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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
19 December 2024 (19.12.2024)

WIPO | PCT

(10) International Publication Number
WO 2024/258850 A2

(51) International Patent Classification:
Not classified

(21) International Application Number:
PCT/US2024/033408

(22) International Filing Date:
11 June 2024 (11.06.2024)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
18/333,941 13 June 2023 (13.06.2023) US

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

(72) Inventors: **WIEKER, Jeffrey Carlyle**; c/o Capital One Services, LLC, 1680 Capital One Drive, McLean, Virginia 22102 (US). **MORETON, Paul Y.**; c/o Capital One Services, LLC, 1680 Capital One Drive, McLean, Virginia 22102 (US).

(74) Agent: **RUSSELL, Matthew** 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,

(54) Title: CONTACTLESS CARD-BASED AUTHENTICATION VIA WEB-BROWSER

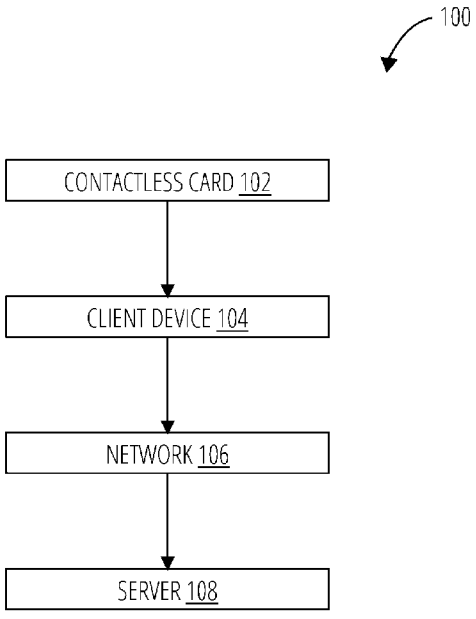

FIG. 1

(57) Abstract: A computer-implemented method to enable short-range wireless communication via a webpage on a computing device includes receiving, via a web-browser executing on the computing device, from the webpage, a first request to execute a computer-executable instruction, the computer-executable instruction requests data from an enterprise server. The method also includes triggering, in response to a second request from the enterprise server to authenticate the first request, the web-browser to execute a predetermined computer program. The method also includes scanning, by the predetermined computer program a cryptogram from a contactless card to authenticate the cryptogram and cause the enterprise server to send the data. The method also includes executing, via the web-browser, the computer-executable instruction from the first request in response to receiving the data sent by the enterprise server.

WO 2024/258850 A2

[Continued on next page]

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, 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:

— *without international search report and to be republished upon receipt of that report (Rule 48.2(g))*

CONTACTLESS CARD-BASED AUTHENTICATION VIA WEB-BROWSER

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application claims priority to U.S. Patent Application Serial No. 18/333,941, filed April 13, 2023, the disclosure of which is incorporated herein by reference in its entirety.

BACKGROUND

[0002] Authenticating a user involved in the operations is vital during several online functions, such as login transactions, payment transactions, document signing transactions, etc. Contactless cards are frequently used for such verification. Contactless cards are based on radio-frequency identification (RFID) technology that may be embedded into credit cards, identification cards, and other smart cards. This technology allows users to make online transactions, such as credit card transactions, identification transactions, etc., by performing certain gestures, such as bringing an assigned smart card within a specific distance of (or tapping on) specific areas of devices, such as point-of-sale terminals, mobile phones, etc. Such gestures enable the transfer of certain data for the purposes of completing the online operation(s). Before employing such gesture-based features, the devices, cards, etc., having such capability must be appropriately activated. However, existing authentication processes are limited to using specific applications, restricting access to certain portals and functions.

BRIEF SUMMARY

[0003] One general aspect includes a computer-implemented method to enable short-range wireless communication via a webpage on a computing device. The computer-implemented method includes receiving, via a web-browser executing on the computing device, from the webpage, a first request to execute a computer-executable instruction, the computer-executable instruction requests data from an enterprise server. The method also includes receiving, from the enterprise server, a second request to authenticate the first request from the webpage. The method also includes triggering, in response to the second request from the enterprise server, the web-browser to execute a predetermined computer program. The method also includes scanning, by the predetermined computer program, using a short-range wireless communication module of the computing device, a cryptogram from a contactless card. The method also includes in response to determining that the first request is authentic based on the cryptogram, transmitting, by the predetermined computer program, a response to the enterprise server

causing the enterprise server to send the data. The method also includes executing, via the web-browser, the computer-executable instruction from the first request in response to receiving the data sent by the enterprise server. Other embodiments of this aspect include corresponding computer systems, apparatus, and computer programs recorded on one or more computer storage devices, each configured to perform the actions of the methods.

[0004] One general aspect includes a computer device that includes a memory and a processor, the memory storing instructions that, when executed by the processor, configure the computing device to receive, via a web-browser executing on the computing device, from a webpage, a first request to execute a computer-executable instruction, the computer-executable instruction requests data from an enterprise server. The device is also configured to receive, from the enterprise server, a second request to authenticate the first request from the webpage. The device is also configured to trigger, in response to the second request from the enterprise server, the web-browser to execute a predetermined computer program. The device is also configured to scan, by the predetermined computer program, using a short-range wireless communication module of the computing device, a cryptogram from a contactless card. The device is also configured to, in response to determining that the first request is authentic based on the cryptogram, transmit, by the predetermined computer program, a response to the enterprise server causing the enterprise server to send the data. The device is also configured to execute, via the web-browser, the computer-executable instruction from the first request in response to receiving the data sent by the enterprise server. Other embodiments of this aspect include corresponding computer systems, apparatus, and computer programs recorded on one or more computer storage devices, each configured to perform the actions of the methods.

[0005] One general aspect includes a non-transitory computer-readable storage medium. The non-transitory computer-readable storage medium includes instructions to receive, via a web-browser executing on the computing device, from a webpage, a first request to execute a computer-executable instruction, the computer-executable instruction requests data from an enterprise server. The medium also includes instructions to receive, from the enterprise server, a second request to authenticate the first request from the webpage. The medium also includes instructions to trigger, in response to the second request from the enterprise server, the web-browser to execute a predetermined computer program. The medium also includes instructions to scan, by the predetermined computer program, using a short-range wireless communication module of the computing device, a cryptogram from a contactless card. The medium also

includes instructions to, in response to determining that the first request is authentic based on the cryptogram, transmit, by the predetermined computer program, a response to the enterprise server causing the enterprise server to send the data. The medium also includes instructions to execute, via the web-browser, the computer-executable instruction from the first request in response to receiving the data sent by the enterprise server. Other embodiments of this aspect include corresponding computer systems, apparatus, and computer programs recorded on one or more computer storage devices, each configured to perform the actions of the methods.

[0006] Non-transitory computer program products (i.e., physically embodied computer program products) are also described that store instructions, which, when executed by one or more data processors of one or more computing systems, cause at least one data processor to perform operations herein. Similarly, computer systems are also described, which may include one or more data processors and memory coupled to the one or more data processors. The memory may temporarily or permanently store instructions that cause at least one processor to perform one or more of the operations described herein. In addition, methods can be implemented by one or more data processors, which are either within a single computing system or distributed among two or more computing systems. Such computing systems can be connected and can exchange data and/or commands or other instructions or the like via one or more connections, including but not limited to a connection over a network (e.g., the Internet, a wireless wide area network, a local area network, a wide area network, a wired network, or the like), via a direct connection between one or more of the multiple computing systems, etc.

[0007] The details of one or more variations of the subject matter described herein are set forth in the accompanying drawings and the description below. Other features and advantages of the subject matter described herein will be apparent from the description and drawings, and from the claims.

BRIEF DESCRIPTION OF THE SEVERAL VIEWS OF THE DRAWINGS

[0008] 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.

[0009] FIG. 1 illustrates an aspect of the subject matter in accordance with one embodiment.

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

[0011] FIG. 3 illustrates an aspect of the subject matter in accordance with one embodiment.

[0012] FIG. 4 illustrates method 400 in accordance with one embodiment.

[0013] FIG. 5 illustrates an aspect of the subject matter in accordance with one embodiment.

[0014] FIG. 6 illustrates an aspect of the subject matter in accordance with one embodiment.

[0015] FIG. 7 illustrates a contactless card 102 in accordance with one embodiment.

[0016] FIG. 8 illustrates a transaction card component 800 in accordance with one embodiment.

[0017] FIG. 9 illustrates a sequence flow 900 in accordance with one embodiment.

[0018] FIG. 10 illustrates a data structure 1000 in accordance with one embodiment.

[0019] FIG. 11 is a diagram of a key system according to an example embodiment.

[0020] FIG. 12 is a flowchart of a method of generating a cryptogram according to an example embodiment.

[0021] FIG. 13 illustrates an aspect of the subject matter in accordance with one embodiment.

[0022] FIG. 14 illustrates an aspect of the subject matter in accordance with one embodiment.

DETAILED DESCRIPTION

[0023] FIG. 1 illustrates a data transmission system 100 according to an example embodiment. As further discussed below, system 100 may include contactless card 102, client device 104, network 106, and server 108. Although FIG. 1 illustrates single instances of the components, system 100 may include any number of components.

[0024] System 100 may include one or more contactless cards 102, which are further explained below. In some embodiments, contactless card 102 may be in wireless communication, utilizing NFC in an example, with client device 104.

[0025] System 100 may include client device 104, which may be a network-enabled computing device ("computing device" or "computer"). As referred to herein, a network-enabled computer may include but is not limited to a computer device, or communications device including, e.g., a server, a network appliance, a personal computer, a workstation, a phone, a handheld PC, a personal digital assistant, a thin client, a fat client, an Internet browser, or other device. Client device 104 also may be a mobile device; for example, a mobile device may include an iPhone, iPod, iPad from Apple® or any other mobile device running Apple's iOS® operating system, any device running Microsoft's Windows® Mobile

operating system, any device running Google's Android® operating system, and/or any other smartphone, tablet, or like wearable mobile device.

[0026] The client device 104 can include a processor and a memory, and it is understood that the processing circuitry may contain additional components, including processors, memories, error, and parity/CRC checkers, data encoders, anticollision algorithms, controllers, command decoders, security primitives and tamper proofing hardware, as necessary to perform the functions described herein. The client device 104 may further include a display and input devices. The display may be any type of device for presenting visual information, such as a computer monitor, a flat panel display, and a mobile device screen, including liquid crystal displays, light-emitting diode displays, plasma panels, and cathode ray tube displays. The input devices may include any device for entering information into the user's device that is available and supported by the user's device, such as a touchscreen, keyboard, mouse, cursor-control device, touchscreen, microphone, digital camera, video recorder or camcorder. These devices may be used to enter information and interact with the software and other devices described herein.

[0027] In some examples, client device 104 of system 100 may execute one or more applications, such as software applications, that enable, for example, network communications with one or more components of system 100 and transmit and/or receive data.

[0028] The client device 104 may be in communication with one or more server(s) 108 via one or more network(s) 106, and may operate as a respective front-end to back-end pair with server 108. The client device 104 may transmit, for example, from a mobile device application executing on client device 104, one or more requests to server 108. The one or more requests may be associated with retrieving data from server 108. The server 108 may receive the one or more requests from client device 104. Based on the one or more requests from client device 104, server 108 may be configured to retrieve the requested data from one or more databases (not shown). Based on receipt of the requested data from the one or more databases, server 108 may be configured to transmit the received data to client device 104, the received data being responsive to one or more requests.

[0029] System 100 may include one or more networks 106. In some examples, network 106 may be one or more of a wireless network, a wired network, or any combination of a wireless network and a wired network and may be configured to connect client device 104 to server 108. For example, network 106 may include one or more of a fiber optics network, a passive optical

network, a cable network, an Internet network, a satellite network, a wireless local area network (LAN), a Global System for Mobile Communication, a Personal Communication Service, a Personal Area Network, Wireless Application Protocol, Multimedia Messaging Service, Enhanced Messaging Service, Short Message Service, Time Division Multiplexing based systems, Code Division Multiple Access based systems, D-AMPS, Wi-Fi, Fixed Wireless Data, IEEE 1202.11 family of networking, Bluetooth, NFC, Radio Frequency Identification (RFID), Wi-Fi, and/or the like.

[0030] In addition, network 106 may include, without limitation, telephone lines, fiber optics, IEEE Ethernet 802.3, a wide area network, a wireless personal area network, a LAN, or a global network such as the Internet. In addition, network 106 may support an Internet network, a wireless communication network, a cellular network, or the like, or any combination thereof. Network 106 may further include one network, or any number of the exemplary types of networks mentioned above, operating as a stand-alone network or in cooperation with each other. Network 106 may utilize one or more protocols of one or more network elements to which they are communicatively coupled. Network 106 may translate to or from other protocols to one or more protocols of network devices. Although network 106 is depicted as a single network, it should be appreciated that according to one or more examples, network 106 may comprise a plurality of interconnected networks, such as, for example, the Internet, a service provider's network, a cable television network, corporate networks, such as credit card association networks, and home networks.

[0031] System 100 may include one or more servers 108. In some examples, server 108 may include one or more processors, which are coupled to memory. The server 108 may be configured as a central system, server, or platform to control and call various data at different times to execute a plurality of workflow actions. Server 120 may be configured to connect to the one or more databases. The server 108 may be connected to at least one client device 104.

[0032] FIG. 2 depicts an example client device 104 in accordance with one embodiment. The client device 104 is depicted as a mobile device, however, it is understood that the client device 104 can be any other computing device in other embodiments. The client device 104 facilitates executing one or more applications ("apps"). Each application includes one or more computer programs, i.e., one or more computer-executable instructions that are executed by the client device 104. For example, in FIG. 2, the client device 104 is depicted with icons for App 204a, App 204b, and App 204c. It is understood that the client device 104 can include additional,

fewer, and/or different applications. The applications, such as App 204a, App 204b, and App 204c can include games, banking applications, e-commerce applications, utility applications, entertainment applications, social media applications, web browsers, or any other such applications.

[0033] The applications are built for the specific platform of the client device 104, such as iOS® for the Apple® iPhone® or Android® for a Samsung® or any other manufacturer. Generally, the applications are downloaded and installed via an app store and have access to system resources, such as camera, short-range wireless communication module, secure memory, and other hardware and/or software features that the client device 104 provides. In some cases, when an application, such as the App 204a is installed, or used for the first time on the client device 104, a user 202 is requested to authorize/permit the application to use the system resources.

[0034] The contactless card 102 may be affiliated with one or more of the applications. For example, consider that the App 204a is affiliated with the contactless card 102. Here, by virtue of being "affiliated," the client device 104 facilitates App 204a to communicate with the contactless card 102, for example, using a short-range wireless communication module, such as NFC. As discussed herein, the affiliation between the contactless card 102 and the App 204a can be established at an earlier time, for example, during activation of the contactless card 102 and/or at the first use of the App 204a. The communication between the contactless card 102 and the App 204a can include the App 204a receiving information, such as security information, user identification, user authorization, etc., from the contactless card 102. In some examples, the App 204a can request for specific information from the contactless card 102. The contactless card 102 provides the particular information and/or responses based on the request from the App 204a. The App 204a can request information from the contactless card 102 in response to a request received by the App 204a from the server 108, in some cases.

