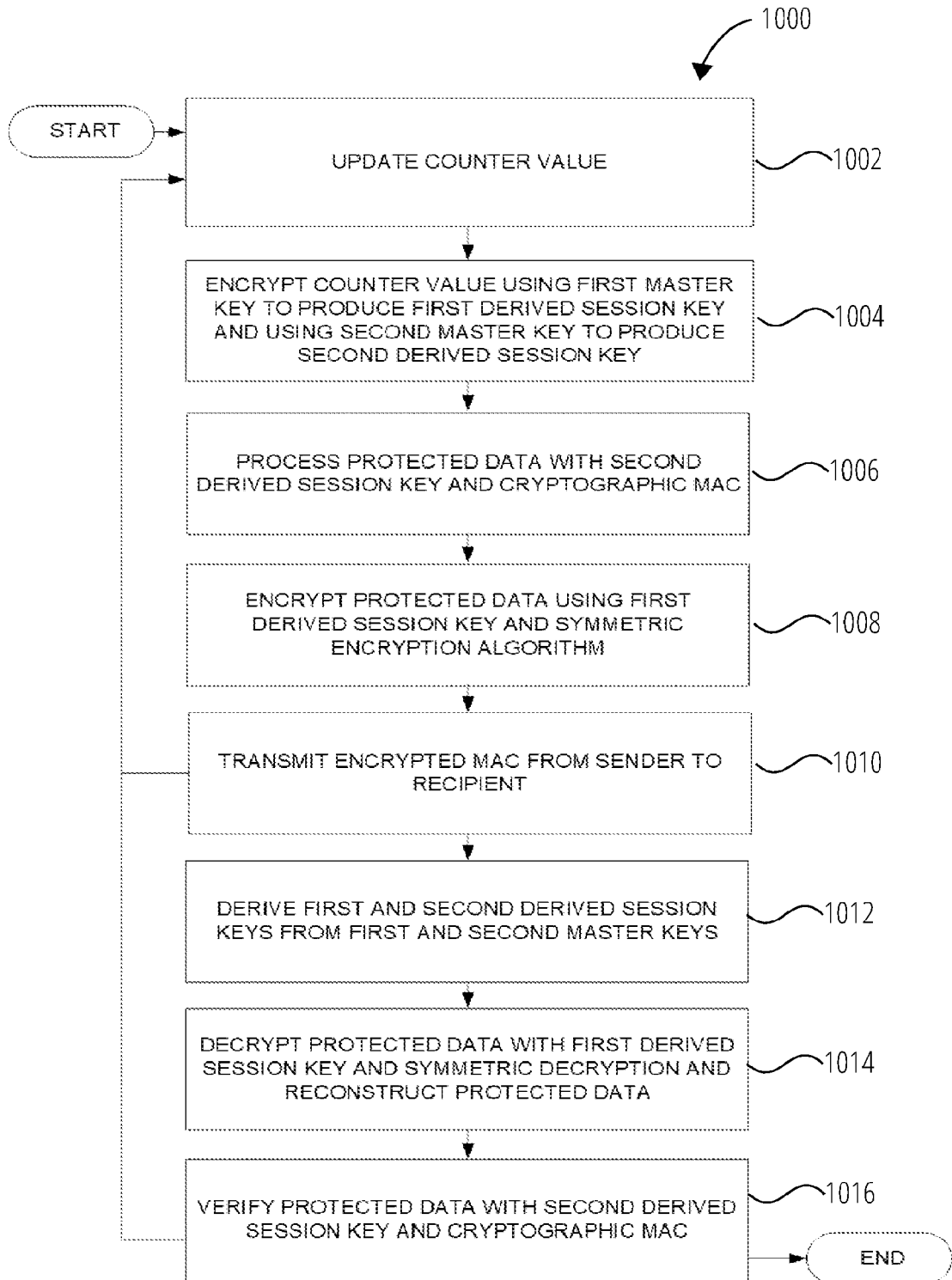


FIG. 10

11/13

