(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
30 November 2023 (30.11.2023)

WIPO | PCT

(10) International Publication Number
WO 2023/229768 A1

(51) International Patent Classification:
H04W 12/30 (2021.01) *H04W 12/069* (2021.01)
H04W 12/47 (2021.01) *H04L 9/40* (2022.01)
H04W 12/106 (2021.01)

(21) International Application Number:
PCT/US2023/019567

(22) International Filing Date:
24 April 2023 (24.04.2023)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
17/664,460 23 May 2022 (23.05.2022) US

(71) Applicant: **CAPITAL ONE SERVICES, LLC.** [US/US]; 1680 Capital One Drive, McLean, Virginia 22102 (US).

(72) Inventors: **LUTZ, Wayne**; c/o Capital One Services, LLC., 1680 Capital One Drive, McLean, Virginia 22102 (US). **OSBORN, Kevin**; c/o Capital One Services, LLC., 1680 Capital One Drive, McLean, Virginia 22102 (US). **RULE, Jeffrey**; c/o Capital One Services, LLC., 1680 Capital One Drive, McLean, Virginia 22102 (US).

(74) Agent: **CARR, Nick** et al.; 2601 Weston Parkway Suite 103, Cary, North Carolina 27513 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CV, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, MG, MK, MN, MU, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, CV,

(54) Title: CONFIGURING APPLICATIONS ON A DEVICE USING A CONTACTLESS CARD

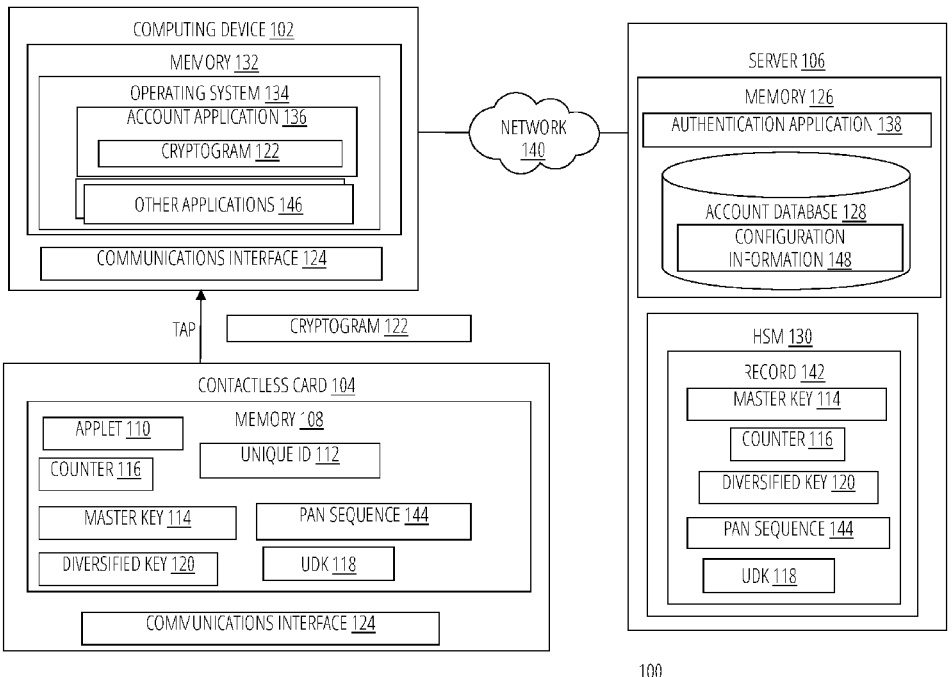


FIG. 1A

(57) Abstract: Systems, methods, apparatuses, and computer-readable media for configuring applications on a device using a contactless card. An application executing on a device may receive a cryptogram from a contactless card associated with an account. The application may transmit the cryptogram to a server. The application may receive, based on the cryptogram, configuration information for a plurality of applications associated with the account. The application may modify a respective configuration of each application on the device based on the configuration information.

WO 2023/229768 A1

[Continued on next page]

GH, GM, KE, LR, LS, MW, MZ, NA, RW, SC, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, ME, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

CONFIGURING APPLICATIONS ON A DEVICE USING A CONTACTLESS CARD

RELATED APPLICATIONS

[0001] This application claims priority to U.S. Patent Application Serial No. 17/664,460, entitled "CONFIGURING APPLICATIONS ON A DEVICE USING A CONTACTLESS CARD" filed on May 23, 2022. The contents of the aforementioned application are incorporated herein by reference in their entirety.

BACKGROUND

[0002] Configuring a new device or additional devices can be tedious, time consuming, and error prone. For example, a user may purchase a new device to replace another device. However, the new device does not include applications and/or configuration from the other device. Conventional solutions to configure devices are vulnerable to errors, have security risks, and are susceptible to malicious attacks.

SUMMARY

[0003] Systems, methods, apparatuses, and computer-readable media for configuring applications on a device using a contactless card. In one aspect, a method, includes receiving, by an application executing on a device, a cryptogram from a contactless card associated with an account, transmitting, by the application, the cryptogram to a server, receiving, by the application based on the cryptogram, configuration information for a plurality of applications associated with the account, and modifying a respective configuration of each application on the device based on the configuration information.

BRIEF DESCRIPTION OF THE SEVERAL VIEWS OF THE DRAWINGS

[0004] To easily identify the discussion of any particular element or act, the most significant digit or digits in a reference number refer to the figure number in which that element is first introduced.

[0005] FIG. 1A illustrates an aspect of the subject matter in accordance with one embodiment.

[0006] FIG. 1B illustrates an aspect of the subject matter in accordance with one embodiment.

[0007] FIG. 1C illustrates an aspect of the subject matter in accordance with one embodiment.

[0008] FIG. 1D illustrates an aspect of the subject matter in accordance with one embodiment.

[0009] FIG. 2A illustrates an aspect of the subject matter in accordance with one embodiment.

[0010] FIG. 2B illustrates an aspect of the subject matter in accordance with one embodiment.

[0011] FIG. 2C illustrates an aspect of the subject matter in accordance with one embodiment.

[0012] FIG. 2D illustrates an aspect of the subject matter in accordance with one embodiment.

[0013] FIG. 3 illustrates a routine 300 in accordance with one embodiment.

[0014] FIG. 4 illustrates a routine 400 in accordance with one embodiment.

[0015] FIG. 5 illustrates a routine 500 in accordance with one embodiment.

[0016] FIG. 6A illustrates a contactless card 104 in accordance with one embodiment.

[0017] FIG. 6B illustrates a contactless card 104 in accordance with one embodiment.

[0018] FIG. 7 illustrates a data structure 700 in accordance with one embodiment.

[0019] FIG. 8 illustrates a computer architecture 800 in accordance with one embodiment.

DETAILED DESCRIPTION

[0020] Embodiments disclosed herein provide secure techniques to automatically configure a device, such as a smartphone, tablet, or any other computing device, using a contactless card. Examples of configuring a device may include installing applications on the device, uninstalling applications from the device, and/or configuring applications on the device (e.g., modifying application settings, application parameters, etc.). Initially, a user may register one or more applications and/or configuration information for one or more applications. Doing so may include storing information for a given device and/or account in an account database. The registration may associate the applications and/or configuration information with a contactless card.

[0021] Subsequently, a user may tap the contactless card to any device, which may include previously registered devices, new devices, existing devices, etc. In some embodiments, tapping the card to the device may cause the card to generate a cryptogram. In some embodiments, the cryptogram is a parameter of a uniform resource locator (URL). When the device receives the URL and/or the cryptogram, the device may launch an application, which may transmit the cryptogram to a server for verification. The server may verify the cryptogram based at least in part on decrypting the cryptogram.

[0022] In other embodiments, tapping the card to the device may cause the device to launch a web browser based on the URL. In such embodiments, the web browser may access a page associated with the URL. The server may identify the cryptogram as a parameter of the URL

accessed by the web browser. The server may verify the cryptogram based at least in part on decrypting the cryptogram.

[0023] Regardless of the application that is launched responsive to tapping the card to the device, subsequent to verifying the cryptogram, the server may use the cryptogram (and/or a parameter of the cryptogram) to identify the associated applications and/or configuration information for the applications in the account database. The server may then transmit the configuration information to the requesting device. In some embodiments, the server sends the configuration information to the device via one or more push notifications. In some embodiments, such as when the configuration information has a size that exceeds a threshold size, the server may send one or more tokens to the device. The device may then use the tokens to request the configuration information from the server.

[0024] Once received, the device may apply the received configuration information to each application. For example, the configuration information may include account login credentials, server information to establish connections to one or more servers, user preferences, settings, and the like. In some embodiments, the configuration information may include information for one or more applications that are not installed on the device. In such embodiments, these applications may be downloaded and installed on the device based on the receipt of the configuration information. Similarly, the configuration information may include information for disabling or uninstalling existing applications. In such embodiments, the device may disable and/or uninstall these applications based on the configuration information. Furthermore, the device may prevent some applications from being enabled and/or reinstalled on the device based on the configuration information.

[0025] Advantageously, embodiments disclosed herein provide secure techniques to automatically configure devices using a contactless card. By requiring cryptographic verification to initiate the configuration, the security of the user's account, applications, devices, and data are improved. Furthermore, doing so ensures that the automatic device configuration is only performed when the user has access to a contactless card that facilitates the cryptogram verification with the server. Additionally, by providing the disclosed automatic configuration functionality, many different applications, operating systems, and/or devices can be automatically configured without requiring integration into every application, operating system, and/or device.

[0026] With general reference to notations and nomenclature used herein, the detailed descriptions herein may be presented in terms of program procedures executed on a computer or network of computers. These procedural descriptions and representations are used by those skilled in the art to effectively convey the substance of their work to others skilled in the art.

[0027] A procedure is here, and generally, conceived to be a self-consistent sequence of operations leading to a desired result. These operations are those requiring physical manipulations of physical quantities. Usually, though not necessarily, these quantities take the form of electrical, magnetic or optical signals capable of being stored, transferred, combined, compared, and otherwise manipulated. It proves convenient at times, principally for reasons of common usage, to refer to these signals as bits, values, elements, symbols, characters, terms, numbers, or the like. It should be noted, however, that all of these and similar terms are to be associated with the appropriate physical quantities and are merely convenient labels applied to those quantities.

[0028] Further, the manipulations performed are often referred to in terms, such as adding or comparing, which are commonly associated with mental operations performed by a human operator. No such capability of a human operator is necessary, or desirable in most cases, in any of the operations described herein, which form part of one or more embodiments. Rather, the operations are machine operations. Useful machines for performing operations of various embodiments include digital computers or similar devices.

[0029] Some embodiments may be described using the expression "coupled" and "connected" along with their derivatives. These terms are not necessarily intended as synonyms for each other. For example, some embodiments may be described using the terms “connected” and/or “coupled” to indicate that two or more elements are in direct physical or electrical contact with each other. The term "coupled," however, may also mean that two or more elements are not in direct contact with each other, but yet still co-operate or interact with each other.

[0030] Various embodiments also relate to apparatus or systems for performing these operations. This apparatus may be specially constructed for the required purpose or it may comprise a computer as selectively activated or reconfigured by a computer program stored in the computer. The procedures presented herein are not inherently related to a particular computer or other apparatus. Various machines may be used with programs written in accordance with the teachings herein, or it may prove convenient to construct more specialized

apparatus to perform the required method steps. The required structure for a variety of these machines will appear from the description given.

[0031] Reference is now made to the drawings, wherein like reference numerals are used to refer to like elements throughout. In the following description, for purposes of explanation, numerous specific details are set forth in order to provide a thorough understanding thereof. However, the novel embodiments can be practiced without these specific details. In other instances, structures and devices are shown in block diagram form in order to facilitate a description thereof. The intention is to cover all modifications, equivalents, and alternatives consistent with the claimed subject matter.

[0032] In the Figures and the accompanying description, the designations "a" and "b" and "c" (and similar designators) are intended to be variables representing any positive integer. Thus, for example, if an implementation sets a value for a = 5, then a complete set of components 123 illustrated as components 123-1 through 123-*a* (or 123a) may include components 123-1, 123-2, 123-3, 123-4, and 123-5. The embodiments are not limited in this context.

[0033] Figure 1A depicts an exemplary computing architecture 100, also referred to as a system, consistent with disclosed embodiments. Although the computing architecture 100 shown in Figures 1A-1C has a limited number of elements in a certain topology, it may be appreciated that the computing architecture 100 may include more or less elements in alternate topologies as desired for a given implementation.

[0034] The computing architecture 100 comprises one or more computing devices 102, one or more servers 106, and one or more contactless cards 104. The contactless card 104 is representative of any type of card, such as a credit card, debit card, ATM card, gift card, payment card, smart card, and the like. The contactless card 104 may comprise one or more communications interfaces 124, such as a radio frequency identification (RFID) chip, configured to communicate with a communications interface 124 (also referred to herein as a "card reader", a "wireless card reader", and/or a "wireless communications interface") of the computing devices 102 via NFC, the EMV standard, or other short-range protocols in wireless communication. Although NFC is used as an example communications protocol herein, the disclosure is equally applicable to other types of wireless communications, such as the EMV standard, Bluetooth, and/or Wi-Fi.

[0035] The computing device 102 is representative of any number and type of computing device, such as smartphones, tablet computers, wearable devices, laptops, portable gaming

devices, virtualized computing system, merchant terminals, point-of-sale systems, servers, desktop computers, and the like. A mobile device may be used as an example of the computing device 102, but should not be considered limiting of the disclosure. The server 106 is representative of any type of computing device, such as a server, workstation, compute cluster, cloud computing platform, virtualized computing system, and the like. Although not depicted for the sake of clarity, the computing device 102, contactless card 104, and server 106 each include one or more processor circuits, e.g., to execute programs, code, and/or instructions.

[0036] As shown, a memory 108 of the contactless card 104 includes an applet 110, a counter 116, one or more master keys 114, one or more diversified keys 120, a unique ID 112, a primary account number (PAN) sequence number 144, and one or more Unique Derived Keys (UDKs) 118. The unique ID 112 may be any identifier that uniquely identifies the contactless card 104 relative to other contactless cards 104. The PAN sequence 144 may include a counter value stored by the contactless card 104. The applet 110 is executable code configured to perform some or all of the operations described herein. The counter 116 is a value that is synchronized between the contactless card 104 and server 106. The counter 116 may comprise a number that changes each time data is exchanged between the contactless card 104 and the server 106 (and/or the contactless card 104 and the computing device 102). The counter 116, master keys 114, diversified keys 120, UDKs 118, PAN sequence 144, and/or unique ID 112 are used to provide security in the system 100 as described in greater detail below.

[0037] As shown, a memory 132 of the computing device 102 includes an instance of an operating system 134. Example operating systems include the Android® OS, iOS®, macOS®, Linux®, and Windows® operating systems. As shown, the operating system 134 includes an account application 136 and one or more other applications 146. The account application 136 allows users to perform various operations, such as configuring the device 102 based on the configuration information 148 stored by the server 106. In some embodiments the account application 136 may further perform other account-related operations such as activating payment cards, viewing account balances, purchasing items, processing payments, and the like. In some embodiments, a user may authenticate using authentication credentials to access certain features of the account application 136. For example, the authentication credentials may include a username (or login) and password, biometric credentials (e.g., fingerprints, Face ID, etc.), and the like. The other applications 146 are representative of any type of application, such as web browsers, email clients, messaging clients, social media applications, and the like.

[0038] As shown, a memory 126 of the server 106 includes an authentication application 138 and an account database 128. The account database 128 generally includes information related to an account holder (e.g., one or more users), one or more accounts of the account holder, and one or more contactless cards 104 of the account. For example, as shown, the account database 128 may include configuration information 148 for a plurality of accounts. The configuration information 148 may include any data to configure the computing device 102, the operating system 134, the account application 136, the other applications 146, and/or any component thereof. For example, the configuration information 148 may include information for connecting to one or more servers (e.g., internet protocol (IP) addresses, ports, etc.), account information (e.g., login/password), inbox contents (e.g., messages, emails, etc.), preferences, variable values, and/or settings. The configuration information 148 may further include executable files, scripts, etc., to install and/or configure one or more portions of the configuration information 148 on a given computing device 102. Embodiments are not limited in this context, as the configuration information 148 may be used to modify any attribute and/or feature of a computing device 102 and/or any software component thereof.