[0035] In some embodiments, the contactless card 102 is a credit card, sometimes referred to as a "smart card," and the App 204a is an application provided by the bank that issues/services the contactless card 102. The user 202 can use the App 204a on the client device 104 to initiate an operation, such as a purchase or renting of a product, a service, and/or a combination thereof. Alternatively, or in addition, the user 202 can use the App 204a to initiate an operation, such as a money transfer, a withdrawal, a deposit, etc., or a combination thereof. It is understood that several other types of operations are possible and that the examples listed

herein are not to be considered limiting. The operation may include the App 204a communicating with the service/product provider's portal 206, which in turn may communicate with the server 108. For example, the service/product provider's portal 206 may communicate with the server 108 to receive confirmation, authorization, or other such information used for completing the operation initiated by the App 204a. The server 108, in response, may request authenticating that the user 202 is an authorized user and in possession of the contactless card 102. Accordingly, the server 108 can instruct the App 204a to confirm the presence of the contactless card 102. In some cases, the App 204a generates and displays a user interface instructing the user 202 to perform a gesture for authenticating the user 202. In some embodiments, the gesture may be performed with the contactless card 102, for example, tap, double-tap, swipe, capturing an image, or any other such gesture. Alternatively, or in addition, the gesture can further include entering an identification code, such as a password, passphrase, a PIN code, etc. Alternatively, or in addition, the gesture includes providing a biometric, such as a fingerprint, an iris scan, a voice sample, etc.

[0036] A "tap" may include the user 202 tapping the contactless card 102 on the client device 104. A "swipe" may include the user 202 swiping the contactless card 102 relative to the client device 104. "Capturing an image" may include using the client devices 104 to capture an image of a portion of the contactless card 102. In some examples, the gesture has to be performed relative to a particular portion of the client device 104, such as a top portion, a side portion, etc. It is understood that several other gestures may be used and that a combination of the gestures can also be used in some examples.

[0037] The gesture facilitates the App 204a to receive the information from the contactless card 102. The reception of the information based on the gesture facilitates confirming possession of the contactless card 102 by the user 202, and in turn, authentication of the user 202. The App 204a provides the information received from the contactless card 102 to the server 108. In some cases, the App 204a transforms the information before sending the information to the server 108. For example, the App 204a may secure the information, such as by encrypting the information before sending it. Alternatively, or in addition, the App 204a may append additional information, such as an identification of the client device 104 (e.g., an IP address, a MAC address, etc.), a timestamp, or any other such information before furnishing the information to the server 108.

[0038] The server 108 authenticates the user 202 upon receiving the information from the client device 104. The server 108, in some cases, further indicates the service/product provider's portal 206 that the user 202 has been authenticated. In response, the service/product provider's portal 206 completes the operation that the App 204a initiated. In this manner, the user 202 can complete the operation using the App 204a and address the technical challenges of authenticating the user 202 and confirming possession of the contactless card 102 by the user 202.

[0039] Technical challenges with initiating and completing the operation with certain service/product providers include the App 204a, which is affiliated with the contactless card 102, not being able to access portal 206 (e.g., website, server, etc.) of the service/product provider. For example, the inaccessibility can be due to incompatibility of the underlying computing technology being used by the App 204a and the service/product provider's portal 206. In turn, a practical effect of the technical challenge is that the user 202 is unable to access the product/service provided by the service/product provider.

[0040] In some cases, a solution to address the technical challenge of such technological incompatibility is for the service/product provider to provide an application, for example, App 204b. The user 202 initiates the operation via the client device 104 using the App 204b. The App 204b communicates with the App 204a upon receiving the request to complete the operation. The App 204b, in turn, facilitates completing the operation as described herein. The App 204b, in this manner, acts as an interface between the App 204a (affiliated with the contactless card 102) and the service/product provider's portal 206, and addresses the technological incompatibility. A technical challenge with this solution of using App 204b is that the developer of the App 204b has to have access to the App 204a and vice versa, which may not always be the case. Additionally, the interactive nature of the applications, in this case App 204a and App 204b, may require that the two applications be updated (by respective distinct developers) when either one of the applications is updated.

[0041] Additionally, the technical challenge with initiating and completing the operation with certain service/product providers persists in the case where the service/product provider does not have an application (App 204b) that can communicate with the App 204a affiliated with the contactless card 102. In such cases, existing solution to access the portal 206 is to use an application, such as a web-browser 208, on the client device 104. However, a technical challenge with using the web-browser 208 to access the portal 206 to initiate and perform the

operation is that the client device 104 does not facilitate the web-browser 208, unlike the application (e.g., App 204a) to use all of the resources of the client device 104. Particularly, the web-browser 208 may be prevented from using some of the resources like the short-range wireless communication module of the client device 104. The client device 104 may prevent the web-browser 208 from accessing some of the resources of the client device 104 for several reasons, including security. For example, if the user 202 may inadvertently access malicious code that is embedded in a webpage 210 accessed by the web-browser 208. Suppose such malicious code were to access resources, such as the short-range wireless communication module of the client device 104. In that case, the malicious code may access sensitive and private data of the user 202. The client device 104 may prevent the web-browser 208 from accessing additional or other resources in other embodiments. Hence, without access to some of the resources of the client device 104, the web-browser 208 is unable to facilitate authenticating the user 202 and/or the possession of the contactless card 102 as described herein (using one or more gestures). It should be noted that the web-browser 208 is another application, but a special type of application identified by the client device 104, and accordingly associated with the restricted access of the resources.

[0042] The technical solutions described herein address such technical challenges, including the technological incompatibility between the service/product provider's portal 206 and the application (App 204a) affiliated with the contactless card 102; and the inability of the web-browser 208 of accessing all of the resources of the client device 104. The technical solutions herein are accordingly rooted in computing technology, particularly addressing incompatibility-related issues. Further, the technical solutions herein provide improvement(s) to computing technology by facilitating the user 202 to access and complete operations on a service/product provider's portal 206 that is incompatible with the application App 204a affiliated with the contactless card 102. The technical solutions described herein provide a practical application to the user 202, because now s/he can access the portal 206 and proceed with one or more operations that could not be completed via the App 204a (and in absence of App 204b).

[0043] Additionally, the technical solutions described herein provide a practical application that the service/product provider does not have to create and distribute an application (e.g., App 204b) for the client device 104. Creating, distributing, and maintaining applications (e.g., App 204a) for every type of client device 104 can become impractical for the service/client provider. Accordingly, the technical solutions described herein also provide a practical

application in this regard by reducing the number of applications that the service/product provider has to develop, distribute, and maintain.

[0044] Additionally, the technical solutions described herein provide a practical application to the user 202 that s/he does not have to install and maintain applications (e.g., App 204b) for each and every service/product provider s/he may be interacting with. By reducing the number of applications in this manner, the technical solutions facilitate the user 202 to increase the efficiency of the limited memory/storage on the client device 104. Further, each application installed on the client device 104 presents a security risk, and minimizing the number of applications installed on the client device 104 may be desired by the user 202.

[0045] To address the technical challenges and to provide the practical applications described herein, the technical solutions described herein facilitate the user 202 to access and complete the one or more operations on the portal 206 via the web-browser 208 of the client device 104. The web-browser 208 can be any type of browser, such as Safari®, Chrome™, Opera™, etc. The web-browser 208 facilitates accessing a webpage 210 provided by the portal 206. The webpage 210 facilitates the user 202 to initiate the operation via the portal 206. It is understood that the portal 206 may provide other ways to begin and to perform the operation instead of the webpage 210, for example, a widget, an applet, a script, or any other such computer programming resource that can be accessed by the web-browser 208.

[0046] The technical solutions described herein address such technical challenges by using a predetermined computer program that the web-browser 208 accesses when the user 202 initiates the operation. In some cases, the web-browser 208 accesses the predetermined computer program in response to the user 202 initiating the operation of a particular type that requires the authentication of the user 202 and/or confirming possession of the contactless card 102. Alternatively, or in addition, the web-browser 208 accesses the predetermined computer program in response to the user 202 initiating the operation on specific portals 206, which are included in a list of portals accessible by the web-browser 208. Alternatively, or in addition, the web-browser 208 accesses the predetermined computer program in response to web-browser 208 receiving a request from the portal 206 and/or the server 108 to authenticate the user 202 and/or confirm the possession of the contactless card 102.

[0047] In some embodiments, the predetermined computer program is one from a group of computer programs associated with the web-browser 208, the group of computer programs comprising an extension, a plugin, a component, and an addon. The predetermined computer

program is depicted as an extension 212, but can be any other type of predetermined computer program that the web-browser 208 can access. Further, while the extension 212 is depicted as being inside the web-browser 208, the extension 212 may be separate from the web-browser 208 in some embodiments. In some embodiments, the extension 212 can be provided by the issuer of the contactless card 102.

[0048] The extension 212 can customize the web browsing user-experience on the client device 104. In some embodiments, the extension 212 can use native APIs and frameworks of the operating system (e.g., iOS® and iPadOS®) of the client device 104, as well as web technologies such as HTML, CSS, and JavaScript. The extension 212 can facilitate the web-browser 208 to read and modify the content of the webpage 210. The extension 212 is built with native programming language (e.g., XCode®) of the client device 104 and facilitates communication and sharing of data with native applications. Accordingly, the extension 212 facilitates integrating application content (e.g., from App 204a) into the web-browser 208 or sending web data (from the web-browser 208) back to the application (e.g., App 204a) to create a unified experience. The extension 212, in some embodiments, may facilitate blocking certain content types on the webpage 210 being accessed by the web-browser 208. Blocking behaviors may include hiding elements, blocking loads, and removing cookies from requests received by the web-browser 208.

[0049] FIG. 3 illustrates an extension 212 in accordance with one or more embodiments. The extension 212 includes at least browser-code 302 and native-code 304. In some embodiments, the browser-code 302 includes computer-executable instructions, for example, JavaScript code and web files, that work in the web-browser 208. The native-code 304 includes computer-executable instructions, for example, using functions/application programming interface (API) of the client device's (104) operating system. The native-code 304 mediates between the application that is affiliated with the contactless card 102, such as the App 204a, and the browser-code 302. In some embodiments, messaging APIs communicate events and event data between the browser-code 302 and the native-code 304.

[0050] It should be noted that the App 204a, the web-browser 208, and the extension 212, each execute (i.e., operate) independently in their own sandboxed environments, i.e., separated containers. Because the App 204a and the extension 212 run in respective sandboxed environments, they cannot share data in their respective containers. In some embodiments, data can be stored in a shared space that both the App 204a and the extension 212 can access and

update. For example, in the iOS® operating system, such a shared space can be enabled by enabling the "app groups" option. It is understood that in other operating system environments, additional and/or different options may have to be enabled.

[0051] The browser-code 302 can facilitate providing a user-interface 306 for the extension 212. In some embodiments, the browser-code 302 sends messages from a background script or from extension pages. The user-interface 306 can include interactive elements that are rendered as part of the web-browser 208 and/or as part of the App 204a. The browser-code 302 can include a specific computer-executable instruction, such as a message call that sends a message directed to the App 204a.

[0052] The App 204a includes a message handler function that is assigned to respond to the message call. The message call and the message handler are paired with each other, i.e., use a predetermined protocol to facilitate transferring particular data as specific parameters. For example, JSON or other such protocols may be used for such transfer of data. For example, the messages can be sent from the App 204a to the extension 212 to notify of events, like when the user 202 clicks a button or when data that the extension 212 script uses changes.

[0053] Content scripts that are injected into web content of the webpage 210 cannot send messages to the App 204a. However, with messaging, the webpage 210 can control features in the extension 212 based on events or data, or the webpage 210 can request and use data from the extension 212. The extension 212 is configured to receive messages from a list of webpages, including the webpage 210, to facilitate the webpage 210 to request and access the data. For example, to enable messaging from the webpage 210, an identity of the webpage 210 is added to the extension's 212 configuration, such as a JSON manifest file of the extension 212. In response, when the webpage 210 sends a message using the web-browser's 208 messaging instructions (e.g., browser.runtime.sendMessage), the extension 212 is notified to handle the message. In some embodiments, the message from the webpage 210 includes an identifier of the extension 212, message data, and a closure to handle the response from the extension 212. The identifier is a unique identification associated with each extension 212.

[0054] In some embodiments, if the extension 212 needs to handle more continuous data from the webpage 210, a port connection is established between the webpage 210 and the extension 212. The extension 212 listens in the background for any such incoming port connection requests from the webpage 210. The created port is then used to communicate data between the extension 212 and the webpage 210 using messages directed to the port. Further, the extension

212 includes computer-executable instructions that add functionality to handle messages that the webpage 210 sends to the extension 212, and respond to that webpage 210.

[0055] Accordingly, the extension 212, i.e., the predetermined computer program, facilitates the webpage 210 to communicate with the App 204a that is affiliated with the contactless card 102. Using such communication enabled by the extension 212, the technical solutions described herein facilitate the user 202 to access the portal 206. Further, in response to one or more requests from the webpages 210 of the portal 206, the technical solutions herein facilitate authenticating the user 202 and confirming possession of the contactless card 102 using the one or more gestures described herein. For example, upon receiving a request from the webpage 210, the extension 212 requests the App 204a to perform the authenticating. The extension 212 subsequently provides a result of the authenticating to the webpage 210.

[0056] FIG. 4 illustrates a method 400 in accordance with one embodiment. Method 400 can be a computer-implemented method, for example, executed by the client device 104. In block 402, method 400 includes receiving, via the web-browser 208 executing on the client device 104, from the webpage 210, a first request to execute a computer-executable instruction, the computer-executable instruction requests data from the server 108. The computer-executable instruction may be part of a secure transaction being performed via the web-browser 208 on the webpage 210 of the service/product provider's portal 206. For example, the secure transaction can be an online operation, and include at least one of a login transaction, a commercial transaction, and a data transfer transaction. The server 108 can be a bank server, an authentication server, an intermediate server, or any other server 108 that facilitates authenticating the user 202 and/or provides information about the user 202 upon authentication.

[0057] For example, the user 202 may initiate the operation with the service/product provider's portal 206 via the webpage 210. In response, the portal 206 may request that the user 202 provide additional information to complete the operation. The additional information can include but is not limited to authentication information, identification information, bank account information, payment authorization information, or the like. The portal 206 may request that a third-party enterprise server, such as the server 108 provide such information. Accordingly, the computer-executable instruction requests data from the server 108 to facilitate such information provision.

[0058] In block 404, method 400 includes receiving a second request from the server 108 to authenticate the first request from the webpage. Authenticating the first request includes

verification that an authorized user initiated the first request. Determining that the user 202 is an authorized user may require confirmation that the user 202 is in possession of the contactless card 102. Hence, the server 108 sends the second request to the client device 104 to perform the authentication and/or confirmation of the possession.