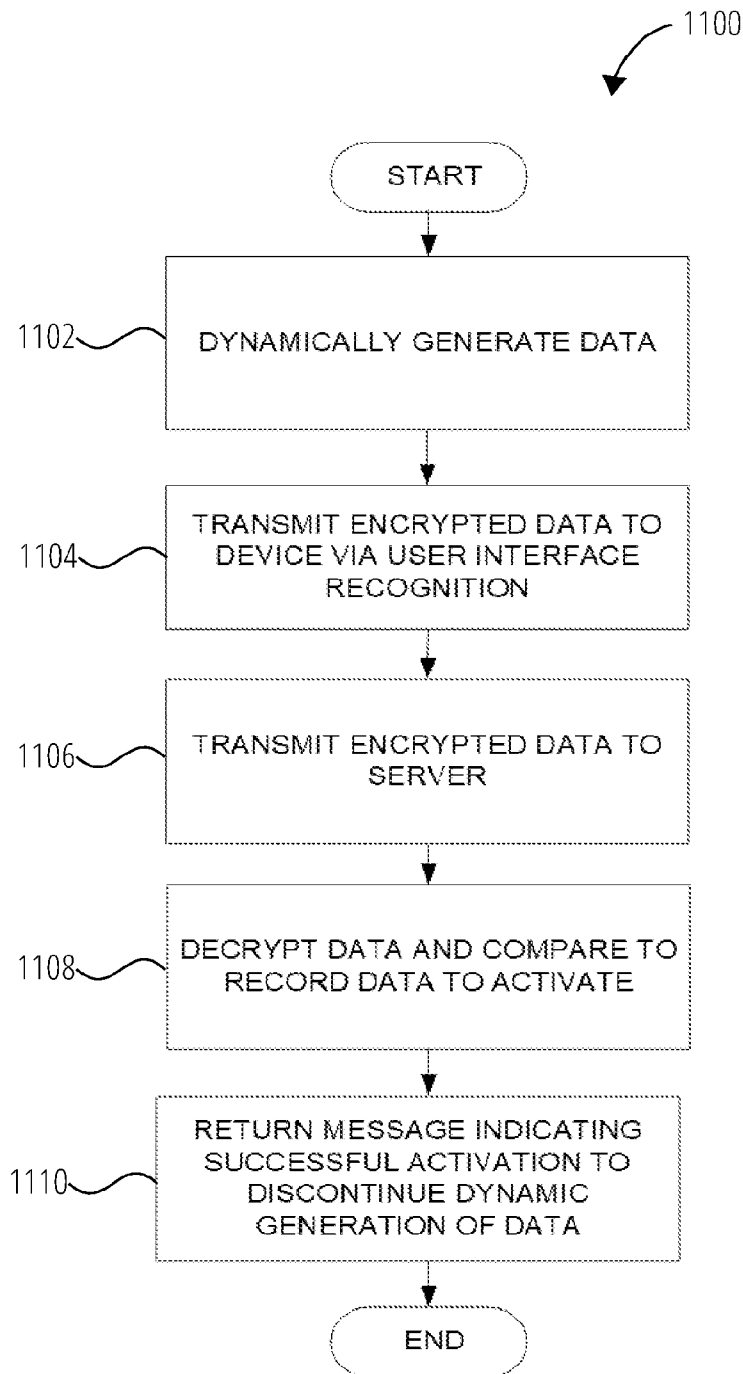


FIG. 11

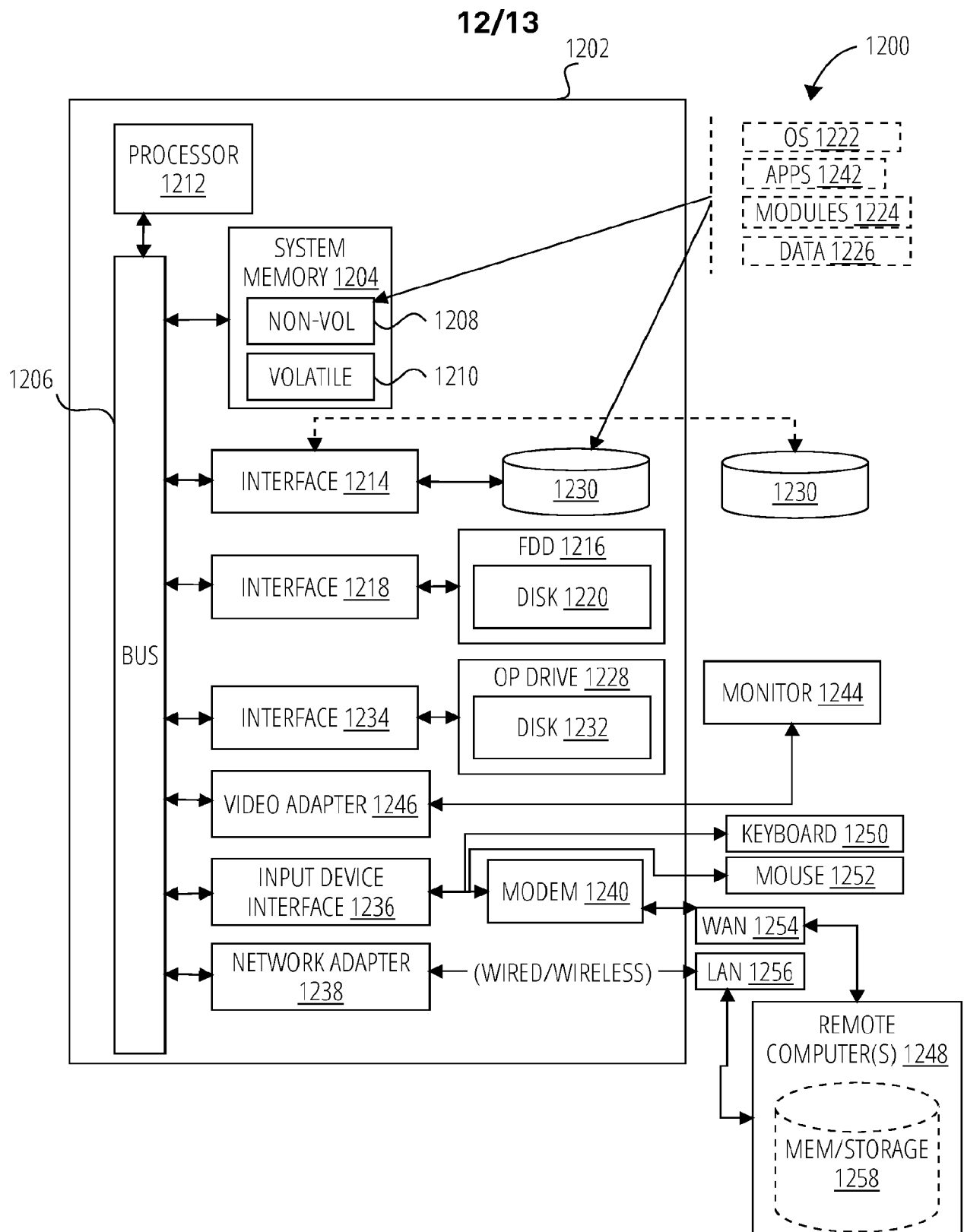