[0039] As stated, the configuration information 148 may be used to automatically configure the computing device 102. Stated differently, the configuration information 148 may represent a "backup" of at least a portion of a given device and/or account (including any applications, operating systems, and/or parameters thereof, such that any device may be restored to a state similar to the backup. For example, when a user purchases a new smartphone, the configuration information 148 from a previous smartphone may be used to automatically configure the new smartphone. For example, the configuration information 148 may be used to install applications on the new smartphone (e.g., applications that are not included with the smartphone when purchased but were installed on the previous smartphone and registered as part of the configuration information 148), remove applications from the new smartphone that were not installed on the previous smartphone (and/or remove restricted applications based on the configuration information 148), apply settings and/or parameters from the old smartphone to the new smartphone, and the like.

[0040] In various embodiments, a user may initially register the configuration information 148 with the server 106. For example, the user may access a feature of the account application 136, access a web page, or any other resource to register the configuration information 148 for any device and/or account. The configuration information 148 may be specific to all devices

associated with an account and/or a subset of devices associated with an account. In addition and/or alternatively, the configuration information 148 may be specific to all applications associated with an account and/or a subset of applications associated with an account. In some embodiments, the account application 136 may gather information describing the computing device 102, e.g., installed applications, configuration files, etc., which may be stored as the configuration information 148. In other embodiments, a user may provide input specifying different applications, settings, features, etc., of a given device. In some embodiments, the configuration information 148 includes package names for each application to be registered, metadata describing the operating system 134 (e.g., device identifiers, device version, preferences, settings, etc.) to be registered, metadata describing each application to be registered, preferences, login information, application identities, or any other attribute of the device, application, operating system 134, and/or component thereof. Furthermore, the configuration information 148 may be associated with an account in the account database 128, which allows the server 106 to further associate the configuration information 148 with one or more contactless cards 104 associated with the account.

[0041] For example, a user may register a social media application and an email client with the server 106. Therefore, the configuration information 148 for the user may include respective package names of the social media application and the email client, connection information for the media application and the email client to connect to respective servers, user identities in the social media application and the email application, an email inbox (or folders) for the email client, social media profiles in the social media application, social media messages in the social media application, respective versions of the social media application and the email client, display settings (e.g., enabling dark mode in either application), enabling push notifications in either application, permissions for each application within the operating system 134, and the like. Embodiments are not limited in this context. Indications of this information may therefore be stored as the configuration information 148 for the account.

[0042] Thereafter, the user may desire to use the configuration information 148 to configure the computing device 102. For example, the computing device 102 may be a new smartphone, and the user may wish to configure the social media application and email client (e.g., to restore all emails in the email client, load their social media profile/settings in the social media application, etc.). In some embodiments, these applications may not be installed on the new smartphone, so the configuration information 148 may further be used to download the social

media application and/or the email client for installation on the computing device 102. To initiate the configuration of a device based on the configuration information 148, the user may tap the contactless card 104 to the computing device 102.

[0043] In the embodiment depicted in Figure 1A, the user may tap the contactless card 104 to the computing device 102 (or otherwise bring the contactless card 104 within communications range of the communications interface 124 of the device 102). The applet 110 may then generate a cryptogram 122. The cryptogram 122 may be generated based on any suitable cryptographic technique. In some embodiments, the cryptogram 122 may be based on the unique ID 112 of the contactless card 104. In some embodiments, the applet 110 may include the cryptogram 122 and an unencrypted identifier (e.g., the counter 116, the PAN sequence 144, the unique ID 112, and/or any other unique identifier) as part of a data package including the cryptogram 122. In at least one embodiment, the data package is an NDEF file.

[0044] As stated, the computing architecture 100 is configured to implement key diversification to secure data, which may be referred to as a key diversification technique herein. Generally, the server 106 (or another computing device) and the contactless card 104 may be provisioned with the same master key 114 (also referred to as a master symmetric key). More specifically, each contactless card 104 is programmed with a distinct master key 114 that has a corresponding pair in the hardware security module (HSM) 130 of the server 106. For example, when a contactless card 104 is manufactured, a unique master key 114 may be programmed into the memory 108 of the contactless card 104. Similarly, the unique master key 114 may be stored in a record 142 in the HSM 130.

[0045] Furthermore, when a given card 104 is manufactured, a UDK 118 may be diversified from the master key 114 via an HSM function that takes, as input, a diversification factor and a reference to the master key 114 index in the HSM 130 (e.g., an index to the record 142). In some embodiments, the diversification factor may be the unique ID 112 and/or PAN sequence 144 of the contactless card 104. The UDK 118 may be stored in the contactless card 104 and the record 142 of the HSM 130. The master key 114 and UDK 118 may be kept secret from all parties other than the contactless card 104 and server 106, thereby enhancing security of the system 100. Although depicted as being stored in the record 142, in some embodiments, the counter 116 and/or PAN sequence 144 are not stored in the HSM 130. For example, the unique ID 112, counter 116, and PAN sequence 144 may be stored in the account database 128.

[0046] In some embodiments, to generate the cryptogram 122, the applet 110 may provide the UDK 118, unique ID 112, and a diversification factor as input to a cryptographic algorithm, thereby producing a diversified key 120. In some embodiments, the diversification factor is the counter 116. In other embodiments, the PAN sequence 144 is the diversification factor. The diversified key 120 may then be used to encrypt some data, such as the diversification factor (e.g., the counter 116 and/or the PAN sequence 144) or other sensitive data. The applet 110 and the server 106 may be configured to encrypt the same type of data to facilitate the decryption and/or verification processing of a cryptogram.

[0047] As stated, the UDKs 118 of the contactless card 104 and server 106 may be used in conjunction with the counters 116 to enhance security using key diversification. As stated, the counters 116 comprise values that are synchronized between the contactless card 104 and server 106. The counter 116 may comprise a number that changes each time data is exchanged between the contactless card 104 and the server 106 (and/or the contactless card 104 and the computing device 102). When preparing to send data (e.g., to the server 106 and/or the device 102), the applet 110 of the contactless card 104 may increment the counter 116. The applet 110 of the contactless card 104 may then provide the UDK 118, unique ID 112, and counter 116 as input to a cryptographic algorithm, which produces a diversified key 120 as output. The cryptographic algorithm may include encryption algorithms, hash-based message authentication code (HMAC) algorithms, cipher-based message authentication code (CMAC) algorithms, and the like. Non-limiting examples of the cryptographic algorithm may include a symmetric encryption algorithm such as 3DES or AES107; a symmetric HMAC algorithm, such as HMAC-SHA-256; and a symmetric CMAC algorithm such as AES-CMAC. Examples of key diversification techniques are described in greater detail in United States Patent Application 16/205,119, filed November 29, 2018. The aforementioned patent application is incorporated by reference herein in its entirety. In some embodiments, the PAN sequence 144 is used as input to the cryptographic algorithm instead of the counter 116 to generate the diversified key 120, e.g., by encrypting the UDK 118, unique ID 112 and PAN sequence 144.

[0048] The applet 110 may then encrypt some data (e.g., the unique ID 112, the counter 116, the PAN sequence 144, a command, and/or any other data) using the diversified key 120 and the data as input to the cryptographic algorithm. For example, encrypting the unique ID 112 with the diversified key 120 may result in an encrypted unique ID 112 (e.g., a cryptogram 122). As stated, the applet 110 and the server 106 may be configured to encrypt the same data.

[0049] In some embodiments, two diversified keys 120 may be generated, e.g., based on one or more portions of the input to the cryptographic function. In some embodiments, the two diversified keys 120 are generated based on two distinct master keys 114, two distinct UDKs 118, the unique ID 112, and the counter 116 (or the PAN sequence 144). In such embodiments, a message authentication code (MAC) is generated using one of the diversified keys 120, and the MAC may be encrypted using the other one of the diversified keys 120. The MAC may be generated based on any suitable data input to a MAC algorithm, such as sensitive data, the unique ID 112, the counter 116, and/or the PAN sequence 144. More generally, the applet 110 and the server 106 may be configured to generate the MAC based on the same data. In some embodiments, the cryptogram 122 is included in a data package such as an NDEF file. The account application 136 may then read the data package including cryptogram 122 via the communications interface 124 of the computing device 102.

[0050] In some embodiments, the applet includes the cryptogram 122 as a parameter of a URL. For example, the URL may be "http://www.example.com/appconfig?param=ABC123&custID=123". In such an example, the cryptogram 122 may correspond to the parameter "ABC123" and the unique ID 112 may correspond to the parameter "custID". Once received by the operating system 134, the operating system 134 may open an application to process the URL. In some embodiments, the URL may be registered with the account application 136, which causes the operating system 134 to launch the account application 136 and provide the URL 120 as input to the account application 136. However, in other examples, the operating system 134 may launch a web browser and provide the URL as input to the web browser. In other embodiments, the cryptogram 122 is not a parameter of the URL but is transmitted with the URL in a data package such as an NDEF file. In such embodiments, the operating system 134 may read the data package including the URL and cryptogram 122 via the communications interface 124 of the computing device 102.

[0051] In the example depicted in Figure 1A, the operating system 134 may launch the account application 136 based on reading the cryptogram 122. As stated, the cryptogram 122 may be a parameter of a URL. In such examples, the URL may be associated with the account application 136. Doing so causes the operating system 134 to launch the account application 136. The account application 136 may then extract the cryptogram 122 and transmit the cryptogram 122 to the server 106 for verification.

[0052] In embodiments where the cryptogram 122 is not a parameter of the URL, the operating system 134 may launch the account application 136 based on reading the cryptogram 122 and/or the NDEF file including the cryptogram 122. The operating system 134 may provide the cryptogram 122 to the account application 136. The account application 136 may then extract the cryptogram 122 and transmit the cryptogram 122 to the server 106 for verification.

[0053] In some embodiments the account application 136 (or another application to configure the computing device 102 based on the configuration information 148) may not be installed on the device. In some such embodiments, the operating system 134 may cause the account application 136 to be downloaded and installed on the computing device 102 based on the URL and/or the cryptogram 122. In other such embodiments, an application clip (e.g., an Apple® App Clip), instant application (e.g., an Android instant application), or progressive web application (e.g., an Android progressive web application) that includes the disclosed functionality of the account application 136 may be downloaded and installed on the computing device 102 to configure the computing device 102 based on the configuration information 148. In such embodiments, the app clips, instant applications, and/or progressive web applications may automatically be removed from the computing device 102 after the configuration information 148 is used to configure the computing device 102.

[0054] Figure 1B depicts an embodiment where the account application 136 transmits the cryptogram 122 to the server 106. The server 106 may provide the cryptogram 122 to the authentication application 138 and/or the HSM 130 for verification based at least in part on the instance of the master key 114 and/or UDK 118 stored by the server 106. In some embodiments, the authentication application 138 and/or the HSM 130 may identify the UDK 118 (or master key 114) and counter 116 using the unencrypted unique ID 112 provided to the server 106. In examples where the PAN sequence 144 is used to generate the cryptogram 122, the server 106 may identify the PAN sequence 144 in the account database 128 and/or HSM 130 using the unencrypted unique ID 112. In some examples, the authentication application 138 may provide the UDK 118, unique ID 112, and counter 116 as input to the cryptographic function of the HSM 130, which produces one or more diversified keys 120 as output. In other embodiments, the server encrypts the UDK 118, unique ID 112, and PAN sequence 144 to generate the diversified keys 120. The resulting diversified keys 120 may correspond to the diversified keys 120 of the contactless card 104, which may be used to decrypt the cryptogram 122 and/or verify the MAC once decrypted. For example, the server 106 may generate a MAC

based on the same data as the applet 110, e.g., the sensitive data, the unique ID 112, the counter 116, and/or the PAN sequence 144. If the MAC generated by the server 106 matches the decrypted MAC in the cryptogram 122, the server 106 may verify or otherwise authenticate the cryptogram 122.

[0055] Regardless of the decryption technique used, the authentication application 138 and/or the HSM 130 may successfully decrypt the cryptogram 122 and verify the MAC, thereby verifying and/or authenticating the cryptogram 122. If the decryption and/or MAC verification is successful, the authentication application 138 may reference the account database 128 to identify the configuration information 148 for the account. In some embodiments, the authentication application 138 may access the configuration information 148 based on the unencrypted unique ID 112 provided to the server 106. Embodiments are not limited in this context, as any suitable identifier may be used to access the configuration information 148 for an account. The server 106 may further transmit, to the computing device 102 (e.g., to the account application 136 and/or the operating system 134), a response indicating that the server 106 verified the cryptogram 122 and that the configuration information 148 can be used to configure the computing device 102.

[0056] The authentication application 138 may then parse the configuration information 148 to identify each application, operating system, and/or setting registered therein. Continuing with the previous example, the configuration information 148 may include indications of the social media application and the email client. As described in greater detail below, the server 106 may then transmit the configuration information 148 (and/or portions thereof) to the computing device 102.

[0057] Returning to the decryption, if the authentication application 138 is unable to decrypt the cryptogram 122 (and/or is unable to verify the MAC) the authentication application 138 does not validate the cryptogram 122. In such an example, the authentication application 138 determines to refrain from accessing the configuration information 148 to automatically configure the computing device 102. The authentication application 138 may transmit, to the computing device 102, a response comprising an indication of the failed decryption and/or verification and an indication that configuration information 148 cannot be used to configure the computing device 102.

[0058] Figure 1C depicts an embodiment where the authentication application 138 transmits a push notification 150 that includes the configuration information 148 to the account application

136 for each identified application or other software component in the configuration information 148 in the account database 128. In some embodiments, the push notification 150 includes an indication that the server 106 verified the cryptogram 122 and that the configuration information 148 can be used to configure the computing device 102. Although depicted as being transmitted in a single push notification, in some embodiments, the server 106 may transmit multiple push notifications to the computing device 102, where each push notification includes at least a portion of the configuration information 148. In some embodiments, a given push notification may be received in a background of the operating system 134 (e.g., background push notifications that may not be outputted for display on the computing device 102 but are sent directly to the corresponding application and/or to the operating system 134). In some embodiments, the push notifications are foreground push notifications (e.g., notifications that are displayed on the computing device 102). In some embodiments, the configuration information 148 includes one or more configuration files.

[0059] Furthermore, in some embodiments, a size of the configuration information 148 (and/or a portion thereof) may exceed a size threshold (e.g., 1 megabyte, 10 megabytes, etc.). In such embodiments, the server 106 may generate one or more tokens, associate each token with one or more portions of the configuration information 148, and transmit the one or more tokens to the computing device 102. The computing device 102 may then transmit the tokens to the server 106 to pull the relevant configuration information 148 (and/or a portion thereof) from the server 106. For example, by associating the token with configuration information 148, the server 106 may provide the associated configuration information 148 based on receipt of the token from the account application 136, the operating system 134, and/or the other applications 146.

[0060] In some embodiments, the computing device 102 may transmit the configuration information 148 and/or tokens to the contactless card 104. The contactless card 104 may then store the configuration information 148 and/or tokens in the memory 108. Subsequently, the computing device 102 may receive the configuration information 148 from the contactless card 104 (e.g., via NFC responsive to a tap of the contactless card 104 to the computing device 102) to configure the computing device 102 and any component thereof as described herein.

[0061] Furthermore, as shown, Figure 1C depicts an embodiment where the computing device 102 includes three different applications in addition to the account application 136, e.g., other application 146-1, other application 146-2, and other application 146-N. Advantageously, the

account application 136 may use the configuration information 148 to configure these applications on the computing device 102.

[0062] Figure 1D depicts an embodiment where the account application 136 uses one or more portions of the configuration information 148 received from the server 106 to configure the computing device 102. As stated, the account application 136 may condition the use of the configuration information 148 based on the response from the server 106 indicating that the server 106 verified the cryptogram 122 and that the configuration information 148 can be used to configure the computing device 102. As shown, the account application 136 uses configuration information 148-1 to configure other application 146-1 and configuration information 148-N to configure other application 146-N. In some embodiments, the account application 136 may parse the configuration information 148 to generate the configuration information 148-1 through 148-N. In other embodiments, the server 106 generates the configuration information 148 to include multiple files, one file for each application or other entity being configured on the computing device 102. Therefore, in such embodiments, the configuration information 148-1 through 148-N is generated by the server 106. In some embodiments, at least a portion of the configuration information 148 is associated with the account application 136 and is therefore used to configure the account application 136 (e.g., load user profiles, settings, etc.).