[0059] In block 406, method 400 includes triggering, in response to the second request, by the web-browser 208 execution of a predetermined computer program, i.e., the extension 212. In some embodiments, the second request may be issued by the server 108. For example, the portal 206 may request that a third-party enterprise server, such as the server 108 provide authentication information of the user 202. Accordingly, the web-browser 208 triggers the extension 212 to request data from the server 108 to facilitate such information provision. In other embodiments, the second request may be issued by the portal 206. For example, the portal 206 may determine that the operation initiated by the user 202 is via the web-browser 208. In response, the portal 206 triggers the extension 212 for the web-browser 208 to request the authentication via the server 108. In yet other embodiments, the second request may be issued by the client device 104 itself. For example, the web-browser 208 may identify the operation being performed as one of the predetermined operations, such as a payment operation, a login operation, a banking operation, a document signing operation, etc., which may benefit from additional security. In response, the web-browser 208 proactively triggers the extension 212, which causes the user 202 to be authenticated using the contactless card 102.

[0060] As described herein, upon receipt of the second request, the extension 212 causes the the client device 104 to authenticate the user 202 and confirm possession of the contactless card 102. For example, the extension 212 can be configured to use message handling to listen for the second request from the server 108. The second request includes an identification of the extension 212 and a computer-executable instruction to cause the client device 104 to authenticate the user 202. The identification of the server 108 may be stored in the list of servers that are enabled for communication with the extension 212.

[0061] In block 408, method 400 includes scanning, by the extension 212, using a short-range wireless communication module of the client device 104, authentication information of the user 202 from the contactless card 102. In some embodiments, the extension 212 uses the App 204a (native application) to facilitate authenticating the user 202 and confirming possession of the contactless card 102. For example, the gesture-based technique(s) described herein can be used for the authentication. In some embodiments, the App 204a can be initiated and executed on

the client device 104, and generating and displaying the user-interface 306 that instructs the user 202 to perform one or more gestures with the contactless card 102 in relation to the client device 104. Alternatively, the App 204a is executed in the background, and the user-interface 306 is generated and displayed as part of the web-browser 208, with the instruction to perform the contactless card 102 related gesture.

[0062] As part of the gesture-based authentication, the client device 104 receives from the contactless card 102, an identification information that is stored on the contactless card 102. In some embodiments, the identification information can be provided in the form of a cryptogram. The identification information or any other information received from the contactless card 102 can be in any other secure form in other embodiments.

[0063] In block 410, method 400 includes, in response to determining that the first request is authentic based on the authentication information received, transmitting, by the extension 212, a response to the server 108 causing the server 108 to send the data requested by the portal 206. In some embodiments, determining that the first request is authentic based on the cryptogram includes validating the cryptogram by the extension 212. The validation may be performed by the extension 212 based on a comparison with stored information. In some embodiments, the extension 212 transmits a response of the authentication to the server 108 causing the server 108 to send the data for the computer-executable instruction from the first request. The server 108 may send the data only if the authentication passes. If the authentication fails, the server 108 may send another data that causes a user-notification, and aborting the operation.

[0064] In some embodiments, determining that the first request is authentic based on the cryptogram includes transmitting the cryptogram by the extension 212, for receipt by the server 108. The server 108 validates the information in the cryptogram, for example, by comparing the information with stored information. In response to validating the cryptogram by the server 108, the server 108 sends the data for the computer-executable instruction.

[0065] In some embodiments, upon authenticating (by the client device 104 or by the server 108), the server 108 may send the data directly to the portal 206. Alternatively, or in addition, the server 108 sends the data to the client device 104, which, in turn, sends the data to the portal 206 via the web-browser 208.

[0066] In block 412, method 400 includes executing, via the web-browser 208, the computer-executable instruction from the first request in response to receiving the data sent by the server 108. Accordingly, the client device 104 executes the computer-executable instruction from the

first request only upon authenticating the user 202. In this manner, the method 400 facilitates the web-browser 208 to be used by the user 202 to perform an operation that requires that the user 202 be authenticated using a contactless card 102 related gesture that uses one or more resources of the client device 104. In some embodiments, executing the computer-executable instruction may include completing the operation by providing payment information, login information, banking information, or any other such information that facilitates completing the operation.

[0067] Consider an example scenario where a customer (user 202) is purchasing an item from an online portal (206) using a credit card (contactless card 102) via a web-browser (208), mobile phone, or any other computing device (client device 104). The user 202 initiates the checkout process (online operation) via the portal. The portal may require that the user provides additional authentication information and sends a request to the web-browser. In some examples, the request from the portal may request to trigger the extension (212) of the web-browser. Alternatively, the web-browser, in response to the request from the portal, the web-browser determines the extension that has to be triggered. In yet other embodiments, the web-browser, in response to the first request from the portal, issues a request to an enterprise server (server 108) affiliated with the credit card to provide the authentication information. The enterprise server responds with a second request, in response to which the web-browser triggers the extension. Upon being triggered, the extension facilitates an application (App 204a) affiliated with the credit card to perform a gesture-based authentication. The result of the authentication is provided to the enterprise server and/or the portal. Based on the authentication result, the operation is either completed (checkout processed) or aborted (checkout denied).

[0068] It should be understood that the above is just one example scenario of a practical application of the technical solutions described herein and that not a limiting example. The technical solutions described herein can be used in other example scenarios, such as facilitating the user 202 to login to a system with additional authentication. Alternatively, the technical solutions described herein can facilitate the user to transfer digital information using additional security measures using the gesture-based authentication. Several other practical applications of the technical solutions described herein are possible.

[0069] FIG. 5 illustrates an example routine for initiating and completing an operation or transaction using a web-browser of a client device, where the operation requires a user-

authentication based on a physical transaction card. Although the example routine depicts a particular sequence of operations, the sequence may be altered without departing from the scope of the present disclosure. For example, some of the operations depicted may be performed in parallel or in a different sequence that does not materially affect the function of the routine. In other examples, different components of an example device or system that implements the routine may perform functions at substantially the same time or in a specific sequence.

[0070] According to some examples, the method includes receiving computer-executable instructions to initiate a transaction with an online portal 206 via a web-browser 208 at block 502. For example, the transaction can include a login transaction, an online purchase, a banking transaction (e.g., money transfer, payment, etc.), a data transfer transaction (e.g., content upload, content download, etc.), etc. It should be noted that the transaction is being performed via the web-browser 208 (e.g., Safari, Chrome, Firefox, etc.) and not an application or other type of computer program. The transaction can be initiated by the user 202 in some embodiments.

[0071] According to some examples, the method includes determining that authentication is to be performed at block 504. The determination may be made by the portal 206 in some embodiments. In other embodiments, the determination may be made by a server 108 (distinct from the portal 206), where the server 108 performs one or more operations that are part of the transaction. For example, the portal 206 may request an authentication via the server 108, which, in turn, performs the authentication via the client device 104. In yet other embodiments, the web-browser 208 determines that the authentication is required. The authentication can include authenticating the user 202 based on a transaction card, such as the contactless card 102. The authentication can further include ensuring that the user 202 is in physical possession of the contactless card 102.

[0072] According to some examples, the method includes detecting an extension 212 of the web-browser 208 to perform the authentication at block 506. The extension 212 may be a predetermined computer program that the web-browser 208 can access and cause to execute. The extension 212 can be one of several extensions of the web-browser 208. The portal 206, the server 108 may request the web-browser 208 to determine if the extension 212 is installed on the client device 104 being used for the transaction. Alternatively, the web-browser 208 may detect if the extension 212 exists independently, without an incoming request. In some

embodiments, if the extension 212 does not exist, the transaction may not be completed via the web-browser 208. A notification may be generated and displayed via the user-interface 306 indicating that the transaction was not completed and, in some embodiments, suggesting alternatives to the user 202.

[0073] According to some examples, in the case where the extension 212 is available and detected, the method includes triggering the extension 212 to perform the authentication at block 508. Upon being triggered, the extension 212 facilitates performing the authentication. For example, the extension 212 uses an application (App 204a) affiliated with the contactless card 102 to perform a gesture-based authentication.

[0074] According to some examples, the method includes providing the result of the authentication facilitated by the extension 212 at block 510. According to some examples, the method includes proceeding with the initiated transaction further based on the authentication result at block 512.

[0075] In this manner, technical solutions described herein facilitate initiating and completing an operation or transaction using a web-browser 208 of a client device 104, where the operation requires a user-authentication based on a physical presence of a contactless card 102 in a predetermined vicinity of the client device 104.

[0076] FIG. 6 illustrates a data transmission system according to an example embodiment. System 600 may include a transmitting or transmitting device 604, a receiving or receiving device 608 in communication, for example, via network 606, with one or more servers 602. Transmitting or transmitting device 604 may be the same as, or similar to, client device 104 discussed above with reference to FIG. 1. Receiving or receiving device 608 may be the same as, or similar to, client device 104 discussed above with reference to FIG. 1. Network 606 may be similar to network 106 discussed above with reference to FIG. 1. Server 602 may be similar to server 108 discussed above with reference to FIG. 1. Although FIG. 6 shows single instances of components of system 600, system 600 may include any number of the illustrated components.

[0077] When using symmetric cryptographic algorithms, such as encryption algorithms, hash-based message authentication code (HMAC) algorithms, and cipher-based message authentication code (CMAC) algorithms, it is important that the key remain secret between the party that originally processes the data that is protected using a symmetric algorithm and the

key, and the party who receives and processes the data using the same cryptographic algorithm and the same key.

[0078] It is also important that the same key is not used too many times. If a key is used or reused too frequently, that key may be compromised. Each time the key is used, it provides an attacker an additional sample of data which was processed by the cryptographic algorithm using the same key. The more data which the attacker has which was processed with the same key, the greater the likelihood that the attacker may discover the value of the key. A key used frequently may be comprised in a variety of different attacks.

[0079] Moreover, each time a symmetric cryptographic algorithm is executed, it may reveal information, such as side-channel data, about the key used during the symmetric cryptographic operation. Side-channel data may include minute power fluctuations which occur as the cryptographic algorithm executes while using the key. Sufficient measurements may be taken of the side-channel data to reveal enough information about the key to allow it to be recovered by the attacker. Using the same key for exchanging data would repeatedly reveal data processed by the same key.

[0080] However, by limiting the number of times a particular key will be used, the amount of side-channel data which the attacker is able to gather is limited and thereby reduce exposure to this and other types of attack. As further described herein, the parties involved in the exchange of cryptographic information (e.g., sender and recipient) can independently generate keys from an initial shared master symmetric key in combination with a counter value, and thereby periodically replace the shared symmetric key being used with needing to resort to any form of key exchange to keep the parties in sync. By periodically changing the shared secret symmetric key used by the sender and the recipient, the attacks described above are rendered impossible.

[0081] Referring to FIG. 6, system 600 may be configured to implement key diversification. For example, a sender and recipient may desire to exchange data (e.g., original sensitive data) via respective devices 604 and 608. As explained above, although single instances of transmitting device 604 and receiving device 608 may be included, it is understood that one or more transmitting devices 604 and one or more receiving devices 608 may be involved so long as each party shares the same shared secret symmetric key. In some examples, the transmitting device 604 and receiving device 608 may be provisioned with the same master symmetric key. Further, it is understood that any party or device holding the same secret symmetric key may perform the functions of the transmitting device 604 and similarly any party holding the same

secret symmetric key may perform the functions of the receiving device 608. In some examples, the symmetric key may comprise the shared secret symmetric key which is kept secret from all parties other than the transmitting device 604 and the receiving device 608 involved in exchanging the secure data. It is further understood that both the transmitting device 604 and receiving device 608 may be provided with the same master symmetric key, and further that part of the data exchanged between the transmitting device 604 and receiving device 608 comprises at least a portion of data which may be referred to as the counter value. The counter value may comprise a number that changes each time data is exchanged between the transmitting device 604 and the receiving device 608.

[0082] System 600 may include one or more networks 606. In some examples, network 606 may be one or more of a wireless network, a wired network or any combination of wireless network and wired network, and may be configured to connect one or more transmitting devices 604 and one or more receiving devices 608 to server 602. For example, network 606 may include one or more of a fiber optics network, a passive optical network, a cable network, an Internet network, a satellite network, a wireless LAN, a Global System for Mobile Communication, a Personal Communication Service, a Personal Area Network, Wireless Application Protocol, Multimedia Messaging Service, Enhanced Messaging Service, Short Message Service, Time Division Multiplexing based systems, Code Division Multiple Access based systems, D-AMPS, Wi-Fi, Fixed Wireless Data, IEEE 1202.11 family network, Bluetooth, NFC, RFID, Wi-Fi, and/or the like.

[0083] In addition, network 606 may include, without limitation, telephone lines, fiber optics, IEEE Ethernet 1302.3, a wide area network, a wireless personal area network, a LAN, or a global network such as the Internet. In addition, network 606 may support an Internet network, a wireless communication network, a cellular network, or the like, or any combination thereof. Network 606 may further include one network, or any number of the exemplary types of networks mentioned above, operating as a stand-alone network or in cooperation with each other. Network 606 may utilize one or more protocols of one or more network elements to which they are communicatively coupled. Network 606 may translate to or from other protocols to one or more protocols of network devices. Although network 606 is depicted as a single network, it should be appreciated that according to one or more examples, network 606 may comprise a plurality of interconnected networks, such as, for example, the Internet, a

service provider's network, a cable television network, corporate networks, such as credit card association networks, and home networks.

[0084] In some examples, one or more transmitting devices 604 and one or more receiving devices 608 may be configured to communicate and transmit and receive data between each other without passing through network 606. For example, communication between the one or more transmitting devices 604 and the one or more receiving devices 608 may occur via at least one of NFC, Bluetooth, RFID, Wi-Fi, and/or the like.

[0085] At block 610, when the transmitting device 604 is preparing to process the sensitive data with symmetric cryptographic operation, the sender may update a counter. In addition, the transmitting device 604 may select an appropriate symmetric cryptographic algorithm, which may include at least one of a symmetric encryption algorithm, HMAC algorithm, and a CMAC algorithm. In some examples, the symmetric algorithm used to process the diversification value may comprise any symmetric cryptographic algorithm used as needed to generate the desired length diversified symmetric key. Non-limiting examples of the symmetric algorithm may include a symmetric encryption algorithm such as 3DES or AES128; a symmetric HMAC algorithm, such as HMAC-SHA-256; and a symmetric CMAC algorithm such as AES-CMAC. It is understood that if the output of the selected symmetric algorithm does not generate a sufficiently long key, techniques such as processing multiple iterations of the symmetric algorithm with different input data and the same master key may produce multiple outputs which may be combined as needed to produce sufficient length keys.

[0086] At block 612, the transmitting device 604 may take the selected cryptographic algorithm, and using the master symmetric key, process the counter value. For example, the sender may select a symmetric encryption algorithm, and use a counter which updates with every conversation between the transmitting device 604 and the receiving device 608. The transmitting device 604 may then encrypt the counter value with the selected symmetric encryption algorithm using the master symmetric key, creating a diversified symmetric key.

[0087] In some examples, the counter value may not be encrypted. In these examples, the counter value may be transmitted between the transmitting device 604 and the receiving device 608 at block 612 without encryption.