FIG. 12

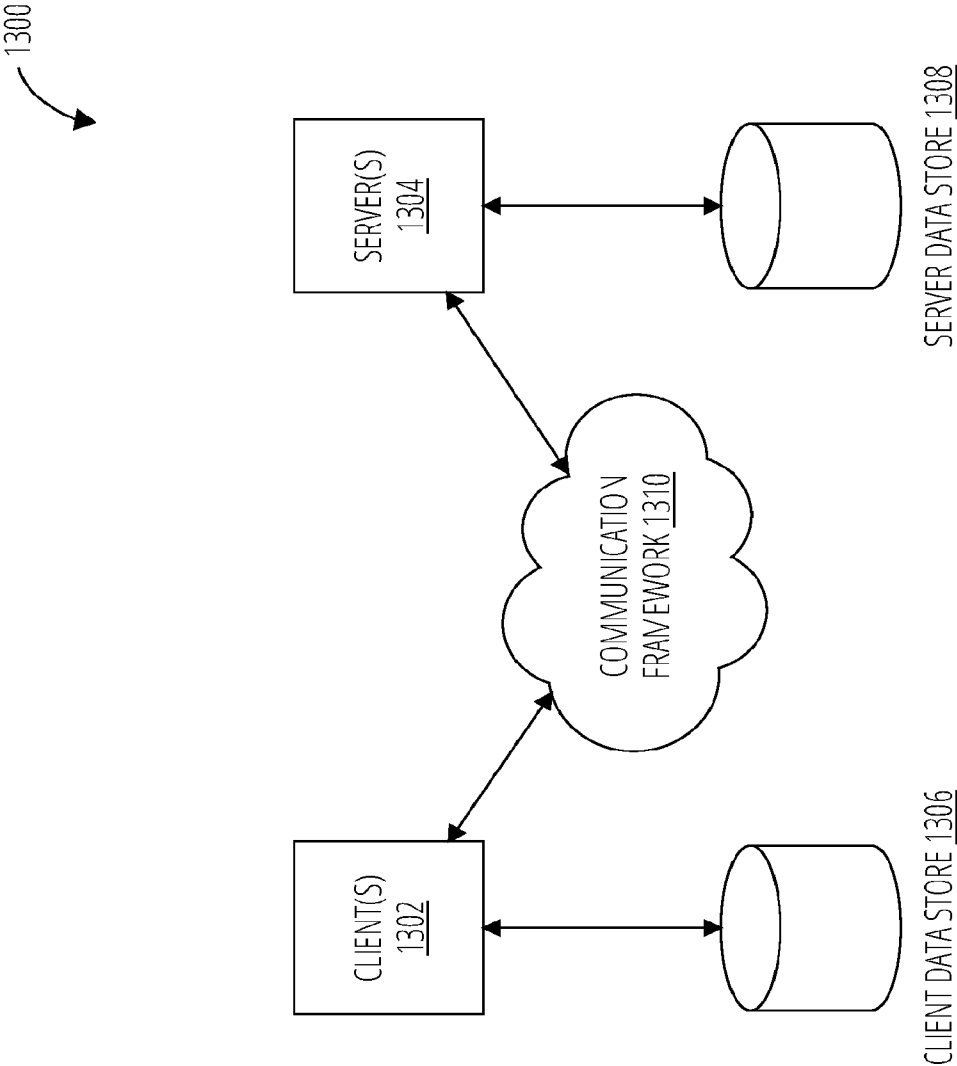