[0063] In some embodiments, the account application 136 saves (or writes) the configuration information 148 as a respective configuration file in a location associated with each application and/or the operating system 134. For example, the account application 136 may store the configuration information 148-1 as a configuration file for the other application 146-1 (e.g., with a file name and/or in a file location associated with files that store configuration information for the application 146-1). As another example, the account application 136 may use a portion of the configuration information 148 to store and/or modify one or more files, registry settings, or other parameters of the operating system 134. In some such examples, the account application 136 may provide the configuration information 148 to the operating system 134, where the operating system 134 may use the configuration information 148 to modify the operating system 134. In some embodiments, using the configuration information 148 may cause new configuration information to be written to the computing device 102, may cause some existing configuration information of the computing device 102 to be modified (e.g., by

changing values, settings, etc.), and/or may cause other existing configuration information of the computing device 102 to remain unmodified.

[0064] Furthermore, the account application 136 uses configuration information 148-3 to determine that an application 146-3 is not installed on the computing device 102. In such embodiments, the account application 136 uses the configuration information 148-3 (e.g., the package name of the application 146-3) to download the application 146-3 from the Internet. The account application 136 may then use the configuration information 148-3 to configure application 146-3 as described herein.

[0065] Similarly, the account application 136 uses the configuration information 148 to determine that other application 146-2 is not registered in the configuration information 148. In some embodiments, the account application 136 determines that the other application 146-2 is a restricted application based on the configuration information 148. For example, an enterprise may restrict the use of social media applications on enterprise devices. Therefore, based on the configuration information 148, the account application 136 causes the computing device 102 to uninstall or otherwise disable the other application 146-2. In some embodiments, the account application 136 may permanently restrict a user from reinstalling or otherwise enabling the other application 146-2 (e.g., by configuring the operating system 134 to restrict the other application 146-2 from being installed and/or enabled on the computing device 102.

[0066] In some embodiments, the account application 136 may need additional permissions to configure the computing device 102. For example, the account application 136 may need access to storage media (e.g., non-volatile and/or volatile memory, not pictured for clarity) of the computing device 102, e.g., to read, write, or otherwise access the storage. In such embodiments, the account application 136 and/or the operating system 134 may request to grant the account application 136 access to the storage, which may be authorized by the user, and in turn, by the operating system 134.

[0067] Therefore, by configuring the computing device 102 using the configuration information 148, the computing device 102 may be automatically configured to include all associated applications, settings, and/or features. For example, by configuring the email client with the configuration information 148, the user may be able to view emails, receive new emails, connect to the email server, and send new emails. Embodiments are not limited in this context. Furthermore, prior to configuring the computing device 102 using the configuration information 148, the user would not be able to view emails, receive emails, connect to the

email server, or send emails with the email client (without manually configuring the email client).

[0068] Figure 2A is a schematic 200 illustrating an embodiment of using a web browser 206 to perform at least a portion of the automatic configuration of the computing device 102. In the example depicted in Figure 2A, the contactless card 104 generates a URL with cryptogram 204 responsive to tapping the contactless card 104 to the computing device 102. The URL with cryptogram 204 may include a cryptogram generated as described above with reference to the cryptogram 122, along with other relevant information (e.g., unencrypted identifiers, etc.). More generally, the URL with cryptogram 204 may be generated as described above with reference to Figures 1A-1D.

[0069] Once read by the operating system 134 (not pictured for clarity), the operating system 134 may launch the web browser 206 (e.g., one of the other applications 146) and provide the URL with cryptogram 204 to the web browser 206. Doing so may cause the web browser 206 to access the URL with cryptogram 204 (or a portion thereof). In some embodiments, the URL with cryptogram 204 is directed to a configuration page for automatically configuring the computing device 102. Embodiments are not limited in this context.

[0070] Figure 2B depicts an embodiment where the web browser 206 accesses the URL with cryptogram 204, e.g., based on a hypertext transfer protocol (HTTP) request that includes the URL with cryptogram 204. As shown, the authentication application 138 may receive the URL with cryptogram 204 based on the HTTP request. The authentication application 138 may then extract the cryptogram from the URL with cryptogram 204 for verification. For example, the server 106 may decrypt the cryptogram and/or verify the MAC contained therein as described above to verify the cryptogram.

[0071] Based on the server 106 verifying the cryptogram (e.g., decrypting the cryptogram in the URL with cryptogram 204 and/or verifying the MAC), the server 106 may reference the account database 128 to identify the associated configuration information 148. The server 106 may further transmit, to the computing device 102 (e.g., to the web browser 206 and/or the operating system 134) based on the verification, a response indicating that the server 106 verified the cryptogram 122 and that the configuration information 148 can be used to configure the computing device 102. In some embodiments, the authentication application 138 may access the configuration information 148 based on the unencrypted unique ID 112 provided to the server 106 in the URL with cryptogram 204. Embodiments are not limited in

this context, as any suitable identifier may be used to access the configuration information 148 for an account.

[0072] If, on the other hand, the server 106 is unable to decrypt the cryptogram and/or verify the MAC, the server 106 does not validate the cryptogram. In such an example, the authentication application 138 determines to refrain from accessing the configuration information 148 to automatically configure the computing device 102. The authentication application 138 may transmit an indication of the failed decryption and/or verification to the computing device 102.

[0073] Figure 2C depicts an embodiment where the server 106 verifies the cryptogram portion of the URL with cryptogram 204. As stated, when the server 106 verifies the cryptogram, the server 106 accesses the configuration information 148 for the account associated with the contactless card 104. The server 106 may then identify a plurality of different applications represented in the configuration information 148. For each identified application, the server 106 may generate a respective push notification that includes the relevant configuration information 148 and/or a portion thereof.

[0074] As shown, the server 106 sends push notification 208, push notification 210, and push notification 212 to the computing device 102. As shown, push notification 208 includes configuration information 148-1, push notification 210 includes configuration information 148-2, and push notification 212 includes configuration information 148-N. In such examples, each respective push notification and each instance of configuration information therein may be associated with a respective application executing on the device. In some embodiments, the push notifications may also include a push notification that is transmitted to the operating system 134, which may include configuration information 148 that is applicable to the operating system 134 (e.g., enabling blue light filters, do not disturb functions, etc.). In some embodiments, the push notifications 208, 210, and/or 212 include a respective indication that the server 106 verified the cryptogram 122 and that the configuration information 148 can be used to configure the computing device 102.

[0075] In some embodiments, the push notifications 208, 210, and/or 212 may be received in a background of the operating system 134 (e.g., background push notifications that may not be outputted for display on the computing device 102 but are sent directly to the corresponding application and/or to the operating system 134). In some embodiments, the push notifications 208, 210, and/or 212 are foreground push notifications (e.g., notifications that are displayed on

the computing device 102). In some embodiments, the configuration information 148 includes one or more configuration files. Furthermore, in some embodiments, a size of the configuration information 148-1 through 148-N may exceed a size threshold. In such embodiments, the server 106 may generate one or more tokens, and include a respective token in each of the push notifications 208, 210, and 212. Once received, each application may then provide the received token to the server 106 to pull the relevant configuration information 148 (and/or a portion thereof) from the server 106.

[0076] Figure 2D depicts an embodiment where the push notifications 208, 210, and 212 are received by the other applications 146-1, 146-2, and 146-N, respectively. Generally, once received, the other applications 146-1, 146-2, and 146-N may use the configuration information 148-1, 148-2, and 148-N, respectively, for configuration. For example, other application 146-1 may be configured based on configuration information 148-1, other application 146-2 may be configured based on configuration information 148-2, and other application 146-N may be configured based on configuration information 148-N. In some embodiments, the other applications 146-1, 146-2, and 146-N and/or the operating system 134 condition the use of the configuration information 148 based on receiving the indication from the server 106 that the server 106 verified the cryptogram 122 and that the configuration information 148 can be used to configure the computing device 102.

[0077] In some embodiments, the configuration information 148 is saved as a respective configuration file in a location associated with each application and/or the operating system 134. For example, the other application 146-1 may store the configuration information 148-1 as a configuration file for the other application 146-1 (e.g., with a file name and/or in a file location associated with files that store configuration information for the application 146-1). As another example, the other application 146-1 may use a portion of the configuration information 148-1 to store and/or modify one or more files, registry settings, or other parameters of the operating system 134 to configure the other application 146-1. In some such examples, the other application 146-1 may provide the configuration information 148-1 to the operating system 134, where the operating system 134 may use the configuration information 148 to modify the operating system 134. Similarly, application 146-2 may store the configuration information 148-2 as a configuration file for the other application 146-2, and application 146-N may store the configuration information 148-N as a configuration file for the other application 146-N.

[0078] In some embodiments, using the configuration information 148-1 through 148-N may cause new configuration information to be written to the computing device 102, may cause some existing configuration information of the computing device 102 to be modified (e.g., by changing values, settings, etc.), and/or may cause other existing configuration information of the computing device 102 to remain unmodified.

[0079] Similar to the embodiments depicted in Figures 1A-1D, the configuration information 148 may be used to install and/or uninstall applications from the computing device 102. For example, the operating system 134 may receive the push notification 210, and determine that the application 146-2 is not installed on the computing device 102. In such an example, the operating system 134 may download and install application 146-2 on the computing device 102, e.g., from an application store, the Internet, etc. Similarly, if the configuration information 148-N specifies that application 146-N is not permitted to be installed on the device and/or was not registered in the configuration information 148, the configuration information 148-N may cause the application 146-N to initiate the removal (and/or disabling) of application 146-N from the computing device 102.

[0080] Advantageously, the configuration information 148 is used to automatically configure and/or restore the computing device 102. By requiring verification of a cryptogram, the configuration of the computing device 102 is completed in a secure manner. More generally, users may quickly restore backups, configure new devices, and/or apply the configuration information 148 from one device to another device. Embodiments are not limited in these contexts.

[0081] Operations for the disclosed embodiments may be further described with reference to the following figures. Some of the figures may include a logic flow. Although such figures presented herein may include a particular logic flow, it can be appreciated that the logic flow merely provides an example of how the general functionality as described herein can be implemented. Further, a given logic flow does not necessarily have to be executed in the order presented unless otherwise indicated. Moreover, not all acts illustrated in a logic flow may be required in some embodiments. In addition, the given logic flow may be implemented by a hardware element, a software element executed by a processor, or any combination thereof. The embodiments are not limited in this context.

[0082] Figure 3 illustrates an embodiment of a logic flow, or routine, 300. The logic flow 300 may be representative of some or all of the operations executed by one or more embodiments

described herein. For example, the logic flow 300 may include some or all of the operations to automatically configure a device using configuration information and a contactless card. The embodiments are not limited in this context.

[0083] In block 302, routine 300 receives, by an account application 136 executing on a computing device 102, a cryptogram (e.g., the cryptogram 122 and/or the cryptogram of the URL with cryptogram 204) from a contactless card 104 associated with an account. In block 304, routine 300 transmits, by the account application 136, the cryptogram to the server 106. In block 306, routine 300 receives, by the account application 136 based on the server verifying the cryptogram, configuration information 148 for a plurality of applications associated with the account. The server 106 may include, with the configuration information 148, indications specifying the server 106 verified the cryptogram 122 and that the configuration information 148 can be used to configure the computing device 102. The applications may include the other applications 146, the operating system 134, and/or the account application 136. In block 308, routine 300 modifies a respective configuration of each application on the computing device 102 based on the configuration information 148.

[0084] Figure 4 illustrates an embodiment of a logic flow, or routine, 400. The logic flow 400 may be representative of some or all of the operations executed by one or more embodiments described herein. For example, the logic flow 400 may include some or all of the operations to automatically configure a device using configuration information and a contactless card. The embodiments are not limited in this context.

[0085] In block 402, routine 400 registers, with a server 106, a plurality of applications and configuration information for each application associated with an account and/or computing device 102. Doing so may cause configuration information 148 to be stored in at least one entry associated with the account in an account database 128. In block 404, routine 400 receives, by a computing device 102, a cryptogram (e.g., the cryptogram 122 and/or the cryptogram of the URL with cryptogram 204) from a contactless card 104 associated with the account. In block 406, routine 400 transmits, by the computing device 102, the cryptogram to the server 106. In block 408, routine 400 determines, by the computing device 102 based on a response received from the server 106, that the server 106 verified the cryptogram and that the configuration information 148 can be used to configure the computing device 102. In block 410, routine 400 receives, by the computing device 102 based on the verification of the cryptogram by the server 106, configuration information 148 for a plurality of applications associated with the account.

The applications may include the other applications 146, the operating system 134, and/or the account application 136.

[0086] In block 412, routine 400 downloads, by the computing device 102, a first application (e.g., one of the other applications 146) to the device based on a determination the first application is not installed on the computing device 102. In block 414, routine 400 installs, by the computing device 102, the first application on the device. In block 416, routine 400 uninstalls a second application (e.g., another one of the other applications 146) from the computing device 102 based on the configuration information 148. For example, the configuration information 148 may specify to uninstall or disable the second application. In block 418, routine 400 modifies, by the computing device 102 a respective configuration of each application based on the configuration information 148.

[0087] Figure 5 illustrates an embodiment of a logic flow, or routine, 500. The logic flow 500 may be representative of some or all of the operations executed by one or more embodiments described herein. For example, the logic flow 500 may include some or all of the operations to automatically configure a device using configuration information and a contactless card. The embodiments are not limited in this context.

[0088] In block 502, routine 500 registers, with a server, a plurality of applications and configuration information for each application with an account. Doing so may cause configuration information 148 to be stored in an entry associated with the account in an account database 128. In block 504, routine 500 receives, by a computing device 102, a URL with cryptogram 204 from a contactless card 104 associated with an account. In block 506, routine 500 launches, by the computing device 102, a web browser based on the URL with cryptogram 204. In block 508, routine 500 transmits, by the web browser 206, at least the cryptogram to the server 106, e.g., by accessing a web page associated with the URL with cryptogram 204. In block 510, routine 500 determines, by the web browser 206 based on a response received from the server, that the server verified the cryptogram from the URL with cryptogram 204 and that the configuration information 148 can be used to configure the computing device 102.

[0089] In block 512, routine 500 receives, by the computing device 102 based on the verification of the cryptogram by the server, configuration information 148 for a plurality of applications associated with the account. The applications may include the other applications 146, the operating system 134, and/or the account application 136. In block 514, routine 500 downloads, by the device, a first application to the computing device 102 based on a

determination the first application is not installed on the computing device 102. In block 516, routine 500 installs, by the computing device 102, the first application on the computing device 102. In block 518, routine 500 uninstalls a second application (e.g., another one of the other applications 146) from the computing device 102 based on the configuration information 148. For example, the configuration information 148 may specify to uninstall or disable the second application. In block 520, routine 500 modifies, by the computing device 102 a respective configuration of each application based on the configuration information 148.

[0090] FIG. 6A is a schematic 600 illustrating an example configuration of a contactless card 104, which may include a payment card, such as a credit card, debit card, or gift card, issued by a service provider as displayed as service provider indicia 602 on the front or back of the contactless card 104. In some examples, the contactless card 104 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 104 may include a substrate 604, 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 104 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 104 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.

[0091] The contactless card 104 may also include identification information 606 displayed on the front and/or back of the card, and a contact pad 608. The contact pad 608 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 104 may also include processing circuitry, antenna and other components as will be further discussed in FIG. 6B. These components may be located behind the contact pad 608 or elsewhere on the substrate 604, e.g. within a different layer of the substrate 604, and may

electrically and physically coupled with the contact pad 608. The contactless card 104 may also include a magnetic strip or tape, which may be located on the back of the card (not shown in FIG. 6A). The contactless card 104 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.

[0092] As illustrated in FIG. 6B, the contact pad 608 of contactless card 104 may include processing circuitry 610 for storing, processing, and communicating information, including a processor 612, a memory 108, and one or more communications interface 124. It is understood that the processing circuitry 610 may contain additional components, including processors, memories, error and parity/CRC checkers, data encoders, anticollision algorithms, controllers, command decoders, security primitives and tamperproofing hardware, as necessary to perform the functions described herein.

[0093] The memory 108 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 104 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-programed 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 108 may be encrypted memory utilizing an encryption algorithm executed by the processor 612 to encrypted data.