[0088] At block 614, the diversified symmetric key may be used to process the sensitive data before transmitting the result to the receiving device 608. For example, the transmitting device 604 may encrypt the sensitive data using a symmetric encryption algorithm using the

diversified symmetric key, with the output comprising the protected encrypted data. The transmitting device 604 may then transmit the protected encrypted data, along with the counter value, to the receiving device 608 for processing.

[0089] At block 616, the receiving device 608 may first take the counter value and then perform the same symmetric encryption using the counter value as input to the encryption, and the master symmetric key as the key for the encryption. The output of the encryption may be the same diversified symmetric key value that was created by the sender.

[0090] At block 618, the receiving device 608 may then take the protected encrypted data and using a symmetric decryption algorithm along with the diversified symmetric key, decrypt the protected encrypted data.

[0091] At block 620, as a result of the decrypting the protected encrypted data, the original sensitive data may be revealed.

[0092] The next time sensitive data needs to be sent from the sender to the recipient via respective transmitting device 604 and receiving device 608, a different counter value may be selected producing a different diversified symmetric key. By processing the counter value with the master symmetric key and same symmetric cryptographic algorithm, both the transmitting device 604 and receiving device 608 may independently produce the same diversified symmetric key. This diversified symmetric key, not the master symmetric key, is used to protect the sensitive data.

[0093] As explained above, both the transmitting device 604 and receiving device 608 each initially possess the shared master symmetric key. The shared master symmetric key is not used to encrypt the original sensitive data. Because the diversified symmetric key is independently created by both the transmitting device 604 and receiving device 608, it is never transmitted between the two parties. Thus, an attacker cannot intercept the diversified symmetric key and the attacker never sees any data which was processed with the master symmetric key. Only the counter value is processed with the master symmetric key, not the sensitive data. As a result, reduced side-channel data about the master symmetric key is revealed. Moreover, the operation of the transmitting device 604 and the receiving device 608 may be governed by symmetric requirements for how often to create a new diversification value, and therefore a new diversified symmetric key. In an embodiment, a new diversification value and therefore a new diversified symmetric key may be created for every exchange between the transmitting device 604 and receiving device 608.

[0094] In some examples, the key diversification value may comprise the counter value. Other non-limiting examples of the key diversification value include: a random nonce generated each time a new diversified key is needed, the random nonce sent from the transmitting device 604 to the receiving device 608; the full value of a counter value sent from the transmitting device 604 and the receiving device 608; a portion of a counter value sent from the transmitting device 604 and the receiving device 608; a counter independently maintained by the transmitting device 604 and the receiving device 608 but not sent between the two devices; a one-time-passcode exchanged between the transmitting device 604 and the receiving device 608; and a cryptographic hash of the sensitive data. In some examples, one or more portions of the key diversification value may be used by the parties to create multiple diversified keys. For example, a counter may be used as the key diversification value. Further, a combination of one or more of the exemplary key diversification values described above may be used.

[0095] In another example, a portion of the counter may be used as the key diversification value. If multiple master key values are shared between the parties, the multiple diversified key values may be obtained by the systems and processes described herein. A new diversification value, and therefore a new diversified symmetric key, may be created as often as needed. In the most secure case, a new diversification value may be created for each exchange of sensitive data between the transmitting device 604 and the receiving device 608. In effect, this may create a one-time use key, such as a single-use session key.

[0096] FIG. 7 illustrates an example configuration of a contactless card 102, 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 702 on the front or back of the contactless card 102. In some examples, the contactless card 102 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 102 may include a substrate 708, 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 102 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 102 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.

[0097] The contactless card 102 may also include identification information 706 displayed on the front and/or back of the card, and a contact pad 704. The contact pad 704 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 102 may also include processing circuitry, antenna and other components as will be further discussed in FIG. 8. These components may be located behind the contact pad 704 or elsewhere on the substrate 708, e.g., within a different layer of the substrate 708, and may electrically and physically coupled with the contact pad 704. The contactless card 102 may also include a magnetic strip or tape, which may be located on the back of the card (not shown in FIG. 7). The contactless card 102 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.

[0098] As illustrated, the contact pad 704 of contactless card 102 may include processing circuitry 816 for storing, processing, and communicating information, including a processor 802, a memory 804, and one or more interface(s) 806. It is understood that the processing circuitry 816 may contain additional components, including processors, memories, error and parity/CRC checkers, data encoders, anticollision algorithms, controllers, command decoders, security primitives and tamper proofing hardware, as necessary to perform the functions described herein.

[0099] The memory 804 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 102 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 804 may be encrypted memory utilizing an encryption algorithm executed by the processor 802 to encrypted data.

[0100] The memory 804 may be configured to store one or more applet(s) 808, one or more counter(s) 810, a customer identifier 814, and the account number(s) 812, which may be virtual account numbers. The one or more applet(s) 808 may comprise one or more software applications configured to execute on one or more contactless cards, such as a Java® Card applet. However, it is understood that applet(s) 808 are not limited to Java Card applets, and instead may be any software application operable on contactless cards or other devices having limited memory. The one or more counter(s) 810 may comprise a numeric counter sufficient to store an integer. The customer identifier 814 may comprise a unique alphanumeric identifier assigned to a user of the contactless card 102, and the identifier may distinguish the user of the contactless card from other contactless card users. In some examples, the customer identifier 814 may identify both a customer and an account assigned to that customer and may further identify the contactless card 102 associated with the customer's account. As stated, the account number(s) 812 may include thousands of one-time use virtual account numbers associated with the contactless card 102. An applet(s) 808 of the contactless card 102 may be configured to manage the account number(s) 812 (e.g., to select an account number(s) 812, mark the selected account number(s) 812 as used, and transmit the account number(s) 812 to a mobile device for autofilling by an autofilling service.

[0101] The processor 802 and memory elements of the foregoing exemplary embodiments are described with reference to the contact pad 704, but the present disclosure is not limited thereto. It is understood that these elements may be implemented outside of the contact pad 704 or entirely separate from it, or as further elements in addition to processor 802 and memory 804 elements located within the contact pad 704.

[0102] In some examples, the contactless card 102 may comprise one or more antenna(s) 818. The one or more antenna(s) 818 may be placed within the contactless card 102 and around the processing circuitry 816 of the contact pad 704. For example, the one or more antenna(s) 818 may be integral with the processing circuitry 816 and the one or more antenna(s) 818 may be used with an external booster coil. As another example, the one or more antenna(s) 818 may be external to the contact pad 704 and the processing circuitry 816.

[0103] In an embodiment, the coil of contactless card 102 may act as the secondary of an air core transformer. The terminal may communicate with the contactless card 102 by cutting power or amplitude modulation. The contactless card 101 may infer the data transmitted from the terminal using the gaps in the contactless card's power connection, which may be functionally maintained through one or more capacitors. The contactless card 102 may communicate back by switching a load on the contactless card's coil or load modulation. Load modulation may be detected in the terminal's coil through interference. More generally, using the antenna(s) 818, processor 802, and/or the memory 804, the contactless card 101 provides a communications interface to communicate via NFC, Bluetooth, and/or Wi-Fi communications.

[0104] As explained above, contactless card 102 may be built on a software platform operable on smart cards or other devices having limited memory, such as JavaCard, and one or more or more applications or applets may be securely executed. Applet(s) 808 may be added to contactless cards to provide a one-time password (OTP) for multifactor authentication (MFA) in various mobile application-based use cases. Applet(s) 808 may be configured to respond to one or more requests, such as near field data exchange requests, from a reader, such as a mobile NFC reader (e.g., of a mobile device or point-of-sale terminal), and produce an NDEF message that comprises a cryptographically secure OTP encoded as an NDEF text tag.

[0105] One example of an NDEF OTP is an NDEF short-record layout (SR=1). In such an example, one or more applet(s) 808 may be configured to encode the OTP as an NDEF type 4 well known type text tag. In some examples, NDEF messages may comprise one or more records. The applet(s) 808 may be configured to add one or more static tag records in addition to the OTP record.

[0106] In some examples, the one or more applet(s) 808 may be configured to emulate an RFID tag. The RFID tag may include one or more polymorphic tags. In some examples, each time the tag is read, different cryptographic data is presented that may indicate the authenticity of the contactless card. Based on the one or more applet(s) 808, 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.

[0107] In some examples, the contactless card 102 and server may include certain data such that the card may be properly identified. The contactless card 102 may include one or more unique identifiers (not pictured). Each time a read operation takes place, the counter(s) 810 may be configured to increment. In some examples, each time data from the contactless card

102 is read (e.g., by a mobile device), the counter(s) 810 is transmitted to the server for validation and determines whether the counter(s) 810 are equal (as part of the validation) to a counter of the server.

[0108] The one or more counter(s) 810 may be configured to prevent a replay attack. For example, if a cryptogram has been obtained and replayed, that cryptogram is immediately rejected if the counter(s) 810 has been read or used or otherwise passed over. If the counter(s) 810 has not been used, it may be replayed. In some examples, the counter that is incremented on the card is different from the counter that is incremented for transactions. The contactless card 101 is unable to determine the application transaction counter(s) 810 since there is no communication between applet(s) 808 on the contactless card 102.

[0109] In some examples, the counter(s) 810 may get out of sync. In some examples, to account for accidental reads that initiate transactions, such as reading at an angle, the counter(s) 810 may increment but the application does not process the counter(s) 810. In some examples, when the mobile device 10 is woken up, NFC may be enabled and the client device 104 may be configured to read available tags, but no action is taken responsive to the reads.

[0110] To keep the counter(s) 810 in sync, an application, such as a background application, may be executed that would be configured to detect when the client device 104 wakes up and synchronize with the server 108 of a banking system indicating that a read that occurred due to detection to then move the counter 710 forward. In other examples, Hashed One Time Password may be utilized such that a window of mis-synchronization may be accepted. For example, if within a threshold of 10, the counter(s) 810 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(s) 810 increases in the appropriate sequence, then it possible to know that the user has done so.

[0111] The key diversification technique described herein with reference to the counter(s) 810, master key, and diversified key, is one example of encryption and/or decryption a key diversification technique. This example key diversification technique should not be considered limiting of the disclosure, as the disclosure is equally applicable to other types of key diversification techniques.

[0112] During the creation process of the contactless card 102, 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 102. 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.

[0113] In some examples, to overcome deficiencies of 3DES algorithms, which may be susceptible to vulnerabilities, a session key may be derived (such as a unique key per session) but rather than using the master key, the unique card-derived keys and the counter may be used as diversification data. For example, each time the contactless card 102 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).

[0114] 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.

[0115] 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.

[0116] FIG. 9 is a timing diagram illustrating an example sequence for providing authenticated access according to one or more embodiments of the present disclosure. Sequence flow 900 may include contactless card 102 and client device 104, which may include an application 902 and processor 904. The application 902 can be any of the applications App 204a, 204b, App 204c, or any other application that executes on the client device 104.

[0117] At line 908, the application 902 communicates with the contactless card 102 (e.g., after being brought near the contactless card 102). Communication between the application 902 and the contactless card 102 may involve the contactless card 102 being sufficiently close to a card reader (not shown) of the client device 104 to enable NFC data transfer between the application 902 and the contactless card 102.

[0118] At line 906, after communication has been established between client device 104 and contactless card 102, contactless card 102 generates a message authentication code (MAC) cryptogram. In some examples, this may occur when the contactless card 102 is read by the application 902. In particular, this may occur upon a read, such as an NFC read, of a near field data exchange (NDEF) tag, which may be created in accordance with the NFC Data Exchange Format. For example, a reader application, such as application 902, may transmit a message, such as an applet select message, with the applet ID of an NDEF producing applet. Upon confirmation of the selection, a sequence of select file messages followed by read file messages may be transmitted. For example, the sequence may include "Select Capabilities file", "Read Capabilities file", and "Select NDEF file". At this point, a counter value maintained by the contactless card 102 may be updated or incremented, which may be followed by "Read NDEF file." At this point, the message may be generated which may include a header and a shared secret. Session keys may then be generated. The MAC cryptogram may be created from the message, which may include the header and the shared secret. The MAC cryptogram may then be concatenated with one or more blocks of random data, and the MAC cryptogram and a random number (RND) may be encrypted with the session key. Thereafter, the cryptogram and the header may be concatenated, and encoded as ASCII hex and returned in NDEF message format (responsive to the "Read NDEF file" message).

[0119] In some examples, the MAC cryptogram may be transmitted as an NDEF tag, and in other examples the MAC cryptogram may be included with a uniform resource indicator (e.g., as a formatted string). In some examples, application 902 may be configured to transmit a request to contactless card 102, the request comprising an instruction to generate a MAC cryptogram.

[0120] At line 910, the contactless card 102 sends the MAC cryptogram to the application 902. In some examples, the transmission of the MAC cryptogram occurs via NFC, however, the present disclosure is not limited thereto. In other examples, this communication may occur

via Bluetooth, Wi-Fi, or other means of wireless data communication. At line 912, the application 902 communicates the MAC cryptogram to the processor 904.

[0121] At line 914, the processor 904 verifies the MAC cryptogram pursuant to an instruction from the application 122. For example, the MAC cryptogram may be verified, as explained below. In some examples, verifying the MAC cryptogram may be performed by a device other than client device 104, such as a server of a banking system in data communication with the client device 104. For example, processor 904 may output the MAC cryptogram for transmission to the server of the banking system, which may verify the MAC cryptogram. In some examples, the MAC cryptogram may function as a digital signature for purposes of verification. Other digital signature algorithms, such as public key asymmetric algorithms, e.g., the Digital Signature Algorithm and the RSA algorithm, or zero knowledge protocols, may be used to perform this verification.

[0122] FIG. 10 illustrates an NDEF short-record layout (SR=1) data structure 1000 according to an example embodiment. One or more applets 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 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.

[0123] FIG. 11 illustrates a diagram of a system 1100 configured to implement one or more embodiments of the present disclosure. As explained below, during the contactless card creation process, two cryptographic keys may be assigned uniquely for each card. The cryptographic keys may comprise symmetric keys which may be used in both encryption and decryption of data. Triple DES (3DES) algorithm may be used by EMV and it is implemented by hardware in the contactless card. By using a key diversification process, one or more keys may be derived from a master key based upon uniquely identifiable information for each entity that requires a key.

[0124] Regarding master key management, two issuer master keys 1102, 1126 may be required for each part of the portfolio on which the one or more applets is issued. For example,

the first master key 1102 may comprise an Issuer Cryptogram Generation/Authentication Key (Iss-Key-Auth) and the second master key 1126 may comprise an Issuer Data Encryption Key (Iss-Key-DEK). As further explained herein, two issuer master keys 1102, 1126 are diversified into card master keys 1108, 1120, which are unique for each card. In some examples, a network profile record ID (pNPR) 522 and derivation key index (pDKI) 1124, as back office data, may be used to identify which Issuer Master Keys 1102, 1126 to use in the cryptographic processes for authentication. The system performing the authentication may be configured to retrieve values of pNPR 1122 and pDKI 1124 for a contactless card at the time of authentication.