FIG. 13

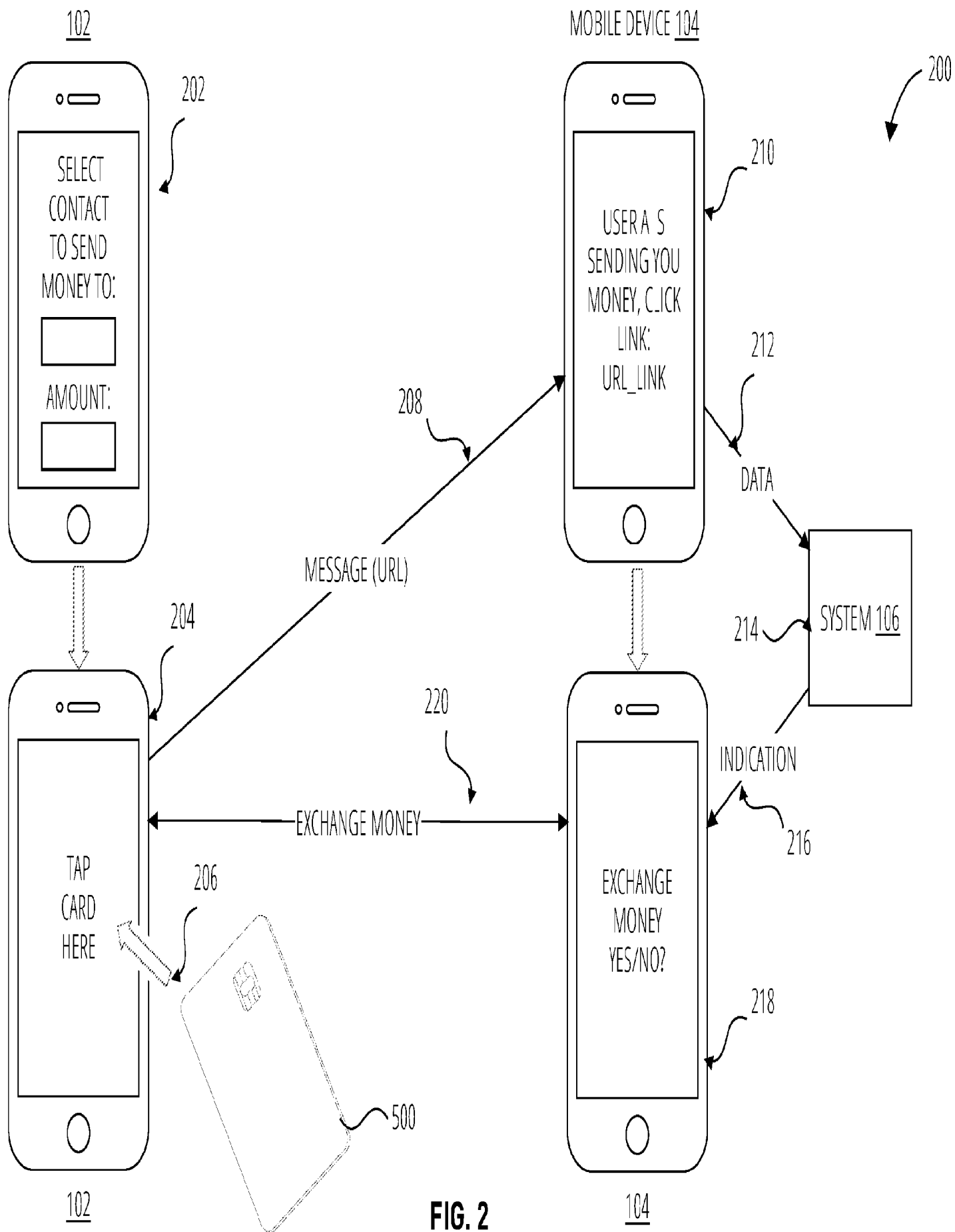


**FIG. 2**