[0094] The memory 108 may be configured to store one or more applet 110, one or more counters 116, a unique ID 112, the master key 114, the UDK 118, diversified key 120, the PAN sequence 144, and the configuration information 148. The configuration information 148 may include one or more tokens generated by the server 106. The one or more applets 110 may comprise one or more software applications configured to execute on one or more contactless cards 104, such as a Java® Card applet. However, it is understood that applets 110 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 counters 116 may comprise a numeric counter sufficient to store an integer. The unique ID 112 may comprise a unique alphanumeric identifier assigned to the contactless card 104, and the identifier may

distinguish the contactless card 104 from other contactless cards 104. In some examples, the unique ID 112 may identify both a customer and an account assigned to that customer.

[0095] The processor 612 and memory elements of the foregoing exemplary embodiments are described with reference to the contact pad 608, but the present disclosure is not limited thereto. It is understood that these elements may be implemented outside of the contact pad 608 or entirely separate from it, or as further elements in addition to processor 612 and memory 108 elements located within the contact pad 608.

[0096] In some examples, the contactless card 104 may comprise one or more antenna(s) 614. The one or more antenna(s) 614 may be placed within the contactless card 104 and around the processing circuitry 610 of the contact pad 608. For example, the one or more antenna(s) 614 may be integral with the processing circuitry 610 and the one or more antenna(s) 614 may be used with an external booster coil. As another example, the one or more antenna(s) 614 may be external to the contact pad 608 and the processing circuitry 610.

[0097] In an embodiment, the coil of contactless card 104 may act as the secondary of an air core transformer. The terminal may communicate with the contactless card 104 by cutting power or amplitude modulation. The contactless card 104 may infer the data transmitted from the terminal using the gaps in the power connection of the contactless card 104, which may be functionally maintained through one or more capacitors. The contactless card 104 may communicate back by switching a load on the coil of the contactless card 104 or load modulation. Load modulation may be detected in the terminal's coil through interference. More generally, using the antenna(s) 614, processor 612, and/or the memory 108, the contactless card 104 provides a communications interface to communicate via NFC, Bluetooth, and/or Wi-Fi communications.

[0098] As explained above, contactless card 104 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 110 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 110 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 computing device 102 or point-of-sale terminal), and produce an NDEF message that comprises a cryptographically secure OTP encoded as an NDEF text tag. The

NDEF message may include a cryptogram such as the cryptogram 122 or URL with cryptogram 204, and any other data.

[0099] One example of an NDEF OTP is an NDEF short-record layout (SR=1). In such an example, one or more applet 110 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 110 may be configured to add one or more static tag records in addition to the OTP record.

[0100] In some examples, the one or more applet 110 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 110, 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.

[0101] In some examples, the contactless card 104 and server may include certain data such that the card may be properly identified. The contactless card 104 may include one or more unique identifiers (not pictured). Each time a read operation takes place, the counter 116 may be configured to increment. In some examples, each time data from the contactless card 104 is read (e.g., by a mobile device), the counter 116 is transmitted to the server for validation and determines whether the counter 116 are equal (as part of the validation) to a counter of the server.

[0102] The one or more counter 116 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 116 has been read or used or otherwise passed over. If the counter 116 has not been used, it may be replayed. In some examples, the counter that is incremented on the contactless card 104 is different from the counter that is incremented for transactions. The contactless card 104 is unable to determine the application transaction counter 116 since there is no communication between applets 110 on the contactless card 104. In some examples, the contactless card 104 may comprise a first applet 440-1, which may be a transaction applet, and a second applet 440-2. Each applet 440-1 and 440-2 may comprise a respective counter 116.

[0103] In some examples, the counter 116 may get out of sync. In some examples, to account for accidental reads that initiate transactions, such as reading at an angle, the counter 116 may increment but the application does not process the counter 116. In some examples, when the

mobile device 10 is woken up, NFC may be enabled and the computing device 102 may be configured to read available tags, but no action is taken responsive to the reads.

[0104] To keep the counter 116 in sync, an application, such as a background application, may be executed that would be configured to detect when the computing device 102 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 116 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 116 may be configured to move forward. But if within a different threshold number, for example within 10 or 1000, 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 116 increases in the appropriate sequence, then it possible to know that the user has done so.

[0105] The key diversification technique described herein with reference to the counter 116, master key 114, UDK 118, and diversified key 120, 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.

[0106] During the creation process of the contactless card 104, 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 104. 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.

[0107] 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 (e.g., the UDKs 118) and the counter may be used as diversification data. For example, each time the contactless card 104 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).

[0108] 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.

[0109] 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.

[0110] FIG. 7 illustrates an NDEF short-record layout (SR=1) data structure 700 according to an example embodiment. One or more applets 110 may be configured to encode an OTP as an NDEF type 4 well known type text tag. In some examples, NDEF messages may comprise one or more records. The applets may be configured to add one or more static tag records in addition to the OTP record. Exemplary tags include, without limitation, Tag type: well known type, text, encoding English (en); Applet ID: D2760000850101; Capabilities: read-only access; Encoding: the authentication message may be encoded as ASCII hex; type-length-value (TLV) data may be provided as a personalization parameter that may be used to generate the NDEF message. In an embodiment, the authentication template may comprise the first record, with a well-known index for providing the actual dynamic authentication data. The data structure 700 may include a cryptogram such as cryptogram 122 or URL with cryptogram 204, and any other data provided by the applet 110.

[0111] FIG. 8 illustrates an embodiment of an exemplary computer architecture 800 suitable for implementing various embodiments as previously described. In one embodiment, the computer architecture 800 may include or be implemented as part of computing architecture 100 and/or 200. In some embodiments, computing system 800 may be representative, for example, of the contactless card 104, computing devices 102, and/or server 106 The embodiments are not limited in this context. More generally, the computing architecture 800 is configured to implement all logic, applications, systems, methods, apparatuses, and functionality described herein with reference to Figures 1A-7.

[0112] 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 800. 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.

[0113] The computer architecture 800 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 800.

[0114] As shown in FIG. 8, the computer architecture 800 includes a computer 812 comprising a processor 802, a system memory 804 and a system bus 806. The processor 802 can be any of various commercially available processors. The computer 812 may be representative of the computing device 102 and/or the server 106.

[0115] The system bus 806 provides an interface for system components including, but not limited to, the system memory 804 to the processor 802. The system bus 806 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 806 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.

[0116] The computer architecture 800 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-writeable 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.

[0117] The system memory 804 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. 8, the system memory 804 can include non-volatile 808 and/or volatile 810. A basic input/output system (BIOS) can be stored in the non-volatile 808.

[0118] The computer 812 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 814, a magnetic disk drive 816 to read from or write to a removable magnetic disk 818, and an optical disk drive 820 to read from or write to a removable optical disk 822 (e.g., a CD-ROM or DVD). The hard disk drive 814, magnetic disk drive 816 and optical disk drive 820

can be connected to the system bus 806 by an HDD interface 824, and FDD interface 826 and an optical disk drive interface 828, respectively. The HDD interface 824 for external drive implementations can include at least one or both of Universal Serial Bus (USB) and IEEE 1394 interface technologies.

[0119] 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 808, and volatile 810, including an operating system 830, one or more applications 832, other program modules 834, and program data 836. In one embodiment, the one or more applications 832, other program modules 834, and program data 836 can include, for example, the various applications and/or components of the system 100.

[0120] A user can enter commands and information into the computer 812 through one or more wire/wireless input devices, for example, a keyboard 838 and a pointing device, such as a mouse 840. 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 802 through an input device interface 842 that is coupled to the system bus 806 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.

[0121] A monitor 844 or other type of display device is also connected to the system bus 806 via an interface, such as a video adapter 846. The monitor 844 may be internal or external to the computer 812. In addition to the monitor 844, a computer typically includes other peripheral output devices, such as speakers, printers, and so forth.

[0122] The computer 812 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) 848. The remote computer(s) 848 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 812, although, for purposes of brevity, only a memory and/or storage device 850 is illustrated. The logical connections depicted include wire/wireless connectivity to a local area network 852 and/or larger networks, for example, a wide area

network 854. 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.

[0123] When used in a local area network 852 networking environment, the computer 812 is connected to the local area network 852 through a wire and/or wireless communication network interface or network adapter 856. The network adapter 856 can facilitate wire and/or wireless communications to the local area network 852, which may also include a wireless access point disposed thereon for communicating with the wireless functionality of the network adapter 856.

[0124] When used in a wide area network 854 networking environment, the computer 812 can include a modem 858, or is connected to a communications server on the wide area network 854 or has other means for establishing communications over the wide area network 854, such as by way of the Internet. The modem 858, which can be internal or external and a wire and/or wireless device, connects to the system bus 806 via the input device interface 842. In a networked environment, program modules depicted relative to the computer 812, or portions thereof, can be stored in the remote memory and/or storage device 850. 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.

[0125] The computer 812 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, ac, ax, 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).

[0126] The various elements of the devices as previously described with reference to FIGS. 1A-12 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.

[0127] One or more aspects of at least one embodiment may be implemented by representative instructions stored on a machine-readable medium which represents various logic within the processor, which when read by a machine causes the machine to fabricate logic to perform the techniques described herein. Such representations, known as "IP cores" may be stored on a tangible, machine readable medium and supplied to various customers or manufacturing facilities to load into the fabrication machines that make the logic or processor. Some embodiments may be implemented, for example, using a machine-readable medium or article which may store an instruction or a set of instructions that, if executed by a machine, may cause the machine to perform a method and/or operations in accordance with the embodiments. Such a machine may include, for example, any suitable processing platform, computing platform, computing device, processing device, computing system, processing system, computer, processor, or the like, and may be implemented using any suitable combination of hardware and/or software. The machine-readable medium or article may include, for example, any suitable type of memory unit, memory device, memory article, memory medium, storage device, storage article, storage medium and/or storage unit, for example, memory, removable or non-removable media, erasable or non-erasable media, writeable or re-writeable media, digital or analog media, hard disk, floppy disk, Compact Disk Read Only Memory (CD-ROM), Compact Disk Recordable (CD-R), Compact Disk Rewriteable (CD-RW), optical disk, magnetic media, magneto-optical media, removable memory cards or disks, various types of Digital Versatile Disk (DVD), a tape, a cassette, or the like. The instructions may include any

suitable type of code, such as source code, compiled code, interpreted code, executable code, static code, dynamic code, encrypted code, and the like, implemented using any suitable high-level, low-level, object-oriented, visual, compiled and/or interpreted programming language.

[0128] 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."

[0129] It will be appreciated that the exemplary devices shown in the block diagrams described above may represent one functionally descriptive example of many potential implementations. Accordingly, division, omission or inclusion of block functions depicted in the accompanying figures does not infer that the hardware components, circuits, software and/or elements for implementing these functions would be necessarily be divided, omitted, or included in embodiments.

[0130] At least one computer-readable storage medium may include instructions that, when executed, cause a system to perform any of the computer-implemented methods described herein.

[0131] Some embodiments may be described using the expression "one embodiment" or "an embodiment" along with their derivatives. These terms mean that a particular feature, structure, or characteristic described in connection with the embodiment is included in at least one embodiment. The appearances of the phrase "in one embodiment" in various places in the specification are not necessarily all referring to the same embodiment. Moreover, unless otherwise noted the features described above are recognized to be usable together in any combination. Thus, any features discussed separately may be employed in combination with each other unless it is noted that the features are incompatible with each other.

[0132] It is emphasized that the Abstract of the Disclosure is provided to allow a reader to quickly ascertain the nature of the technical disclosure. It is submitted with the understanding that it will not be used to interpret or limit the scope or meaning of the claims. In addition, in the foregoing Detailed Description, it can be seen that various features are grouped together in a single embodiment for the purpose of streamlining the disclosure. This method of disclosure

is not to be interpreted as reflecting an intention that the claimed embodiments require more features than are expressly recited in each claim. Rather, as the following claims reflect, inventive subject matter lies in less than all features of a single disclosed embodiment. Thus the following claims are hereby incorporated into the Detailed Description, with each claim standing on its own as a separate embodiment. In the appended claims, the terms "including" and "in which" are used as the plain-English equivalents of the respective terms "comprising" and "wherein," respectively. Moreover, the terms "first," "second," "third," and so forth, are used merely as labels, and are not intended to impose numerical requirements on their objects.

[0133] What has been described above includes examples of the disclosed architecture. It is, of course, not possible to describe every conceivable combination of components and/or methodologies, but one of ordinary skill in the art may recognize that many further combinations and permutations are possible. Accordingly, the novel architecture is intended to embrace all such alterations, modifications and variations that fall within the spirit and scope of the appended claims.

[0134] The foregoing description of example embodiments has been presented for the purposes of illustration and description. It is not intended to be exhaustive or to limit the present disclosure to the precise forms disclosed. Many modifications and variations are possible in light of this disclosure. It is intended that the scope of the present disclosure be limited not by this detailed description, but rather by the claims appended hereto. Future filed applications claiming priority to this application may claim the disclosed subject matter in a different manner, and may generally include any set of one or more limitations as variously disclosed or otherwise demonstrated herein.

CLAIMS

What is claimed is:

1. A method, comprising:

receiving, by an application executing on a device, a cryptogram from a contactless card associated with an account;

transmitting, by the application, the cryptogram to a server;

receiving, by the application based on the cryptogram, configuration information for a plurality of applications associated with the account; and

modifying a respective configuration of each application on the device based on the configuration information.

2. The method of claim 1, further comprising, prior to modifying the configuration of each application:

determining, by the application, that a first application of the plurality of applications is not installed on the device; and

downloading, by the application, the first application to the device; and

installing, by an operating system (OS) of the device, the first application on the device.

3. The method of claim 2, further comprising subsequent to installing the first application on the device:

modifying the configuration of the first application based on the configuration information of the first application.

4. The method of claim 1, wherein receiving the configuration information comprises receiving a plurality of configuration files from the server.

5. The method of claim 1, wherein receiving the configuration information comprises:

receiving, by each application, a respective push notification from the server, the respective push notification comprising the respective configuration information for the respective application, wherein the push notifications comprise background notifications.

6. The method of claim 1, wherein receiving the configuration information comprises:

receiving, by each application, a respective push notification from the server, the respective push notification comprising a respective token, wherein the push notifications comprise background notifications;

transmitting, by the respective application, the respective token to the server; and

receiving, by the respective application, the respective configuration information based on the respective token.

7. The method of claim 1, wherein the configuration information comprises: (i) information for connecting to one or more servers, (ii) account information, (iii) messages, (iv) preferences, and (v) settings.

8. A non-transitory computer-readable storage medium, the computer-readable storage medium including instructions that when executed by a processor of a device, cause the processor to:

receive, by an application, a cryptogram from a contactless card associated with an account;

transmit, by the application, the cryptogram to a server;

receive, by the application based on the cryptogram, configuration information for a plurality of applications associated with the account; and

modify a respective configuration of each application on the device based on the configuration information.

9. The computer-readable storage medium of claim 8, wherein the instructions further configure the processor to, prior to modifying the configuration of each application:

determine, by the application, that a first application of the plurality of applications is not installed on the device; and

download, by the application, the first application to the device; and

install, by an operating system (OS) of the device, the first application on the device.

10. The computer-readable storage medium of claim 9, wherein the instructions further configure the processor to, subsequent to installing the first application on the device:

modify the configuration of the first application based on the configuration information of the first application.

11. The computer-readable storage medium of claim 8, wherein receiving the configuration information comprises receiving a plurality of configuration files from the server.

12. The computer-readable storage medium of claim 8, wherein receiving the configuration information comprises:

receiving, by each application, a respective push notification from the server, the respective push notification comprising the respective configuration information for the respective application, wherein the push notifications comprise background notifications.

13. The computer-readable storage medium of claim 8, wherein receiving the configuration information comprises:

receiving, by each application, a respective push notification from the server, the respective push notification comprising a respective token, wherein the push notifications comprise background notifications;

transmitting, by the respective application, the respective token to the server; and

receiving, by the respective application, the respective configuration information based on the respective token.

14. The computer-readable storage medium of claim 8, wherein the configuration information comprises: (i) information for connecting to one or more servers, (ii) account information, (iii) messages, (iv) preferences, and (v) settings.

15. A computing apparatus comprising:

a processor; and

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

receive, by an application, a cryptogram from a contactless card associated with an account;

transmit, by the application, the cryptogram to a server;

receive, by the application based on the cryptogram, configuration information for a plurality of applications associated with the account; and

modify a respective configuration of each application on the apparatus based on the configuration information.