[0125] In some examples, to increase the security of the solution, a session key may be derived (such as a unique key per session) but rather than using the master key, the unique card-derived keys and the counter may be used as diversification data, as explained above. For example, each time the card is used in operation, a different key may be used for creating the message authentication code (MAC) and for performing the encryption. Regarding session key generation, the keys used to generate the cryptogram and encipher the data in the one or more applets may comprise session keys based on the card unique keys (Card-Key-Auth 1108 and Card-Key-Dek 1120). The session keys (Aut-Session-Key 1130 and DEK-Session-Key 1110) may be generated by the one or more applets and derived by using the application transaction counter (pATC) 1104 with one or more algorithms. To fit data into the one or more algorithms, only the 2 low order bytes of the 4-byte pATC 1104 is used. In some examples, the four byte session key derivation method may comprise: F1 := PATC(lower 2 bytes) || 'F0' || '00' || PATC (four bytes) F1 := PATC(lower 2 bytes) || '0F' || '00' || PATC (four bytes) SK :={(ALG (MK) [F1]) || ALG (MK) [F2] }, where ALG may include 3DES ECB and MK may include the card unique derived master key.

[0126] As described herein, one or more MAC session keys may be derived using the lower two bytes of pATC 1104 counter. At each tap of the contactless card, pATC 1104 is configured to be updated, and the card master keys Card-Key-AUTH 508 and Card-Key-DEK 1120 are further diversified into the session keys Aut-Session-Key 1130 and DEK-Session-KEY 1110. pATC 1104 may be initialized to zero at personalization or applet initialization time. In some examples, the pATC counter 1104 may be initialized at or before personalization, and may be configured to increment by one at each NDEF read.

[0127] Further, the update for each card may be unique, and assigned either by personalization, or algorithmically assigned by pUID or other identifying information. For example, odd numbered cards may increment or decrement by 2 and even numbered cards may increment or decrement by 5. In some examples, the update may also vary in sequential reads, such that one card may increment in sequence by 1, 3, 5, 2, 2, ... repeating. The specific sequence or algorithmic sequence may be defined at personalization time, or from one or more processes derived from unique identifiers. This can make it harder for a replay attacker to generalize from a small number of card instances.

[0128] The authentication message may be delivered as the content of a text NDEF record in hexadecimal ASCII format. In some examples, only the authentication data and an 8-byte random number followed by MAC of the authentication data may be included. In some examples, the random number may precede cryptogram A and may be one block long. In other examples, there may be no restriction on the length of the random number. In further examples, the total data (i.e., the random number plus the cryptogram) may be a multiple of the block size. In these examples, an additional 8-byte block may be added to match the block produced by the MAC algorithm. As another example, if the algorithms employed used 16-byte blocks, even multiples of that block size may be used, or the output may be automatically, or manually, padded to a multiple of that block size.

[0129] The MAC may be performed by a function key (AUT-Session-Key) 1130. The data specified in cryptogram may be processed with javacard.signature method: ALG_DES_MAC8_ISO9797_1_M2_ALG3 to correlate to EMV ARQC verification methods. The key used for this computation may comprise a session key AUT-Session-Key 1130, as explained above. As explained above, the low order two bytes of the counter may be used to diversify for the one or more MAC session keys. As explained below, AUT-Session-Key 1130 may be used to MAC data 1106, and the resulting data or cryptogram An 1114 and random number RND may be encrypted using DEK-Session-Key 1110 to create cryptogram B or output 1118 sent in the message.

[0130] In some examples, one or more HSM commands may be processed for decrypting such that the final 16 (binary, 32 hex) bytes may comprise a 3DES symmetric encrypting using CBC mode with a zero IV of the random number followed by MAC authentication data. The key used for this encryption may comprise a session key DEK-Session-Key 1110 derived from the

Card-Key-DEK 1120. In this case, the ATC value for the session key derivation is the least significant byte of the counter pATC 1104.

[0131] The format below represents a binary version example embodiment. Further, in some examples, the first byte may be set to ASCII 'A'.

Message Format				
1	2	4	8	8
0x43 (Message Type 'A')	Version	pATC	RND	Cryptogram A (MAC)

Cryptogram A (MAC)	**8 bytes**	
MAC of		

2	8	4	4	18 bytes input data
Version	pUID	pATC	Shared Secret	

Message Format			
1	2	4	16
0x43 (Message Type 'A')	Version	pATC	Cryptogram B

Cryptogram A (MAC)	**8 bytes**	
MAC of		

2	8	4	4	18 bytes input data
Version	pUID	pATC	Shared Secret	

Cryptogram B	**16**		
Sym Encryption of			
8	8		

RND	Cryptogram A		

[0001] Another exemplary format is shown below. In this example, the tag may be encoded in hexadecimal format.

Message Format				
2	8	4	8	8
Version	pUID	pATC	RND	Cryptogram A (MAC)
8 bytes				
8	8	4	4	18 bytes input data
pUID	pUID	pATC	Shared Secret	

Message Format				
2	8	4	16	
Version	pUID	pATC	Cryptogram B	
8 bytes				
8		4	4	18 bytes input data
pUID	pUID	pATC	Shared Secret	
Cryptogram B	**16**			
Sym Encryption of				
8	8			
RND	Cryptogram A			

[0132] The UID field of the received message may be extracted to derive, from master keys Iss-Key-AUTH 905 and Iss-Key-DEK 910, the card master keys (Card-Key-Auth 925 and Card-Key-DEK 930) for that particular card. Using the card master keys (Card-Key-Auth 508 and Card-Key-DEK 1120), the counter (pATC) field of the received message may be used to derive the session keys (Aut-Session-Key 1130 and DEK-Session-Key 1110) for that particular card. Cryptogram B 1118 may be decrypted using the DEK-Session-KEY, which yields cryptogram An 1114 and RND, and RND may be discarded. The UID field may be used to look up the shared secret of the contactless card which, along with the Ver, UID, and pATC fields of the message, may be processed through the cryptographic MAC using the re-created Aut-Session-Key to create a MAC output, such as MAC'. If MAC' is the same as cryptogram An 1114, then this indicates that the message decryption and MAC checking have all passed. Then the pATC may be read to determine if it is valid.

[0133] During an authentication session, one or more cryptograms may be generated by the one or more applications. For example, the one or more cryptograms may be generated as a 3DES MAC using ISO 9797-1 Algorithm 3 with Method 2 padding via one or more session keys, such as Aut-Session-Key 1130. The input data 1106 may take the following form: Version (2), pUID (8), pATC (4), Shared Secret (4). In some examples, the numbers in the brackets may comprise length in bytes. In some examples, the shared secret may be generated by one or more random number generators which may be configured to ensure, through one or more secure processes, that the random number is unpredictable. In some examples, the shared secret may comprise a random 4-byte binary number injected into the card at personalization time that is known by the authentication service. During an authentication session, the shared secret may not be provided from the one or more applets to the mobile application. Method 2 padding may include adding a mandatory 0x'80' byte to the end of input data and 0x'00' bytes that may be added to the end of the resulting data up to the 8-byte boundary. The resulting cryptogram may comprise 8 bytes in length.

[0134] In some examples, one benefit of encrypting an unshared random number as the first block with the MAC cryptogram, is that it acts as an initialization vector while using CBC (Block chaining) mode of the symmetric encryption algorithm. This allows the "scrambling" from block to block without having to pre-establish either a fixed or dynamic IV.

[0135] By including the application transaction counter (pATC) as part of the data included in the MAC cryptogram, the authentication service may be configured to determine if the value

conveyed in the clear data has been tampered with. Moreover, by including the version in the one or more cryptograms, it is difficult for an attacker to purposefully misrepresent the application version in an attempt to downgrade the strength of the cryptographic solution. In some examples, the pATC may start at zero and be updated by 1 each time the one or more applications generates authentication data. The authentication service may be configured to track the pATCs used during authentication sessions. In some examples, when the authentication data uses a pATC equal to or lower than the previous value received by the authentication service, this may be interpreted as an attempt to replay an old message, and the authenticated may be rejected. In some examples, where the pATC is greater than the previous value received, this may be evaluated to determine if it is within an acceptable range or threshold, and if it exceeds or is outside the range or threshold, verification may be deemed to have failed or be unreliable. In the MAC operation 1112, data 1106 is processed through the MAC using Aut-Session-Key 1130 to produce MAC output (cryptogram A) 1114, which is encrypted.

[0136] In order to provide additional protection against brute force attacks exposing the keys on the card, it is desirable that the MAC cryptogram 1114 be enciphered. In some examples, data or cryptogram An 1114 to be included in the ciphertext may comprise: Random number (8), cryptogram (8). In some examples, the numbers in the brackets may comprise length in bytes. In some examples, the random number may be generated by one or more random number generators which may be configured to ensure, through one or more secure processes, that the random number is unpredictable. The key used to encipher this data may comprise a session key. For example, the session key may comprise DEK-Session-Key 1110. In the encryption operation 1116, data or cryptogram An 1114 and RND are processed using DEK-Session-Key 510 to produce encrypted data, cryptogram B 1118. The data 1114 may be enciphered using 3DES in cipher block chaining mode to ensure that an attacker must run any attacks over all of the ciphertext. As a non-limiting example, other algorithms, such as Advanced Encryption Standard (AES), may be used. In some examples, an initialization vector of 0x'0000000000000000' may be used. Any attacker seeking to brute force the key used for enciphering this data will be unable to determine when the correct key has been used, as correctly decrypted data will be indistinguishable from incorrectly decrypted data due to its random appearance.

[0137] In order for the authentication service to validate the one or more cryptograms provided by the one or more applets, the following data must be conveyed from the one or more applets to the mobile device in the clear during an authentication session: version number to determine the cryptographic approach used and message format for validation of the cryptogram, which enables the approach to change in the future; pUID to retrieve cryptographic assets, and derive the card keys; and pATC to derive the session key used for the cryptogram.

[0138] FIG. 12 illustrates a method 1200 for generating a cryptogram. For example, at block 1202, a network profile record ID (pNPR) and derivation key index (pDKI) may be used to identify which Issuer Master Keys to use in the cryptographic processes for authentication. In some examples, the method may include performing the authentication to retrieve values of pNPR and pDKI for a contactless card at the time of authentication.

[0139] At block 1204, Issuer Master Keys may be diversified by combining them with the card's unique ID number (pUID) and the PAN sequence number (PSN) of one or more applets, for example, a payment applet.

[0140] At block 1206, Card-Key-Auth and Card-Key-DEK (unique card keys) may be created by diversifying the Issuer Master Keys to generate session keys which may be used to generate a MAC cryptogram.

[0141] At block 1208, the keys used to generate the cryptogram and encipher the data in the one or more applets may comprise the session keys of block 1030 based on the card unique keys (Card-Key-Auth and Card-Key-DEK). In some examples, these session keys may be generated by the one or more applets and derived by using pATC, resulting in session keys Aut-Session-Key and DEK-Session-Key.

[0142] FIG. 13 depicts an exemplary process 1300 illustrating key diversification according to one example. Initially, a sender and the recipient may be provisioned with two different master keys. For example, a first master key may comprise the data encryption master key, and a second master key may comprise the data integrity master key. The sender has a counter value, which may be updated at block 1302, and other data, such as data to be protected, which it may secure share with the recipient.

[0143] At block 1304, the counter value may be encrypted by the sender using the data encryption master key to produce the data encryption derived session key, and the counter value may also be encrypted by the sender using the data integrity master key to produce the

data integrity derived session key. In some examples, a whole counter value or a portion of the counter value may be used during both encryptions.

[0144] In some examples, the counter value may not be encrypted. In these examples, the counter may be transmitted between the sender and the recipient in the clear, i.e., without encryption.

[0145] At block 1306, the data to be protected is processed with a cryptographic MAC operation by the sender using the data integrity session key and a cryptographic MAC algorithm. The protected data, including plaintext and shared secret, may be used to produce a MAC using one of the session keys (AUT-Session-Key).

[0146] At block 1308, the data to be protected may be encrypted by the sender using the data encryption derived session key in conjunction with a symmetric encryption algorithm. In some examples, the MAC is combined with an equal amount of random data, for example each 8 bytes long, and then encrypted using the second session key (DEK-Session-Key).

[0147] At block 1310, the encrypted MAC is transmitted, from the sender to the recipient, with sufficient information to identify additional secret information (such as shared secret, master keys, etc.), for verification of the cryptogram.

[0148] At block 1312, the recipient uses the received counter value to independently derive the two derived session keys from the two master keys as explained above.

[0149] At block 1314, the data encryption derived session key is used in conjunction with the symmetric decryption operation to decrypt the protected data. Additional processing on the exchanged data will then occur. In some examples, after the MAC is extracted, it is desirable to reproduce and match the MAC. For example, when verifying the cryptogram, it may be decrypted using appropriately generated session keys. The protected data may be reconstructed for verification. A MAC operation may be performed using an appropriately generated session key to determine if it matches the decrypted MAC. As the MAC operation is an irreversible process, the only way to verify is to attempt to recreate it from source data.

[0150] At block 1316, the data integrity derived session key is used in conjunction with the cryptographic MAC operation to verify that the protected data has not been modified.

[0151] Some examples of the methods described herein may advantageously confirm when a successful authentication is determined when the following conditions are met. First, the ability to verify the MAC shows that the derived session key was proper. The MAC may only

be correct if the decryption was successful and yielded the proper MAC value. The successful decryption may show that the correctly derived encryption key was used to decrypt the encrypted MAC. Since the derived session keys are created using the master keys known only to the sender (e.g., the transmitting device) and recipient (e.g., the receiving device), it may be trusted that the contactless card which originally created the MAC and encrypted the MAC is indeed authentic. Moreover, the counter value used to derive the first and second session keys may be shown to be valid and may be used to perform authentication operations.

[0152] Thereafter, the two derived session keys may be discarded, and the next iteration of data exchange will update the counter value (returning to block 1302) and a new set of session keys may be created (at block 1310). In some examples, the combined random data may be discarded.

[0153] FIG. 14 illustrates a method 1400 for card activation according to an example embodiment. For example, card activation may be completed by a system including a card, a device, and one or more servers. The contactless card, device, and one or more servers may reference same or similar components that were previously explained, such as contactless card 102, client device 104, and server 108.

[0154] In block 1402 , the card may be configured to dynamically generate data. In some examples, this data may include information such as an account number, card identifier, card verification value, or phone number, which may be transmitted from the card to the device. In some examples, one or more portions of the data may be encrypted via the systems and methods disclosed herein.

[0155] In block 1404, one or more portions of the dynamically generated data may be communicated to an application of the device via NFC or other wireless communication. For example, a tap of the card proximate to the device may allow the application of the device to read the one or more portions of the data associated with the contactless card. In some examples, if the device does not comprise an application to assist in activation of the card, the tap of the card may direct the device or prompt the customer to a software application store to download an associated application to activate the card. In some examples, the user may be prompted to sufficiently gesture, place, or orient the card towards a surface of the device, such as either at an angle or flatly placed on, near, or proximate the surface of the device. Responsive to a sufficient gesture, placement and/or orientation of the card, the device may

proceed to transmit the one or more encrypted portions of data received from the card to the one or more servers.

[0156] In block 1406, the one or more portions of the data may be communicated to one or more servers, such as a card issuer server. For example, one or more encrypted portions of the data may be transmitted from the device to the card issuer server for activation of the card.

[0157] In block 1408, the one or more servers may decrypt the one or more encrypted portions of the data via the systems and methods disclosed herein. For example, the one or more servers may receive the encrypted data from the device and may decrypt it in order to compare the received data to record data accessible to the one or more servers. If a resulting comparison of the one or more decrypted portions of the data by the one or more servers yields a successful match, the card may be activated. If the resulting comparison of the one or more decrypted portions of the data by the one or more servers yields an unsuccessful match, one or more processes may take place. For example, responsive to the determination of the unsuccessful match, the user may be prompted to tap, swipe, or wave gesture the card again. In this case, there may be a predetermined threshold comprising a number of attempts that the user is permitted to activate the card. Alternatively, the user may receive a notification, such as a message on his or her device indicative of the unsuccessful attempt of card verification and to call, email or text an associated service for assistance to activate the card, or another notification, such as a phone call on his or her device indicative of the unsuccessful attempt of card verification and to call, email or text an associated service for assistance to activate the card, or another notification, such as an email indicative of the unsuccessful attempt of card verification and to call, email or text an associated service for assistance to activate the card.

[0158] In block 1410, the one or more servers may transmit a return message based on the successful activation of the card. For example, the device may be configured to receive output from the one or more servers indicative of a successful activation of the card by the one or more servers. The device may be configured to display a message indicating successful activation of the card. Once the card has been activated, the card may be configured to discontinue dynamically generating data so as to avoid fraudulent use. In this manner, the card may not be activated thereafter, and the one or more servers are notified that the card has already been activated.

[0159] The various elements of the devices as previously described with reference to figures herein may include various hardware elements, software elements, or a combination of both.

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

[0160] 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, writable or rewritable media, digital or

analog media, hard disk, floppy disk, Compact Disk Read Only Memory (CD-ROM), Compact Disk Recordable (CD-R), Compact Disk Rewritable (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.

[0161] 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."

[0162] 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 necessarily be divided, omitted, or included in embodiments.

[0163] 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.

[0164] 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.

[0165] 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.

[0166] 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.

[0167] 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 computer-implemented method, comprising:

receiving a first request to execute a computer-executable instruction, the computer-executable instruction requests data from an enterprise server;

receiving, from the enterprise server, a second request to authenticate the first request;

scanning, based on the second request, using a short-range wireless communication module of the computing device, authentication from a contactless card;

in response to determining that the first request is authentic based on the authentication data, transmitting a response to the enterprise server causing the enterprise server to send the data; and

executing, via a web-browser, the computer-executable instruction from the first request in response to receiving the data sent by the enterprise server.

2. The computer-implemented method of claim 1, wherein receiving the second requests causes the web-browser to execute a predetermined computer program from a group of computer programs associated with the web-browser, the group of computer programs comprising an extension, a plugin, a component, and an addon.

3. The computer-implemented method of claim 2, wherein, the predetermined computer program is associated with the enterprise server.

4. The computer-implemented method of claim 1, wherein, to authenticate the first request comprises verification that the first request was initiated by an authorized user.

5. The computer-implemented method of claim 1, wherein, determining that the first request is authentic based on the authentication information comprises: validating a cryptogram by the predetermined computer program, and, in response, transmitting, by a predetermined computer program, a response to the enterprise server causing the enterprise server to send the data for the computer-executable instruction.

6. The computer-implemented method of claim 1, wherein determining that the first request is authentic based on the authentication data comprises: transmitting at least a cryptogram by a

predetermined computer program, for receipt by the enterprise server, and, in response to validating the cryptogram by the enterprise server, sending, by the enterprise server, the data for the computer-executable instruction.

7. The computer-implemented method of claim 1, wherein, the computer-executable instruction is part of a secure transaction being performed via the web-browser on a webpage.

8. The computer-implemented method of claim 7, wherein, the secure transaction comprises at least one from a group of transactions comprising a login transaction, a commercial transaction, and a data transfer transaction.

9. The computer-implemented method of claim 1, wherein, the enterprise server is at least one from a group of servers comprising a bank server, an authentication server, and an intermediate server.

10. A computing apparatus comprising:

a processor; and

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

receive a first request to execute a computer-executable instruction, the computer-executable instruction requests data from an enterprise server;

receive, from the enterprise server, a second request to authenticate the first request;

scan, based on the second request, using a short-range wireless communication module of the computing device, authentication from a contactless card;

in response to determining that the first request is authentic based on the authentication data, transmit a response to the enterprise server causing the enterprise server to send the data; and

execute, via a web-browser, the computer-executable instruction from the first request in response to receiving the data sent by the enterprise server.

11. The computing apparatus of claim 10, wherein receiving the second requests causes the web-browser to execute a predetermined computer program from a group of computer programs associated with the web-browser, the group of computer programs comprising an extension, a plugin, a component, and an addon.

12. The computing apparatus of claim 11, wherein, the predetermined computer program is associated with the enterprise server.

13. The computing apparatus of claim 10, wherein, to authenticate the first request comprises verification that the first request was initiated by an authorized user.

14. The computing apparatus of claim 10, wherein, determining that the first request is authentic based on the authentication information comprises: validate a cryptogram by the predetermined computer program, and, in response, transmitting, by a predetermined computer program, a response to the enterprise server causing the enterprise server to send the data for the computer-executable instruction.

15. The computing apparatus of claim 10, wherein determining that the first request is authentic based on the authentication data comprises: transmit at least a cryptogram by a predetermined computer program, for receipt by the enterprise server, and, in response to validating the cryptogram by the enterprise server, sending, by the enterprise server, the data for the computer-executable instruction.

16. The computing apparatus of claim 10, wherein, the computer-executable instruction is part of a secure transaction being performed via the web-browser on a webpage.

17. The computing apparatus of claim 16, wherein, the secure transaction comprises at least one from a group of transactions comprising a login transaction, a commercial transaction, and a data transfer transaction.

18. The computing apparatus of claim 10, wherein, the enterprise server is at least one from a group of servers comprising a bank server, an authentication server, and an intermediate server.

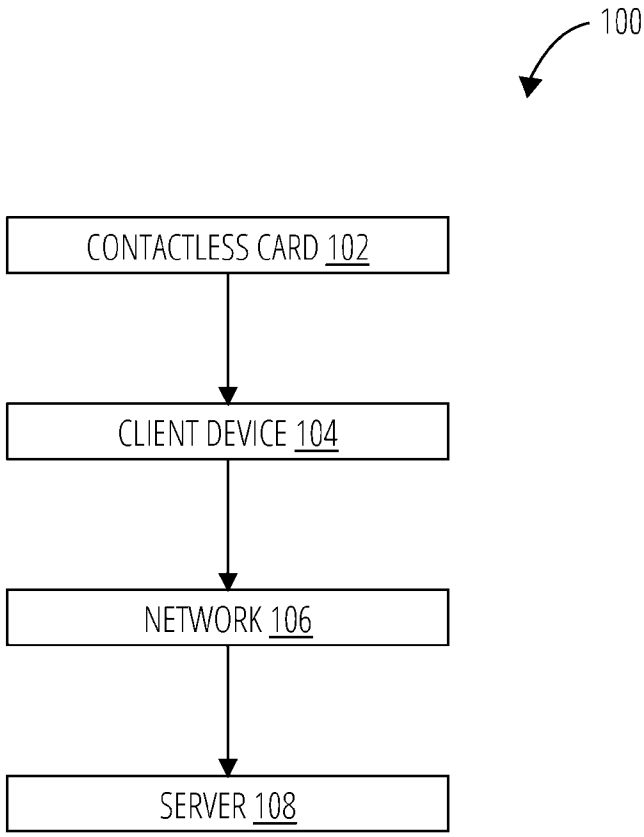
100
CONTACTLESS CARD 102
CLIENT DEVICE 104
NETWORK 106
SERVER 108

FIG. 1

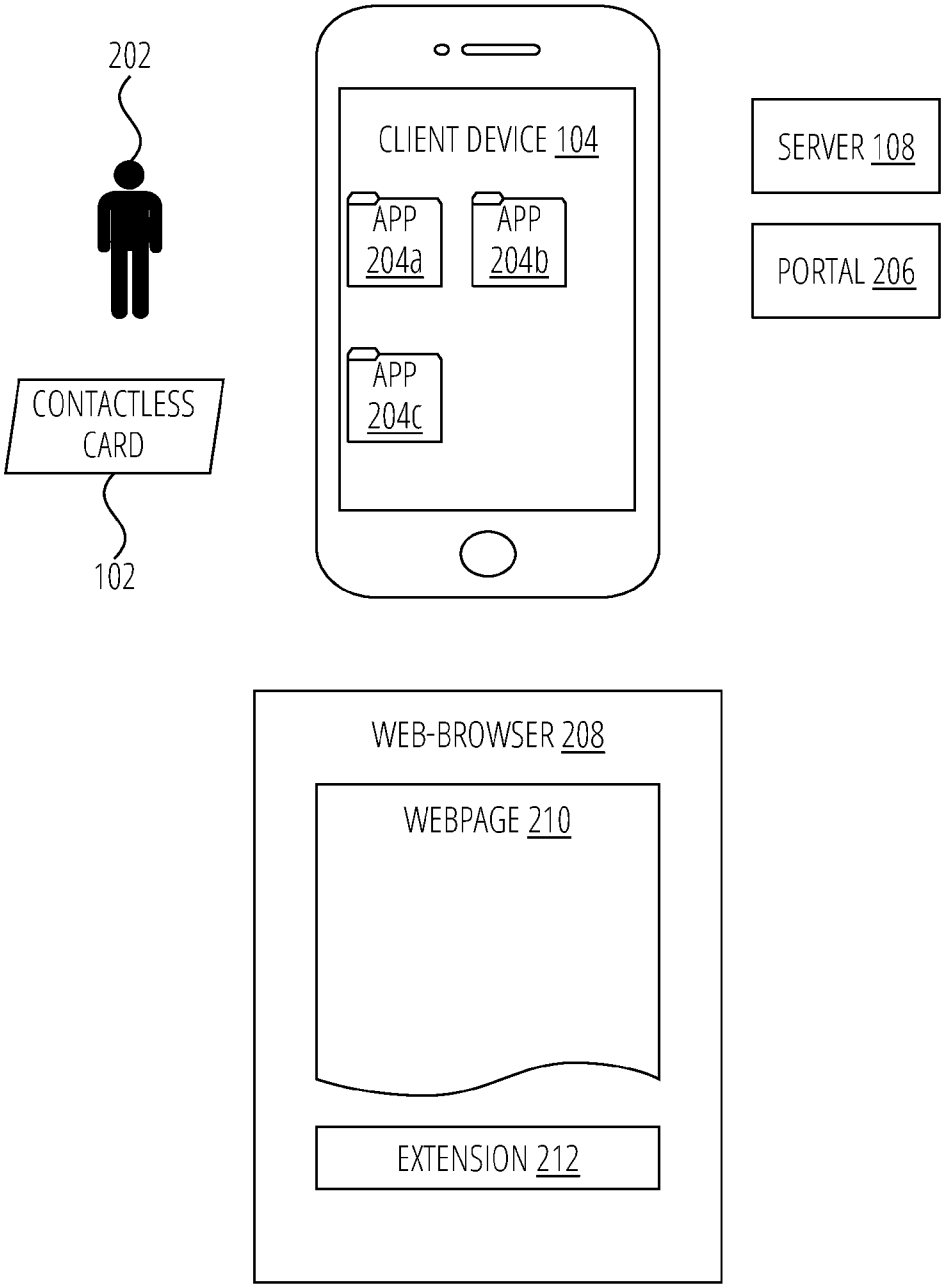
202
CONTACTLESS CARD
102
CLIENT DEVICE 104
APP 204a
APP 204b
APP 204c
SERVER 108
PORTAL 206
WEB-BROWSER 208
WEBPAGE 210
EXTENSION 212

FIG. 2

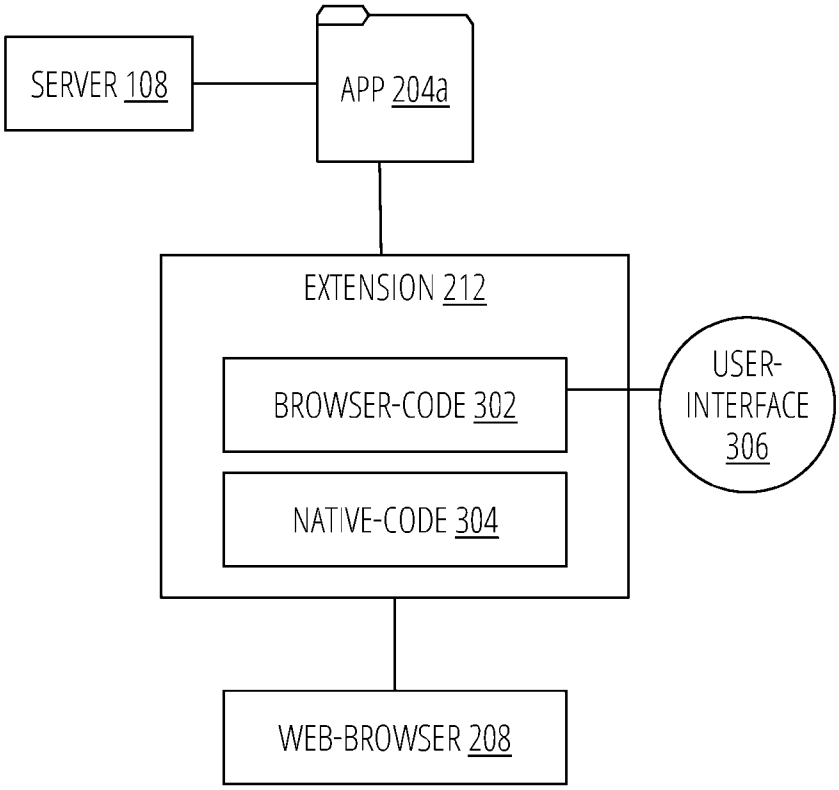
SERVER 108
APP 204a
EXTENSION 212
BROWSER-CODE 302
NATIVE-CODE 304
USER-INTERFACE 306
WEB-BROWSER 208