16. The computing apparatus of claim 15, wherein the instructions further cause the processor to, prior to modifying the configuration of each application:

determine, by the application, that a first application of the plurality of applications is not installed on the apparatus; and

download, by the application, the first application to the apparatus; and

install, by an operating system (OS) of the apparatus, the first application on the apparatus.

17. The computing apparatus of claim 16, wherein the instructions further configure the apparatus to subsequent to installing the first application:

modify the configuration of the first application based on the configuration information of the first application.

18. The computing apparatus of claim 15, wherein receiving the configuration information comprises receiving a plurality of configuration files from the server.

19. The computing apparatus of claim 15, wherein receiving the configuration information comprises:

receiving, by each application, a respective push notification from the server, the respective push notification comprising the respective configuration information for the respective application, wherein the push notifications comprise background notifications.

20. The computing apparatus of claim 15, wherein receiving the configuration information comprises:

receiving, by each application, a respective push notification from the server, the respective push notification comprising a respective token, wherein the push notifications comprise background notifications;

transmitting, by the respective application, the respective token to the server; and

receiving, by the respective application, the respective configuration information based on the respective token.

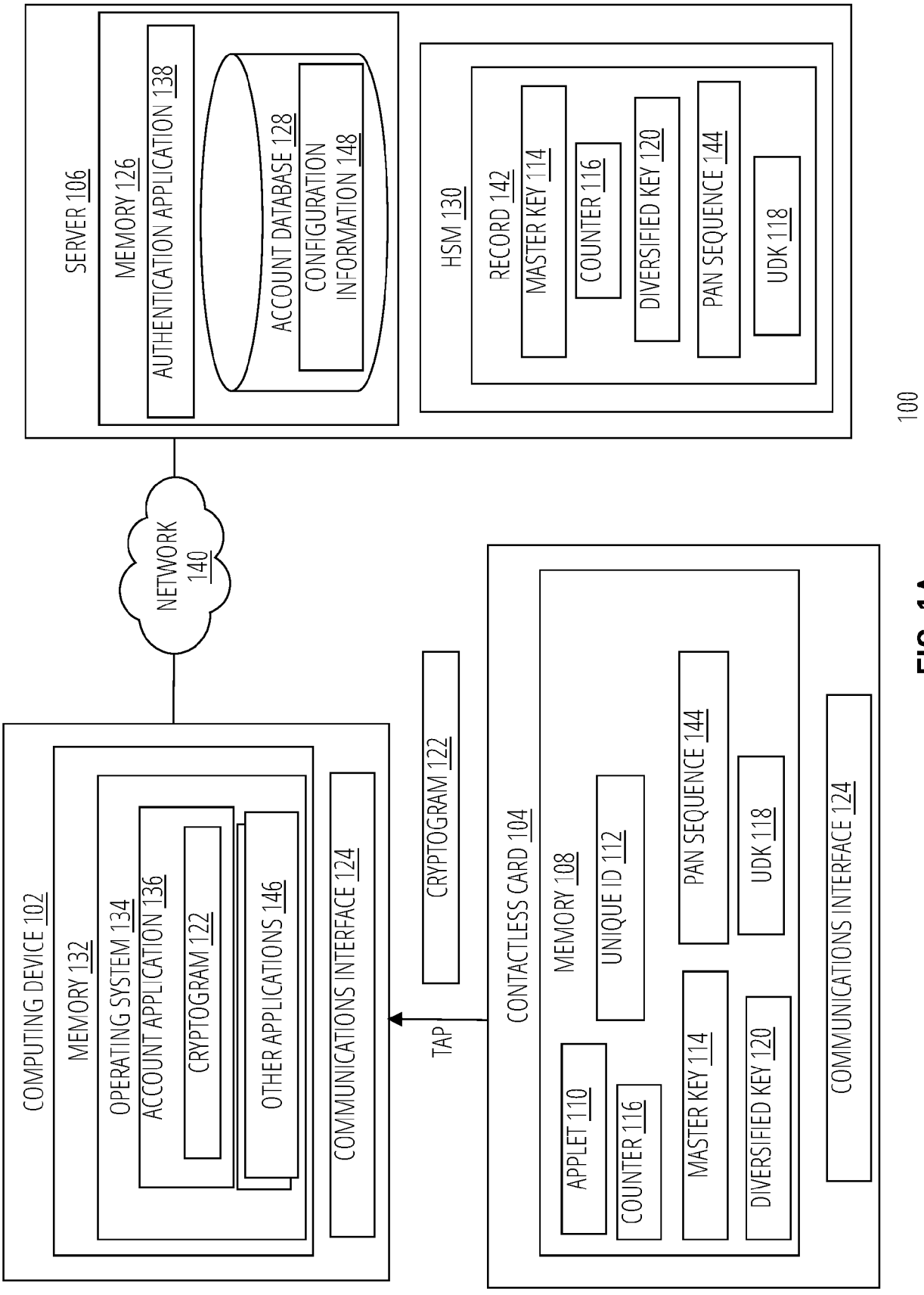

SERVER 106
MEMORY 126
AUTHENTICATION APPLICATION 138
ACCOUNT DATABASE 128
CONFIGURATION INFORMATION 148
HSM 130
RECORD 142
MASTER KEY 114
COUNTER 116
DIVERSIFIED KEY 120
PAN SEQUENCE 144
UDK 118
100
NETWORK 140
COMPUTING DEVICE 102
MEMORY 132
OPERATING SYSTEM 134
ACCOUNT APPLICATION 136
CRYPTOGRAM 122
OTHER APPLICATIONS 146
COMMUNICATIONS INTERFACE 124
CRYPTOGRAM 122
TAP
CONTACTLESS CARD 104
MEMORY 108
APPLET 110
COUNTER 116
UNIQUE ID 112
MASTER KEY 114
DIVERSIFIED KEY 120
PAN SEQUENCE 144
UDK 118
COMMUNICATIONS INTERFACE 124

FIG. 1A

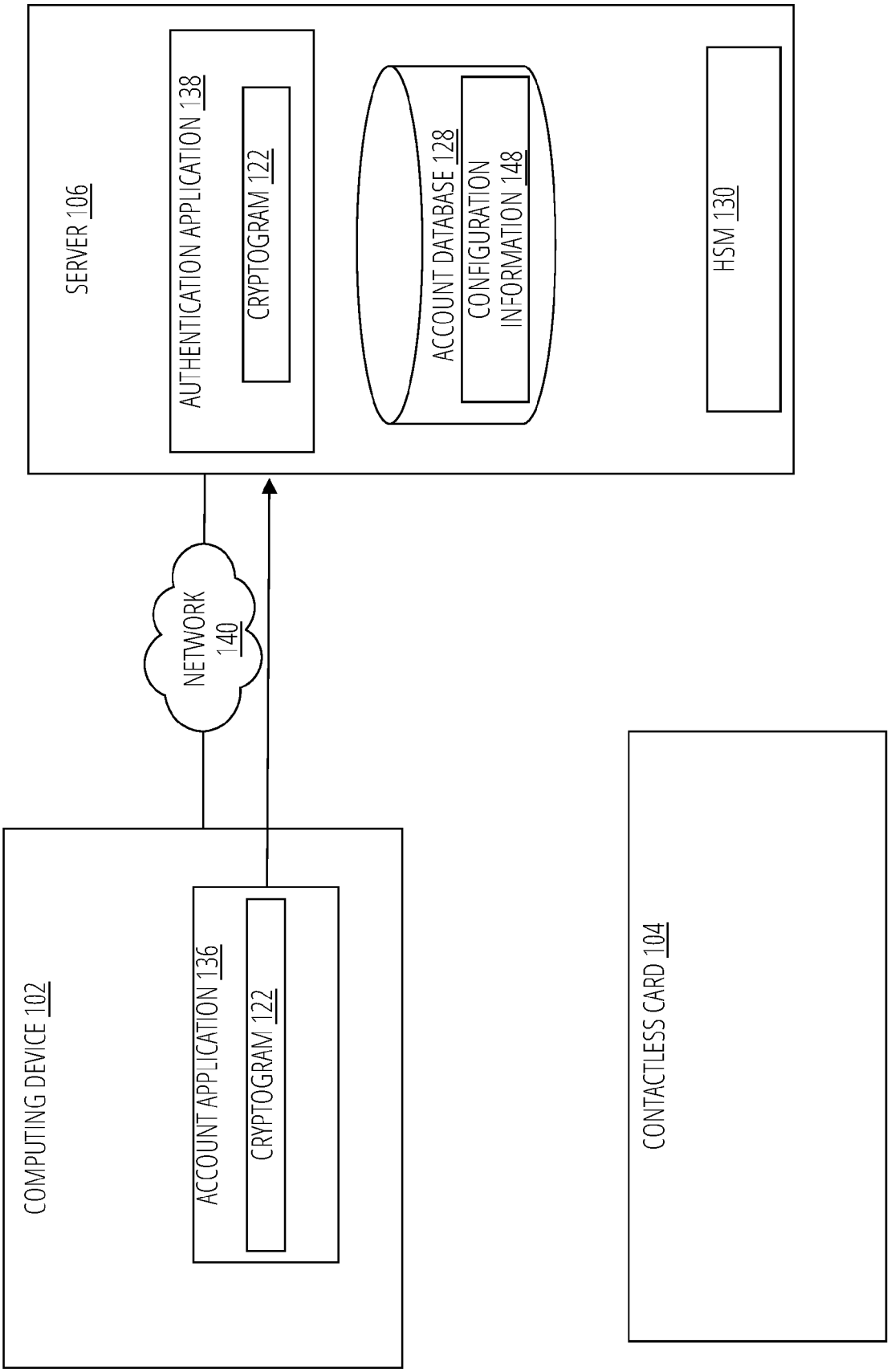

SERVER 106
AUTHENTICATION APPLICATION 138
CRYPTOGRAM 122
ACCOUNT DATABASE 128
CONFIGURATION INFORMATION 148
HSM 130
NETWORK 140
COMPUTING DEVICE 102
ACCOUNT APPLICATION 136
CRYPTOGRAM 122
CONTACTLESS CARD 104

FIG. 1B

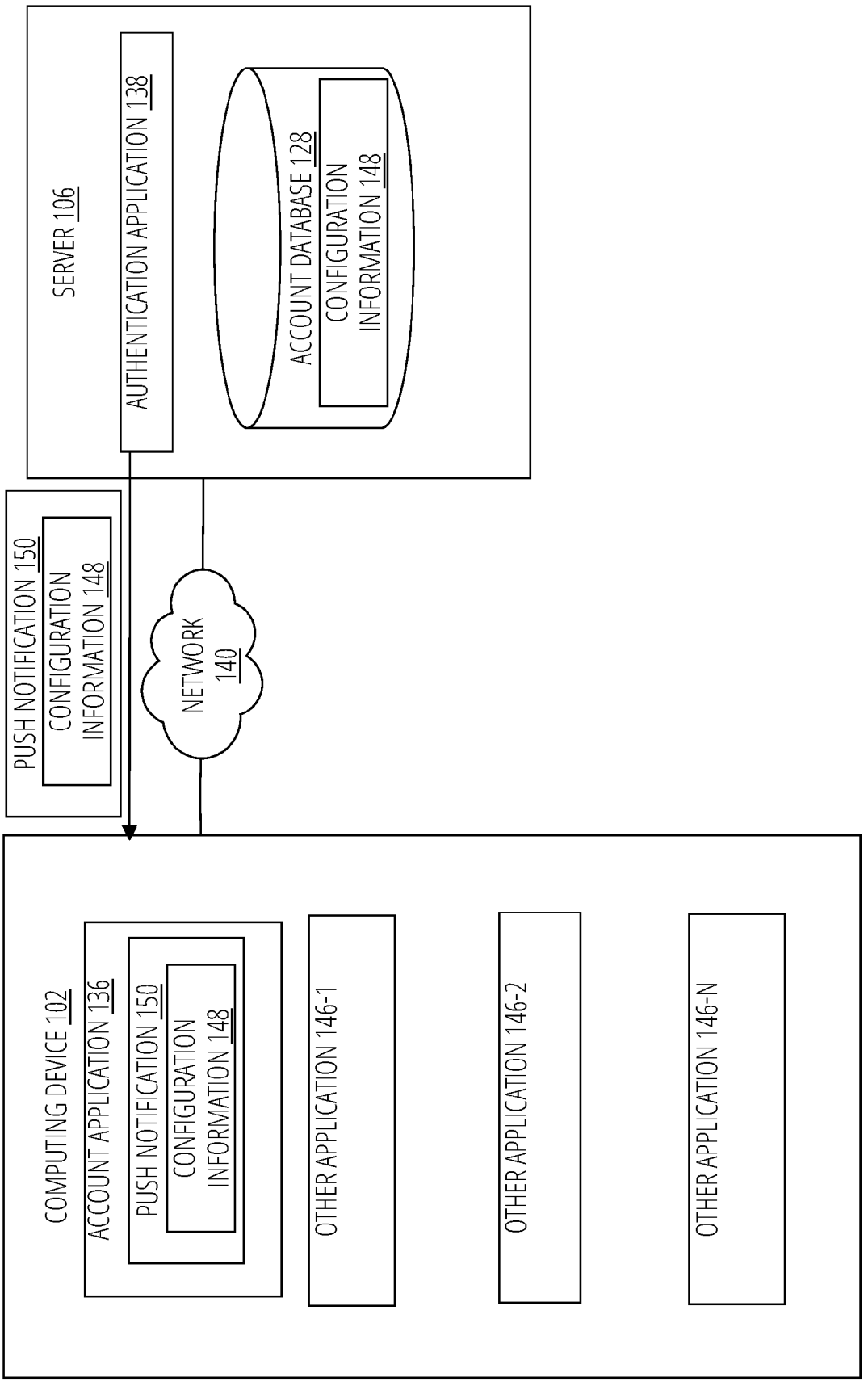

SERVER 106
AUTHENTICATION APPLICATION 138
ACCOUNT DATABASE 128
CONFIGURATION INFORMATION 148
PUSH NOTIFICATION 150
CONFIGURATION INFORMATION 148
NETWORK 140
COMPUTING DEVICE 102
ACCOUNT APPLICATION 136
PUSH NOTIFICATION 150
CONFIGURATION INFORMATION 148
OTHER APPLICATION 146-1
OTHER APPLICATION 146-2
OTHER APPLICATION 146-N

FIG. 1C

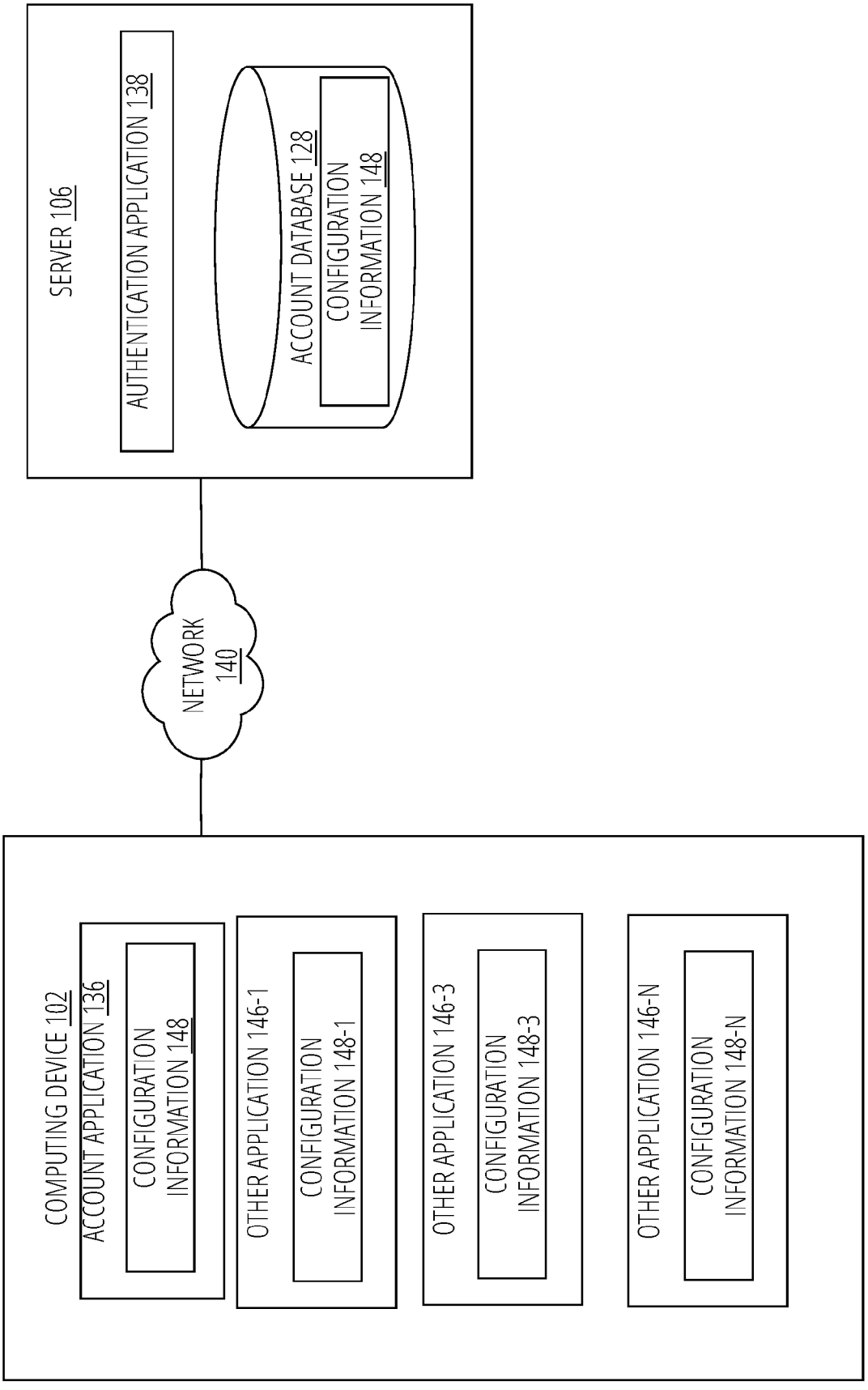

SERVER 106
AUTHENTICATION APPLICATION 138
ACCOUNT DATABASE 128
CONFIGURATION INFORMATION 148
NETWORK 140
COMPUTING DEVICE 102
ACCOUNT APPLICATION 136
CONFIGURATION INFORMATION 148
OTHER APPLICATION 146-1
CONFIGURATION INFORMATION 148-1
OTHER APPLICATION 146-3
CONFIGURATION INFORMATION 148-3
OTHER APPLICATION 146-N
CONFIGURATION INFORMATION 148-N