FIG. 3

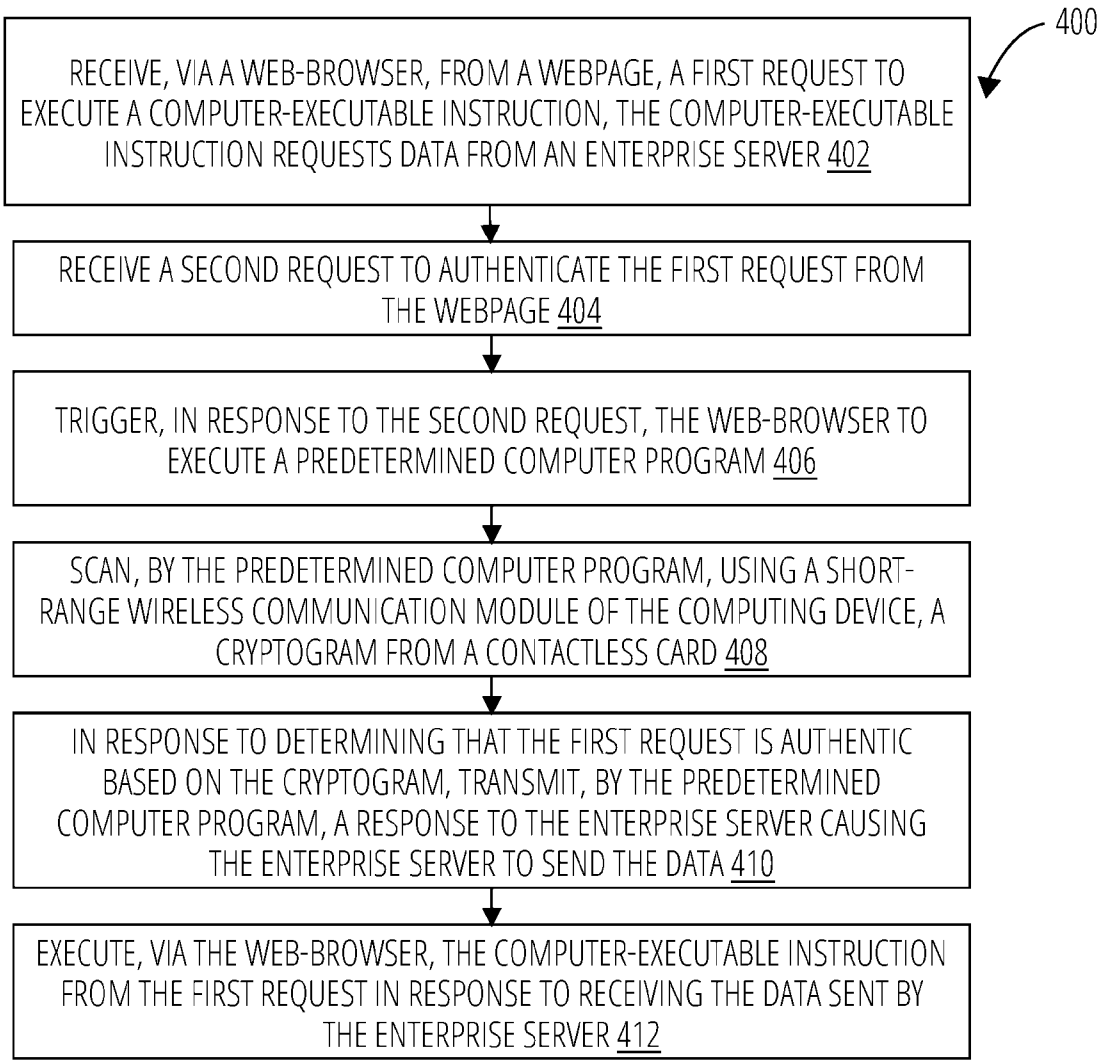
400
RECEIVE, VIA A WEB-BROWSER, FROM A WEBPAGE, A FIRST REQUEST TO EXECUTE A COMPUTER-EXECUTABLE INSTRUCTION, THE COMPUTER-EXECUTABLE INSTRUCTION REQUESTS DATA FROM AN ENTERPRISE SERVER 402
RECEIVE A SECOND REQUEST TO AUTHENTICATE THE FIRST REQUEST FROM THE WEBPAGE 404
TRIGGER, IN RESPONSE TO THE SECOND REQUEST, THE WEB-BROWSER TO EXECUTE A PREDETERMINED COMPUTER PROGRAM 406
SCAN, BY THE PREDETERMINED COMPUTER PROGRAM, USING A SHORT-RANGE WIRELESS COMMUNICATION MODULE OF THE COMPUTING DEVICE, A CRYPTOGRAM FROM A CONTACTLESS CARD 408
IN RESPONSE TO DETERMINING THAT THE FIRST REQUEST IS AUTHENTIC BASED ON THE CRYPTOGRAM, TRANSMIT, BY THE PREDETERMINED COMPUTER PROGRAM, A RESPONSE TO THE ENTERPRISE SERVER CAUSING THE ENTERPRISE SERVER TO SEND THE DATA 410
EXECUTE, VIA THE WEB-BROWSER, THE COMPUTER-EXECUTABLE INSTRUCTION FROM THE FIRST REQUEST IN RESPONSE TO RECEIVING THE DATA SENT BY THE ENTERPRISE SERVER 412

FIG. 4

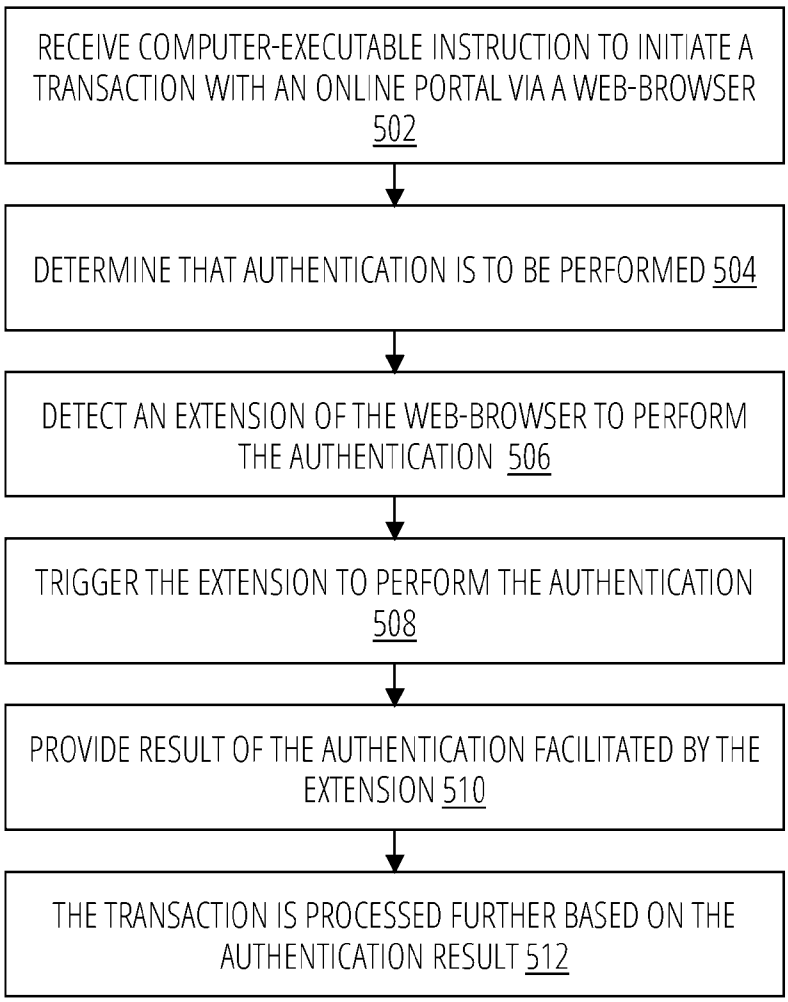
RECEIVE COMPUTER-EXECUTABLE INSTRUCTION TO INITIATE A TRANSACTION WITH AN ONLINE PORTAL VIA A WEB-BROWSER 502
DETERMINE THAT AUTHENTICATION IS TO BE PERFORMED 504
DETECT AN EXTENSION OF THE WEB-BROWSER TO PERFORM THE AUTHENTICATION 506
TRIGGER THE EXTENSION TO PERFORM THE AUTHENTICATION 508
PROVIDE RESULT OF THE AUTHENTICATION FACILITATED BY THE EXTENSION 510
THE TRANSACTION IS PROCESSED FURTHER BASED ON THE AUTHENTICATION RESULT 512

FIG. 5

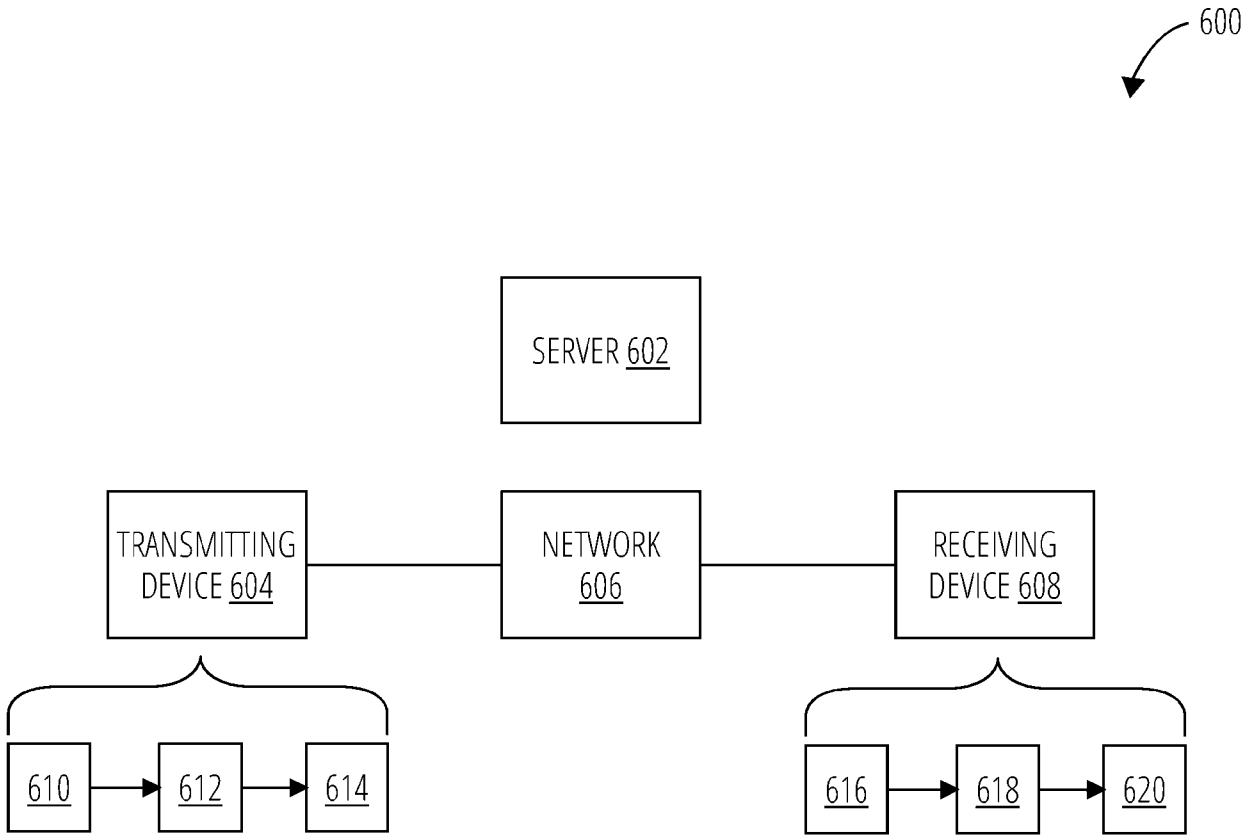
600
SERVER 602
TRANSMITTING DEVICE 604
NETWORK 606
RECEIVING DEVICE 608
610
612
614
616
618
620

FIG. 6

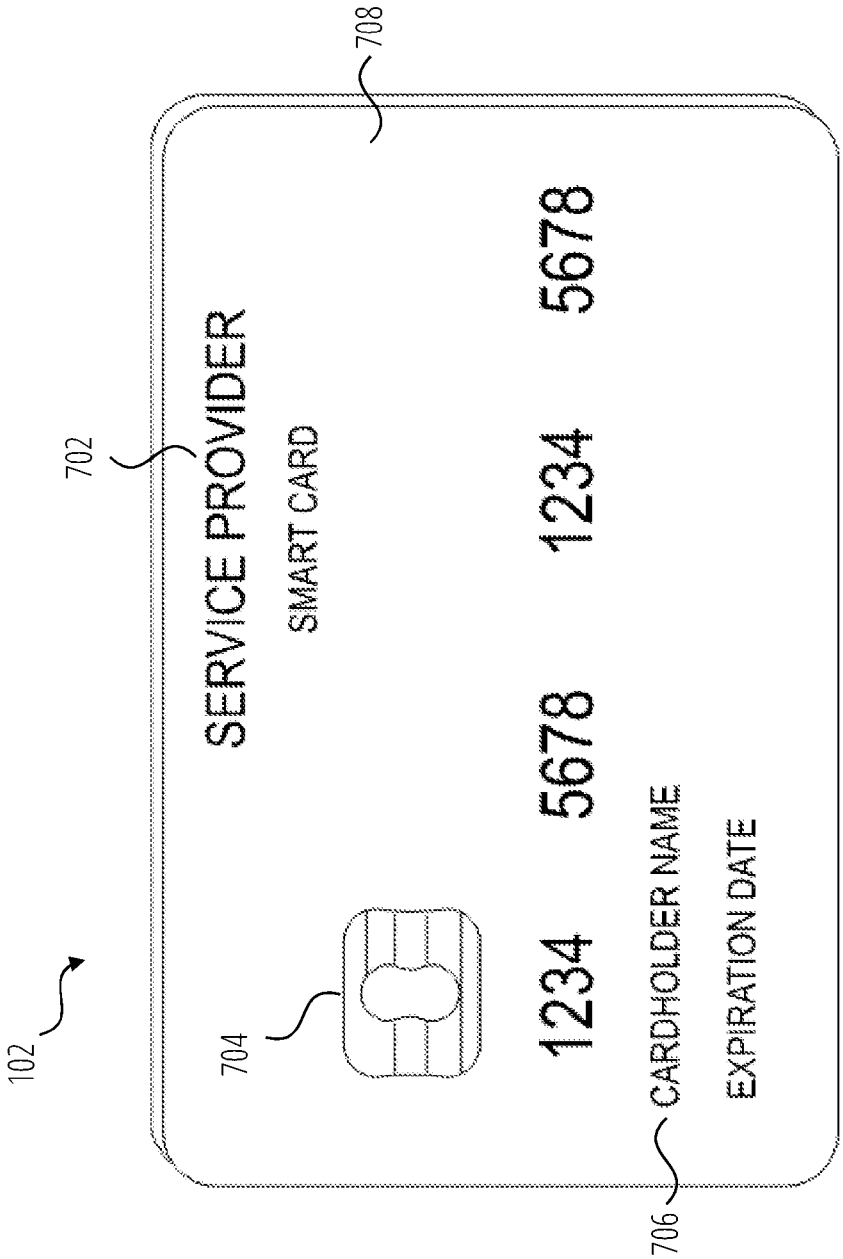
102
704
702
SERVICE PROVIDER
SMART CARD
1234 5678 1234 5678
CARDHOLDER NAME
EXPIRATION DATE
706
708