FIG. 1D

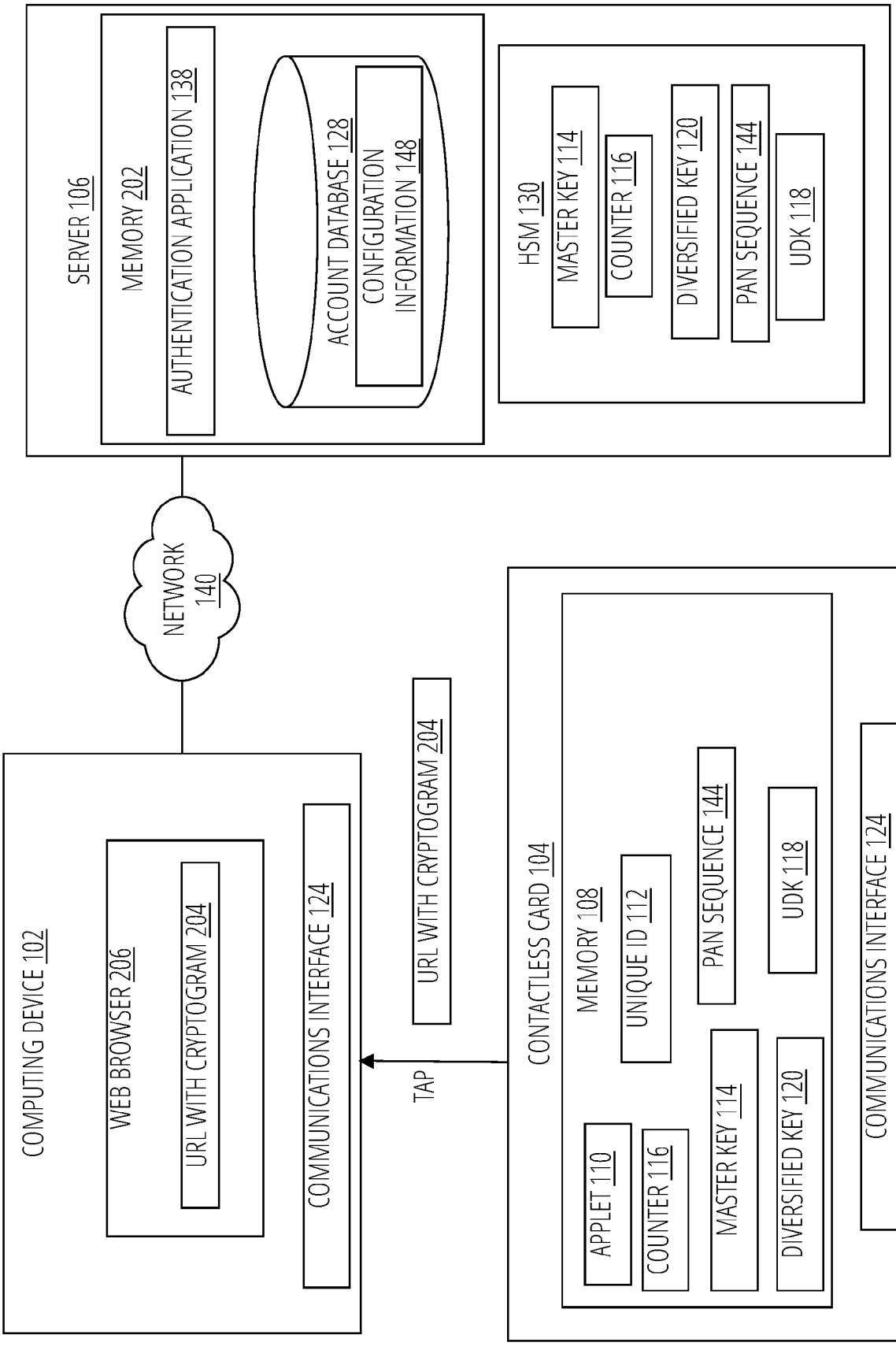

SERVER 106
MEMORY 202
AUTHENTICATION APPLICATION 138
ACCOUNT DATABASE 128
CONFIGURATION INFORMATION 148
HSM 130
MASTER KEY 114
COUNTER 116
DIVERSIFIED KEY 120
PAN SEQUENCE 144
UDK 118
200
NETWORK 140
COMPUTING DEVICE 102
WEB BROWSER 206
URL WITH CRYPTOGRAM 204
COMMUNICATIONS INTERFACE 124
URL WITH CRYPTOGRAM 204
TAP
CONTACTLESS CARD 104
MEMORY 108
APPLET 110
UNIQUE ID 112
COUNTER 116
MASTER KEY 114
PAN SEQUENCE 144
DIVERSIFIED KEY 120
UDK 118
COMMUNICATIONS INTERFACE 124

FIG. 2A

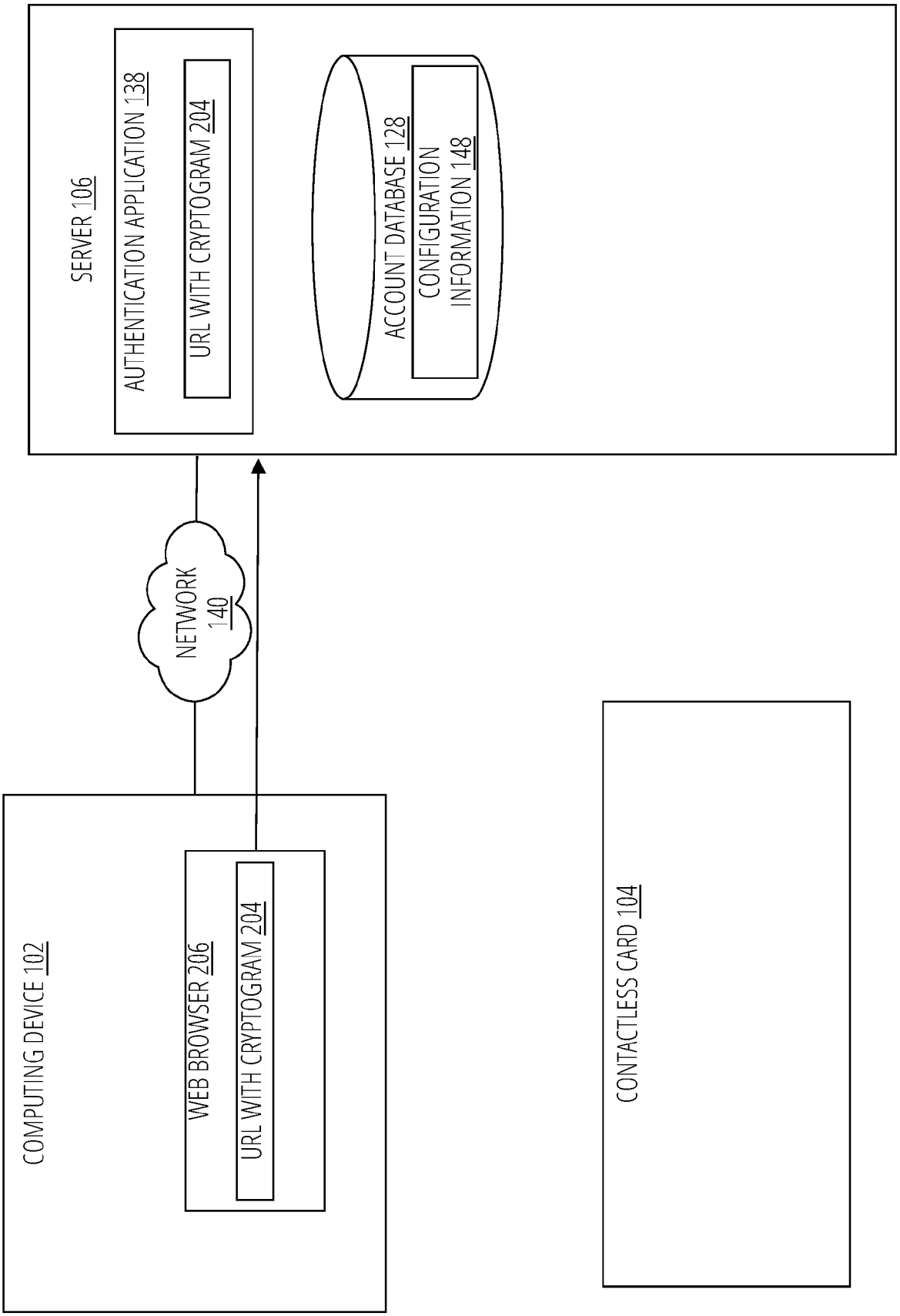

SERVER 106
AUTHENTICATION APPLICATION 138
URL WITH CRYPTOGRAM 204
ACCOUNT DATABASE 128
CONFIGURATION INFORMATION 148
NETWORK 140
COMPUTING DEVICE 102
WEB BROWSER 206
URL WITH CRYPTOGRAM 204
CONTACTLESS CARD 104

FIG. 2B

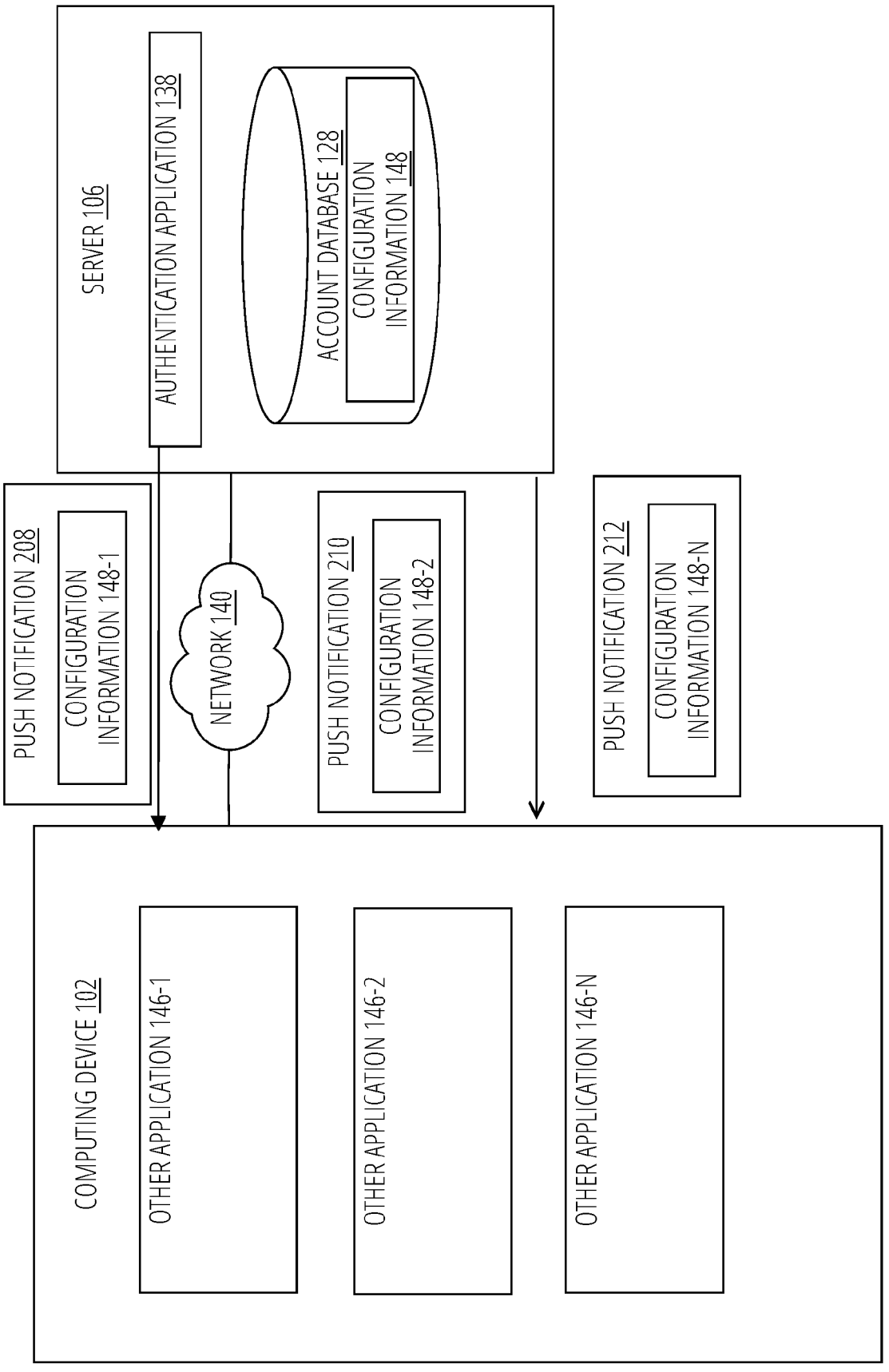

SERVER 106
AUTHENTICATION APPLICATION 138
ACCOUNT DATABASE 128
CONFIGURATION INFORMATION 148
PUSH NOTIFICATION 208
CONFIGURATION INFORMATION 148-1
NETWORK 140
PUSH NOTIFICATION 210
CONFIGURATION INFORMATION 148-2
PUSH NOTIFICATION 212
CONFIGURATION INFORMATION 148-N
COMPUTING DEVICE 102
OTHER APPLICATION 146-1
OTHER APPLICATION 146-2
OTHER APPLICATION 146-N

FIG. 2C

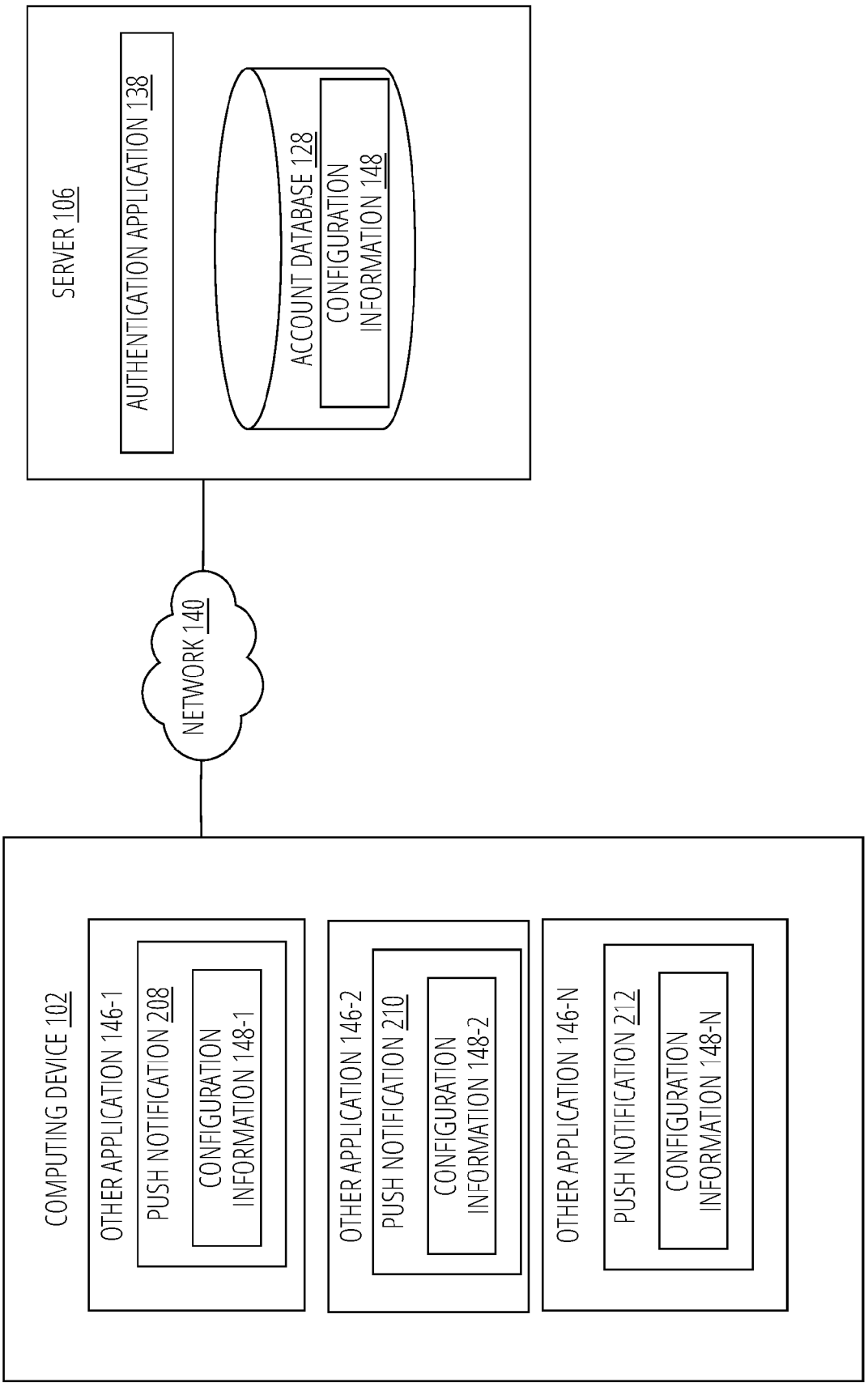

SERVER 106
AUTHENTICATION APPLICATION 138
ACCOUNT DATABASE 128
CONFIGURATION INFORMATION 148
NETWORK 140
COMPUTING DEVICE 102
OTHER APPLICATION 146-1
PUSH NOTIFICATION 208
CONFIGURATION INFORMATION 148-1
OTHER APPLICATION 146-2
PUSH NOTIFICATION 210
CONFIGURATION INFORMATION 148-2
OTHER APPLICATION 146-N
PUSH NOTIFICATION 212
CONFIGURATION INFORMATION 148-N

FIG. 2D

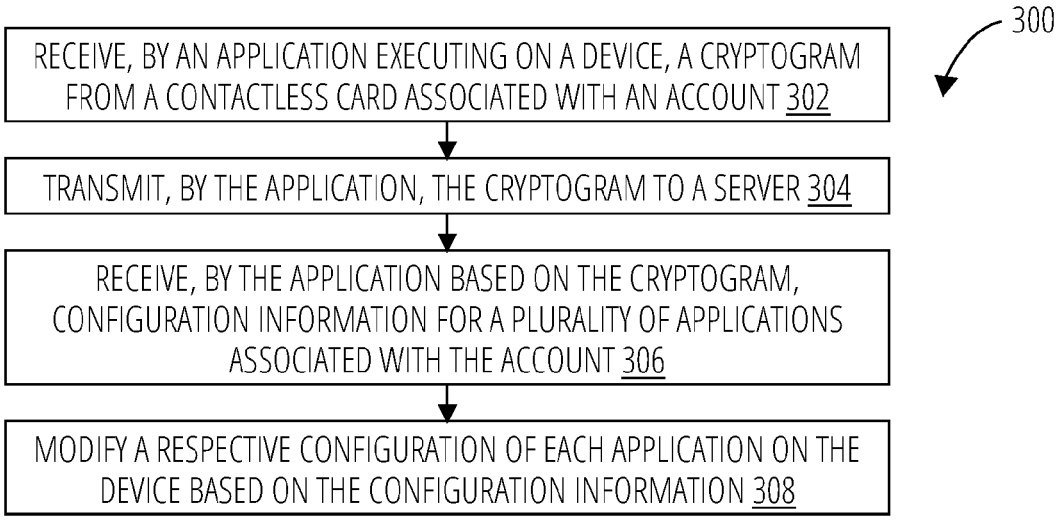

300
RECEIVE, BY AN APPLICATION EXECUTING ON A DEVICE, A CRYPTOGRAM FROM A CONTACTLESS CARD ASSOCIATED WITH AN ACCOUNT 302
TRANSMIT, BY THE APPLICATION, THE CRYPTOGRAM TO A SERVER 304
RECEIVE, BY THE APPLICATION BASED ON THE CRYPTOGRAM, CONFIGURATION INFORMATION FOR A PLURALITY OF APPLICATIONS ASSOCIATED WITH THE ACCOUNT 306
MODIFY A RESPECTIVE CONFIGURATION OF EACH APPLICATION ON THE DEVICE BASED ON THE CONFIGURATION INFORMATION 308

FIG. 3

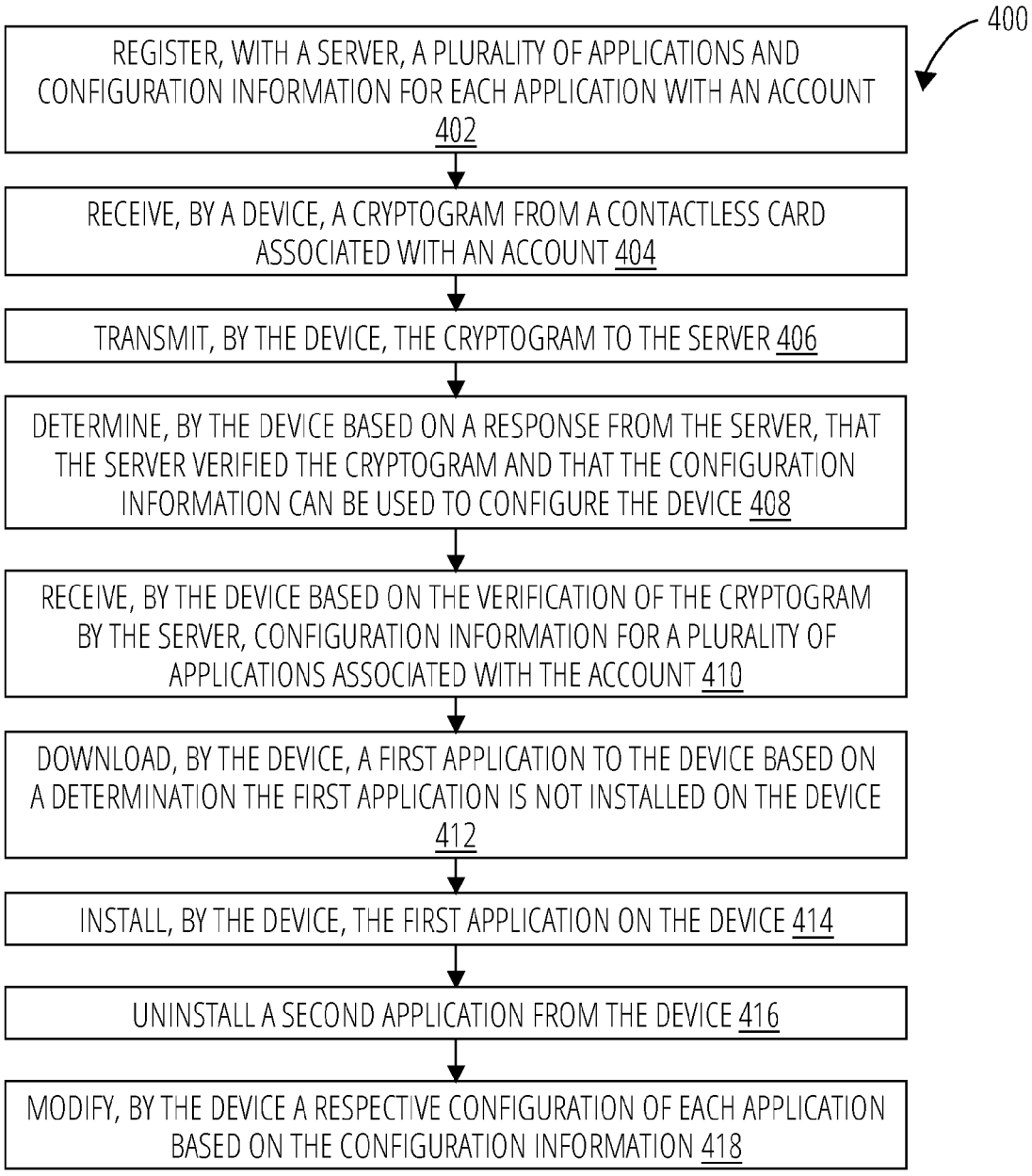

400
REGISTER, WITH A SERVER, A PLURALITY OF APPLICATIONS AND CONFIGURATION INFORMATION FOR EACH APPLICATION WITH AN ACCOUNT 402
RECEIVE, BY A DEVICE, A CRYPTOGRAM FROM A CONTACTLESS CARD ASSOCIATED WITH AN ACCOUNT 404
TRANSMIT, BY THE DEVICE, THE CRYPTOGRAM TO THE SERVER 406
DETERMINE, BY THE DEVICE BASED ON A RESPONSE FROM THE SERVER, THAT THE SERVER VERIFIED THE CRYPTOGRAM AND THAT THE CONFIGURATION INFORMATION CAN BE USED TO CONFIGURE THE DEVICE 408
RECEIVE, BY THE DEVICE BASED ON THE VERIFICATION OF THE CRYPTOGRAM BY THE SERVER, CONFIGURATION INFORMATION FOR A PLURALITY OF APPLICATIONS ASSOCIATED WITH THE ACCOUNT 410
DOWNLOAD, BY THE DEVICE, A FIRST APPLICATION TO THE DEVICE BASED ON A DETERMINATION THE FIRST APPLICATION IS NOT INSTALLED ON THE DEVICE 412
INSTALL, BY THE DEVICE, THE FIRST APPLICATION ON THE DEVICE 414
UNINSTALL A SECOND APPLICATION FROM THE DEVICE 416
MODIFY, BY THE DEVICE A RESPECTIVE CONFIGURATION OF EACH APPLICATION BASED ON THE CONFIGURATION INFORMATION 418

FIG. 4

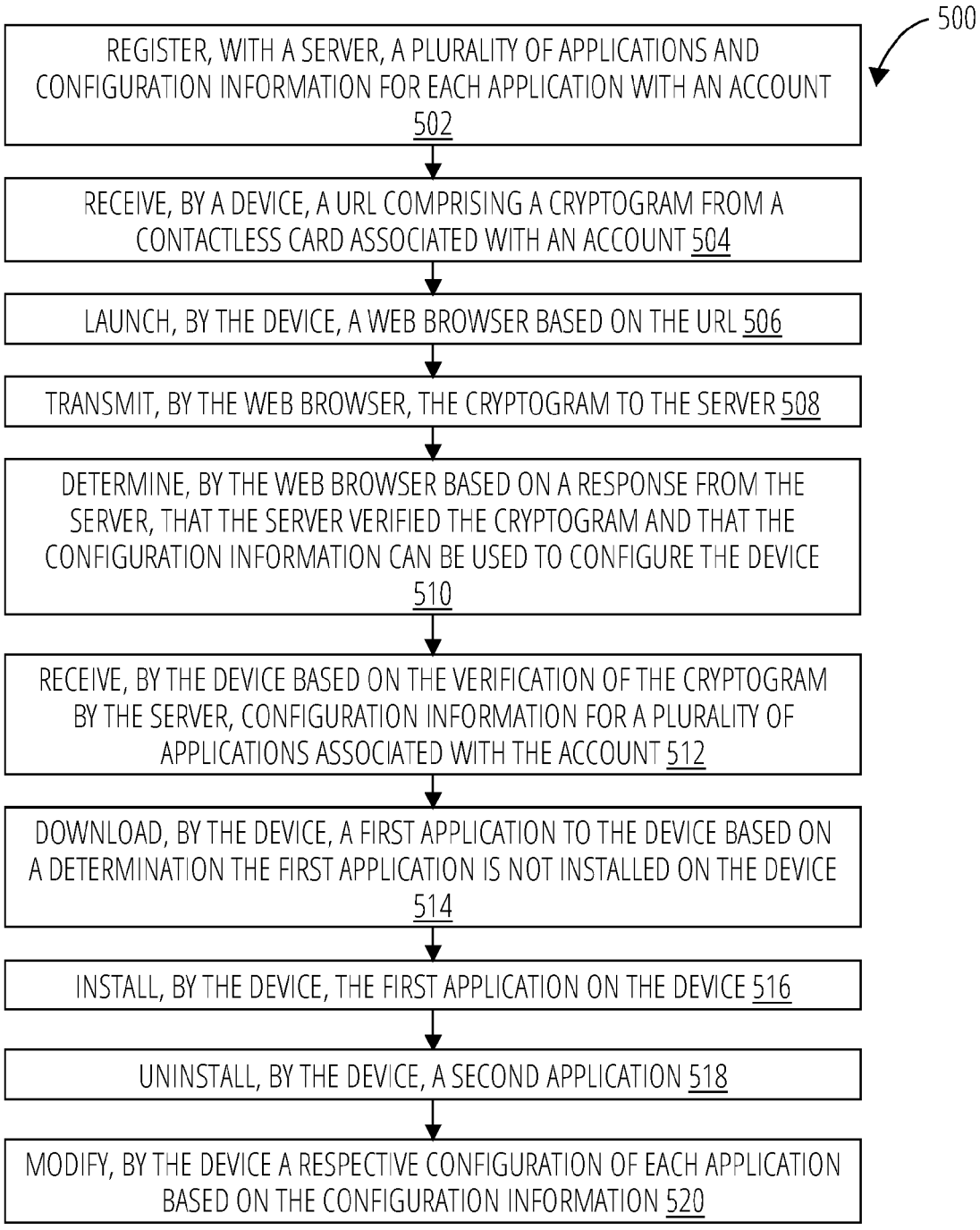

500
REGISTER, WITH A SERVER, A PLURALITY OF APPLICATIONS AND CONFIGURATION INFORMATION FOR EACH APPLICATION WITH AN ACCOUNT 502
RECEIVE, BY A DEVICE, A URL COMPRISING A CRYPTOGRAM FROM A CONTACTLESS CARD ASSOCIATED WITH AN ACCOUNT 504
LAUNCH, BY THE DEVICE, A WEB BROWSER BASED ON THE URL 506
TRANSMIT, BY THE WEB BROWSER, THE CRYPTOGRAM TO THE SERVER 508
DETERMINE, BY THE WEB BROWSER BASED ON A RESPONSE FROM THE SERVER, THAT THE SERVER VERIFIED THE CRYPTOGRAM AND THAT THE CONFIGURATION INFORMATION CAN BE USED TO CONFIGURE THE DEVICE 510
RECEIVE, BY THE DEVICE BASED ON THE VERIFICATION OF THE CRYPTOGRAM BY THE SERVER, CONFIGURATION INFORMATION FOR A PLURALITY OF APPLICATIONS ASSOCIATED WITH THE ACCOUNT 512
DOWNLOAD, BY THE DEVICE, A FIRST APPLICATION TO THE DEVICE BASED ON A DETERMINATION THE FIRST APPLICATION IS NOT INSTALLED ON THE DEVICE 514
INSTALL, BY THE DEVICE, THE FIRST APPLICATION ON THE DEVICE 516
UNINSTALL, BY THE DEVICE, A SECOND APPLICATION 518
MODIFY, BY THE DEVICE A RESPECTIVE CONFIGURATION OF EACH APPLICATION BASED ON THE CONFIGURATION INFORMATION 520

FIG. 5

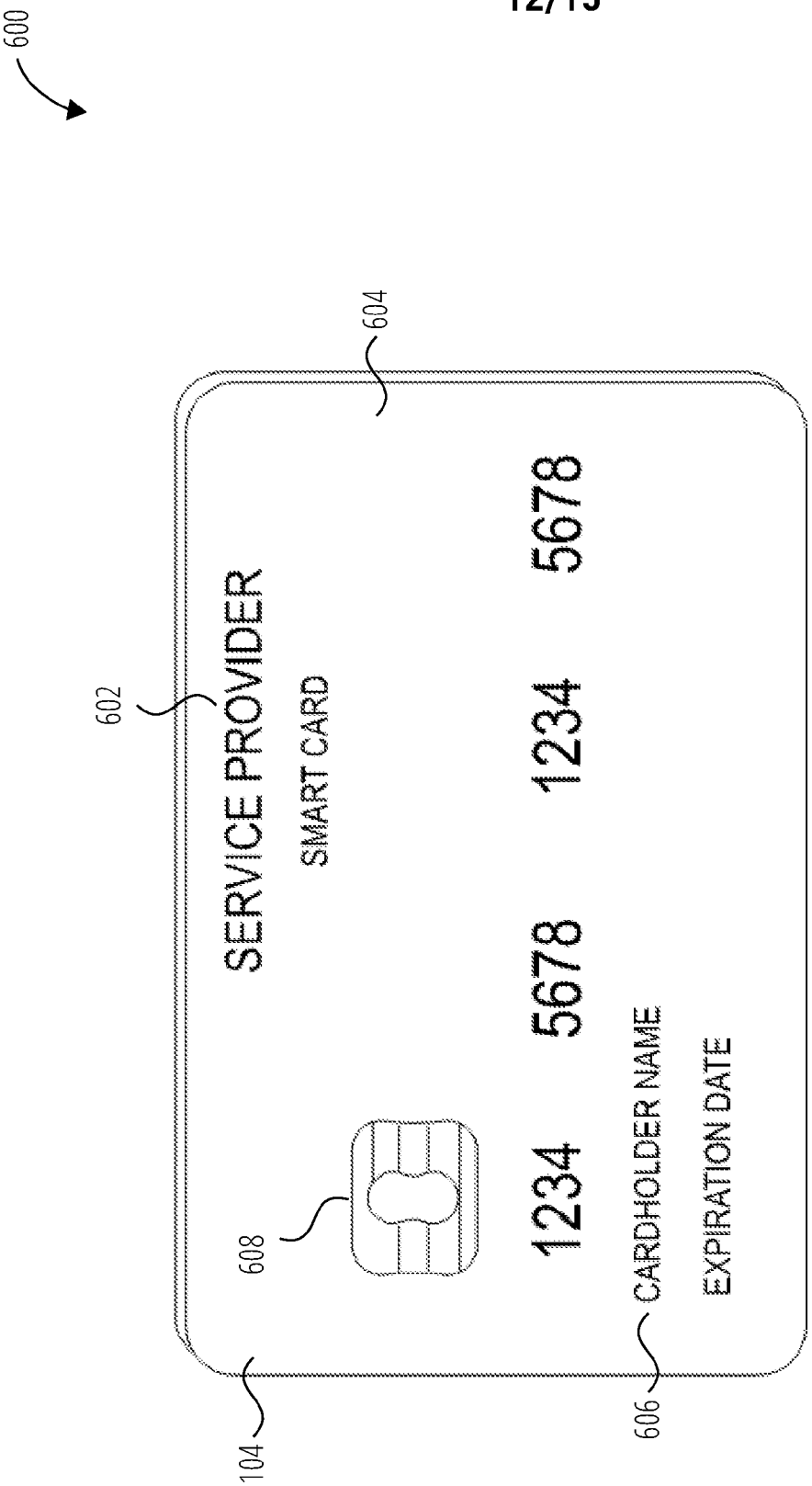

600
604
602
SERVICE PROVIDER
SMART CARD
5678
1234
5678
1234
608
CARDHOLDER NAME
EXPIRATION DATE
104
606

FIG. 6A

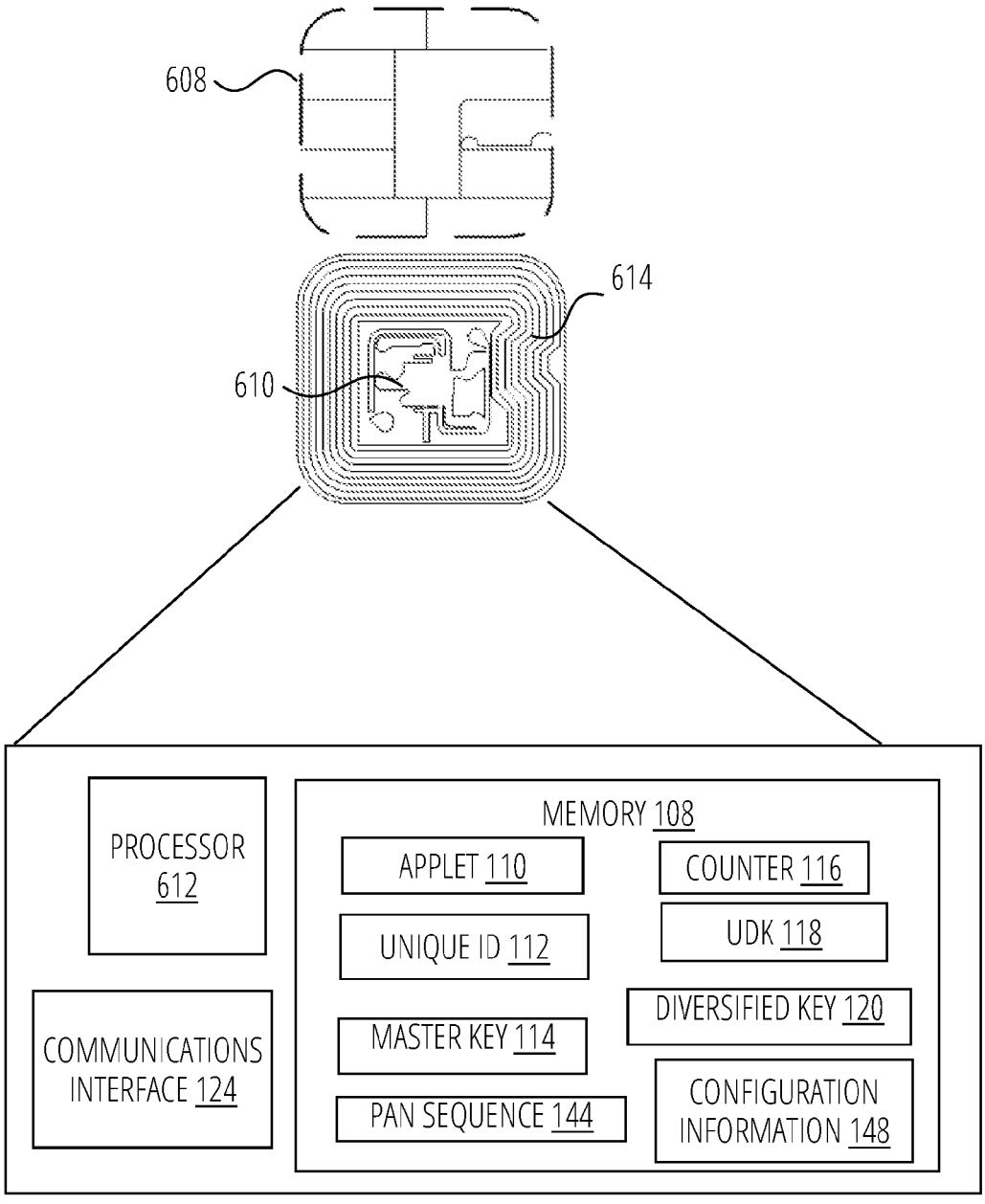

608
614
610
PROCESSOR 612
COMMUNICATIONS INTERFACE 124
MEMORY 108
APPLET 110
COUNTER 116
UNIQUE ID 112
UDK 118
DIVERSIFIED KEY 120
MASTER KEY 114
PAN SEQUENCE 144
CONFIGURATION INFORMATION 148

FIG. 6B

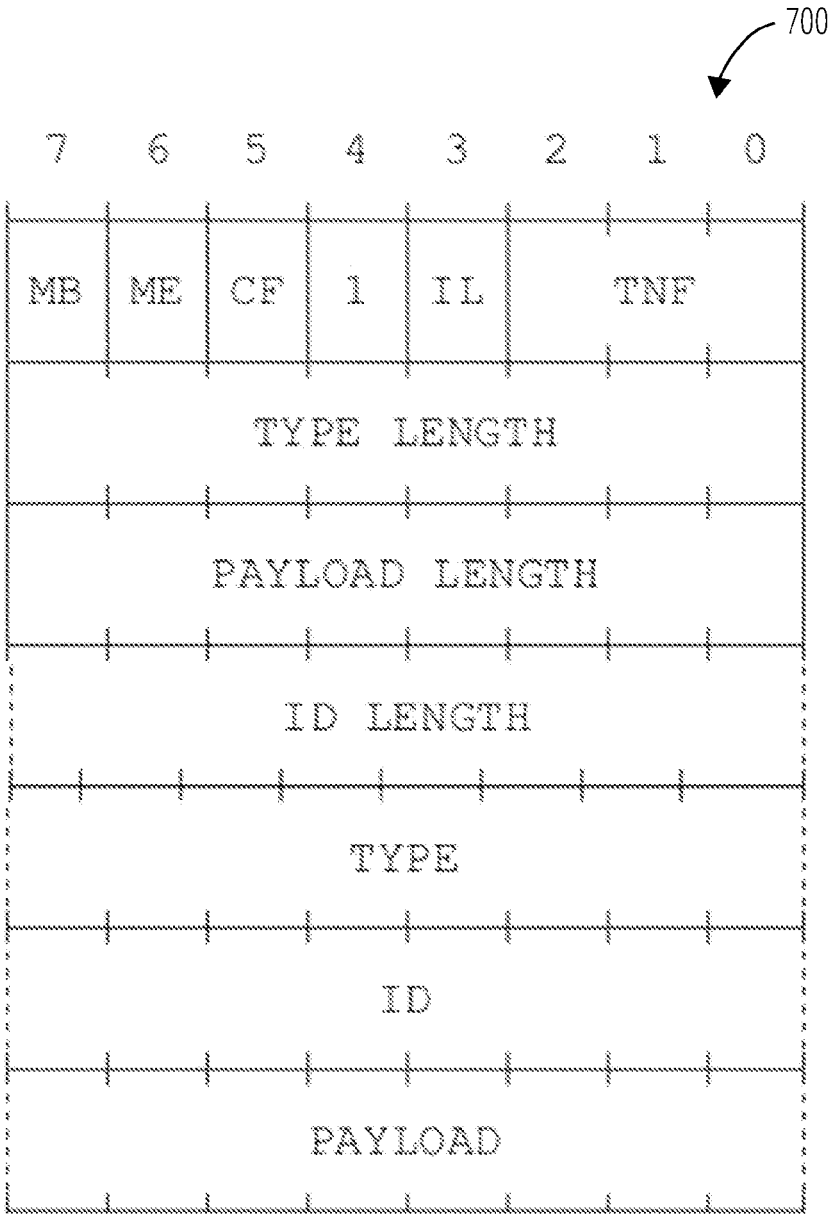

700
7 6 5 4 3 2 1 0
MB
ME
CF
1
IL
TNF
TYPE LENGTH
PAYLOAD LENGTH
ID LENGTH
TYPE
ID
PAYLOAD

FIG. 7

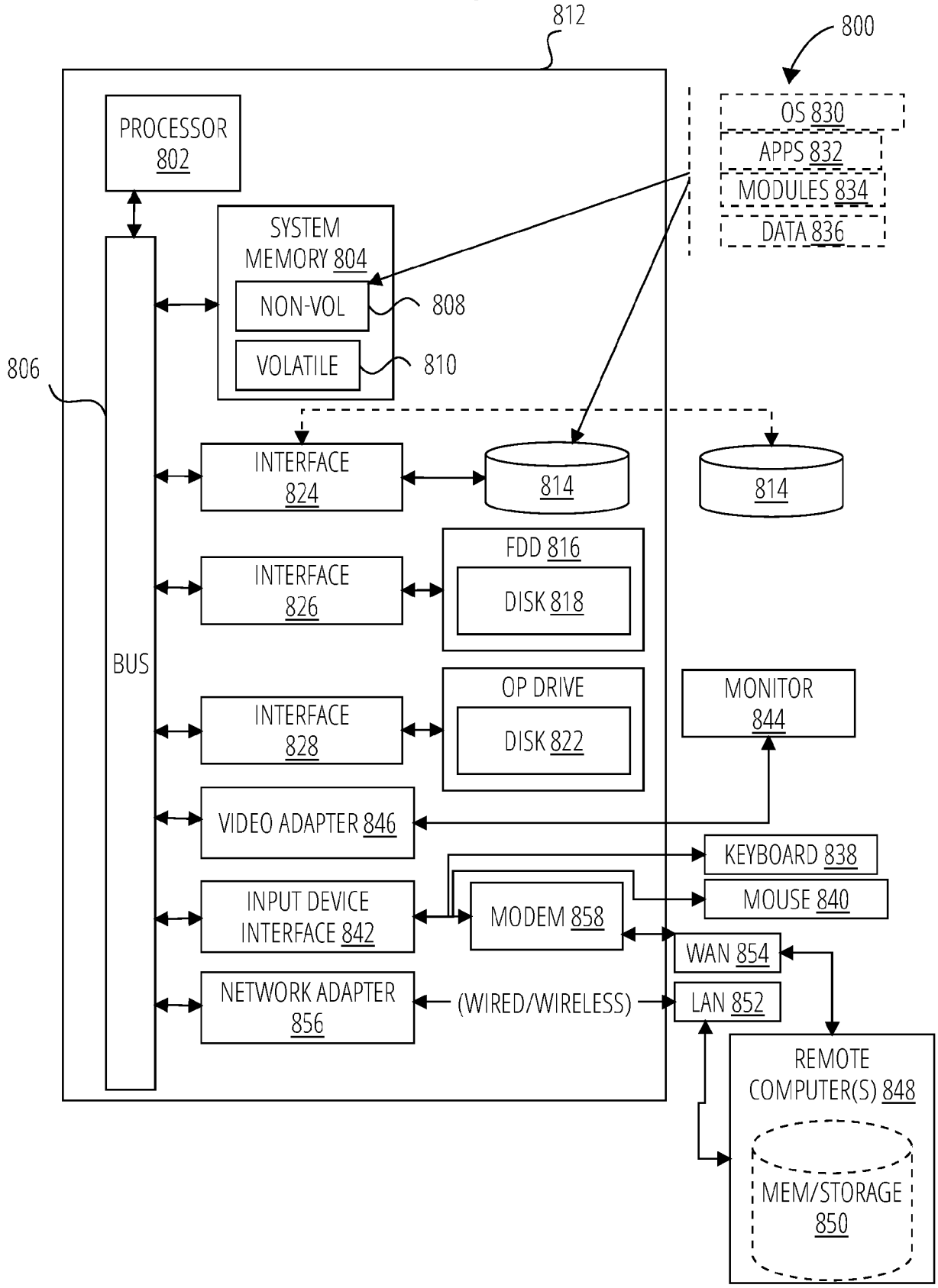

812
800
PROCESSOR 802
OS 830
APPS 832
MODULES 834
DATA 836
SYSTEM MEMORY 804
NON-VOL
808
VOLATILE
810
806
INTERFACE 824
814
814
FDD 816
INTERFACE 826
DISK 818
BUS
OP DRIVE
MONITOR 844
INTERFACE 828
DISK 822
VIDEO ADAPTER 846
KEYBOARD 838
INPUT DEVICE INTERFACE 842
MODEM 858
MOUSE 840
WAN 854
NETWORK ADAPTER 856
(WIRED/WIRELESS)
LAN 852
REMOTE COMPUTER(S) 848
MEM/STORAGE 850

FIG. 8