FIG. 7

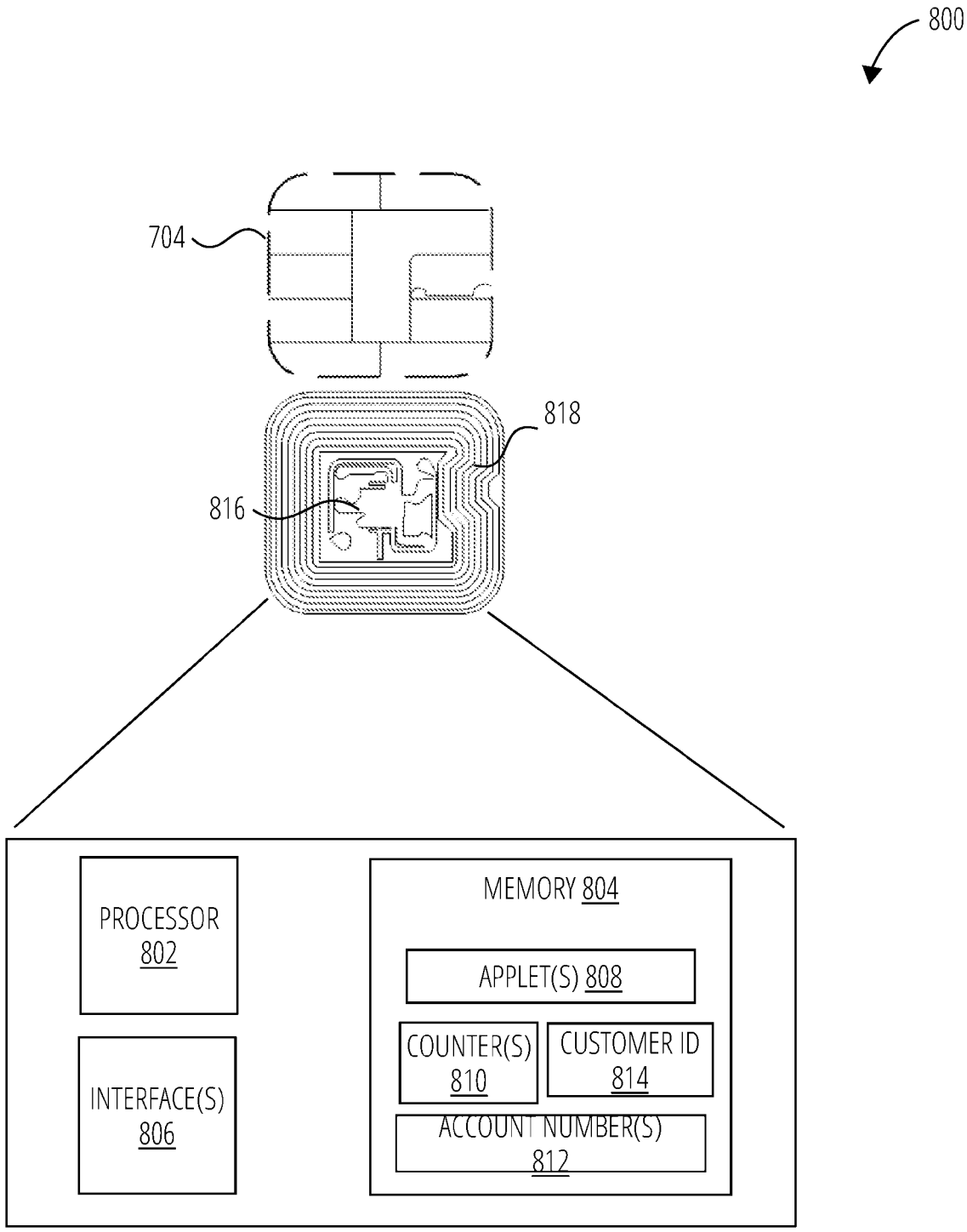
800
704
818
816
PROCESSOR 802
INTERFACE(S) 806
MEMORY 804
APPLET(S) 808
COUNTER(S) 810
CUSTOMER ID 814
ACCOUNT NUMBER(S) 812

FIG. 8

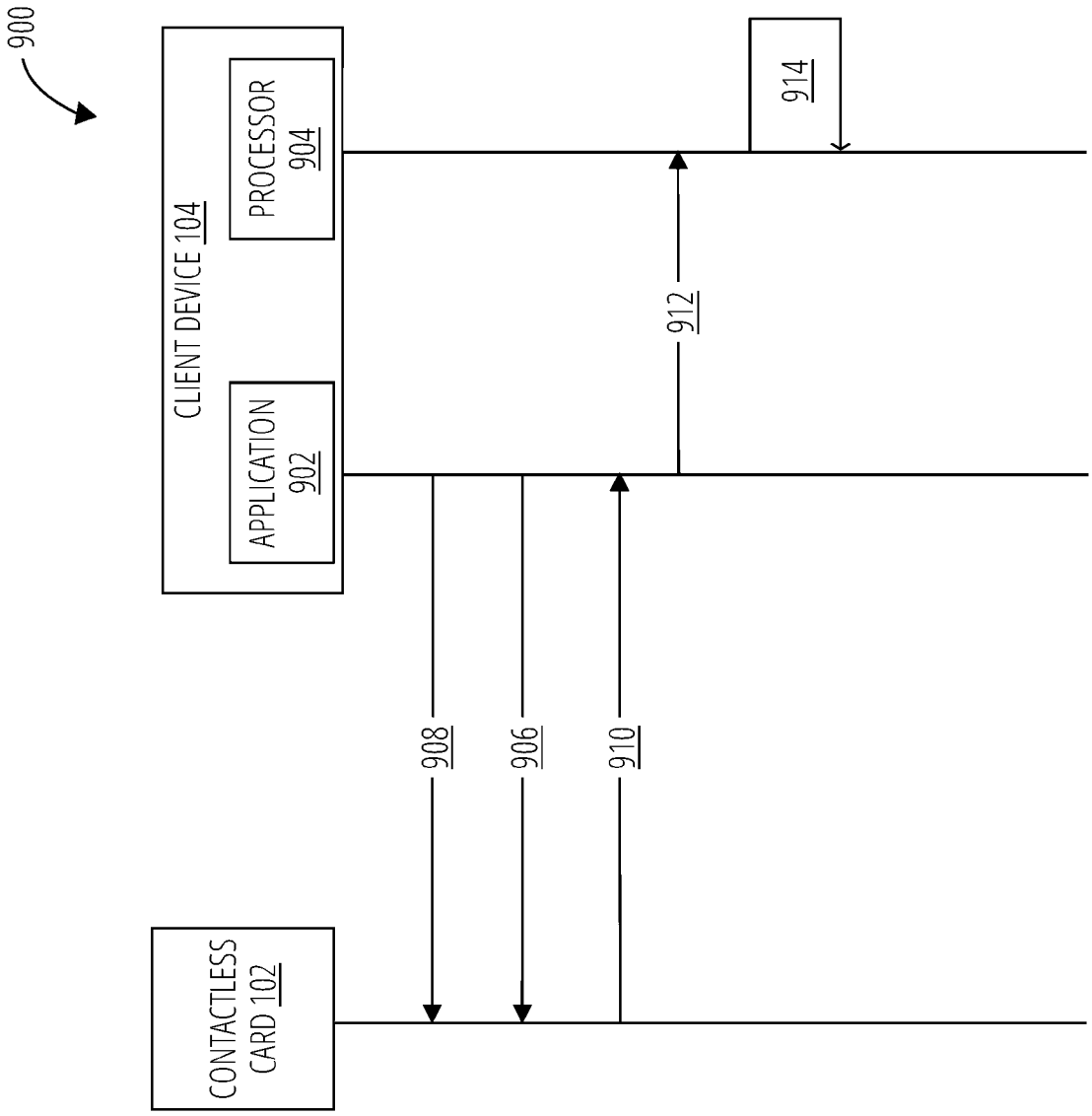
900
CLIENT DEVICE 104
PROCESSOR 904
APPLICATION 902
914
912
908
906
910
CONTACTLESS CARD 102

FIG. 9

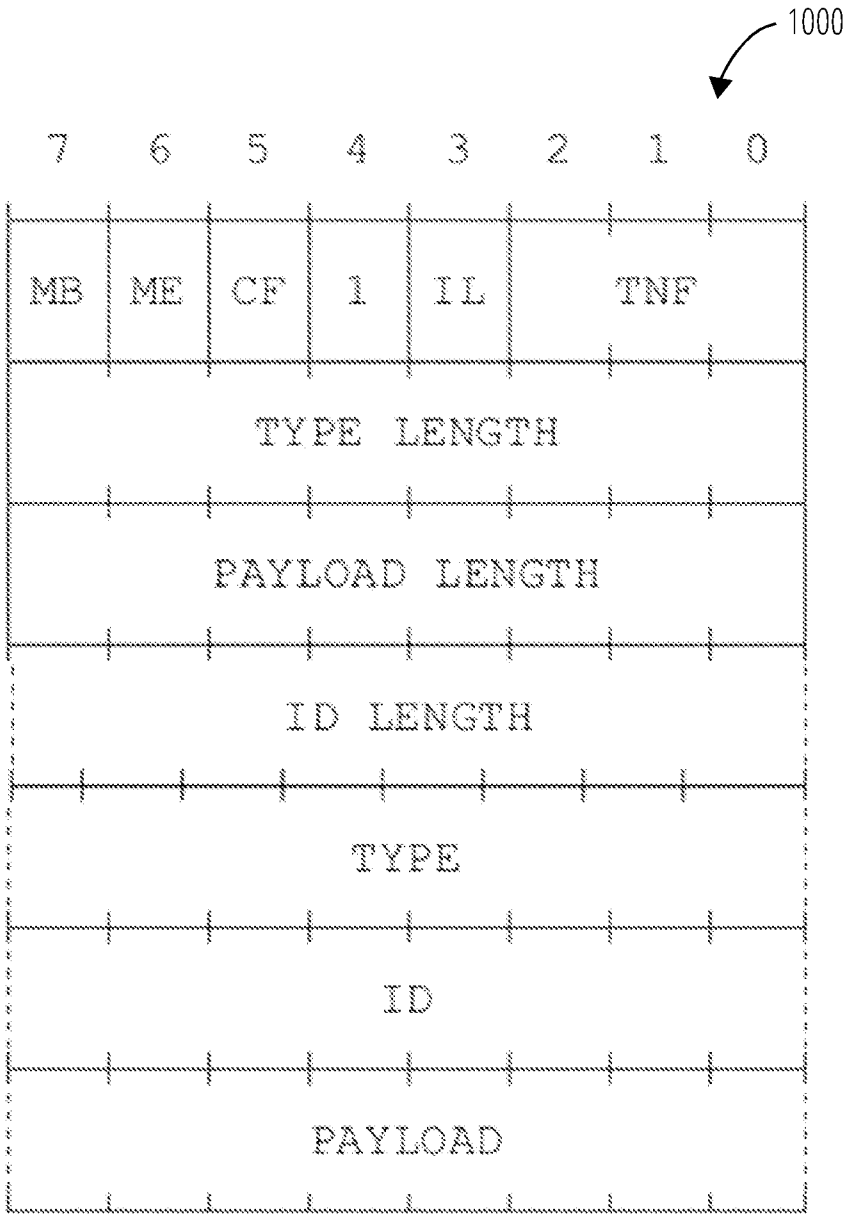
1000
7 6 5 4 3 2 1 0
MB
ME
CF
1
IL
TNF
TYPE LENGTH
PAYLOAD LENGTH
ID LENGTH
TYPE
ID
PAYLOAD

FIG. 10

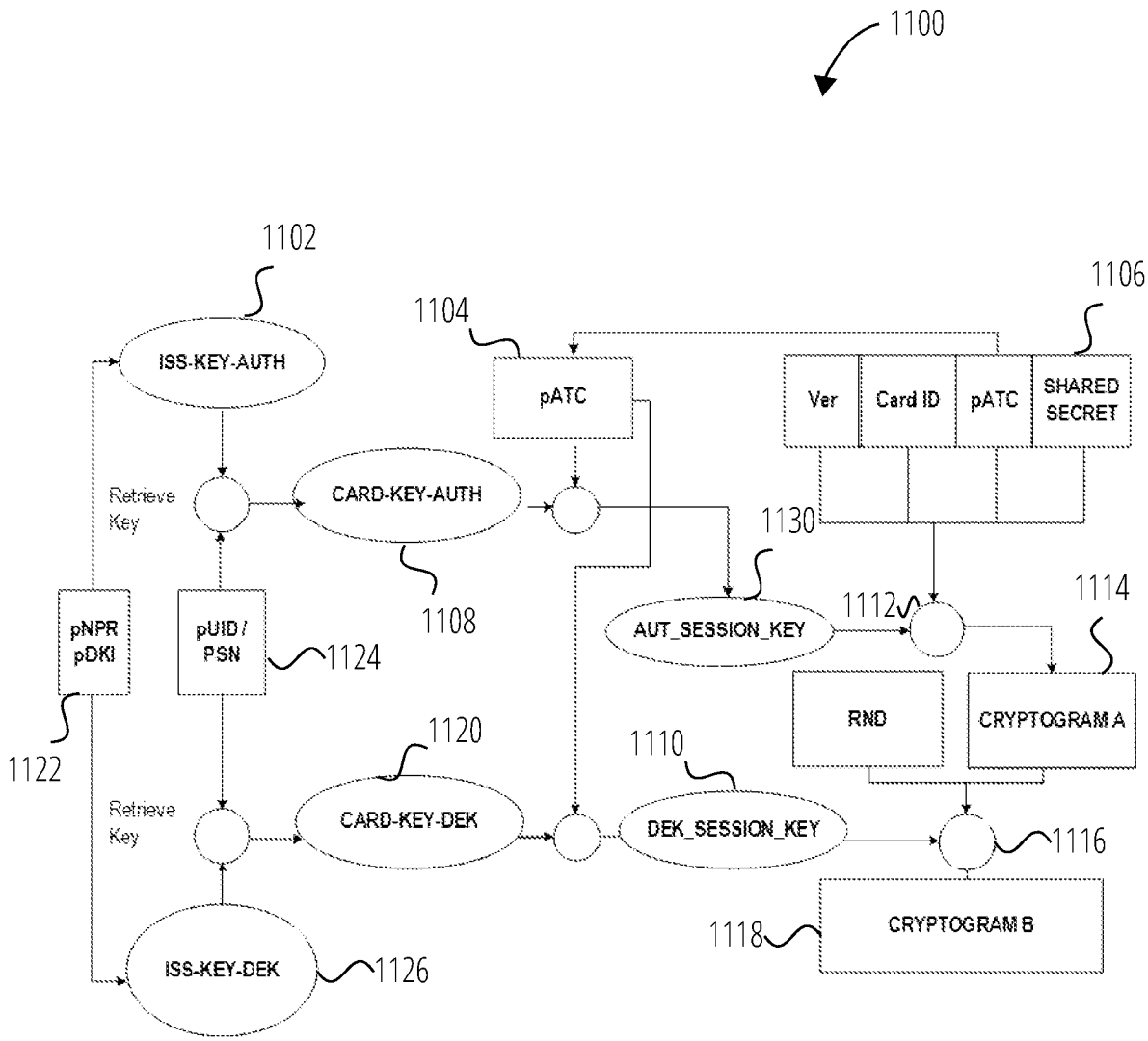
1100
1102
ISS-KEY-AUTH
1104
pATC
1106
Ver
Card ID
pATC
SHARED SECRET
Retrieve Key
CARD-KEY-AUTH
1108
1130
AUT_SESSION_KEY
1112
1114
pNPR pDKI
1122
pUID / PSN
1124
RND
CRYPTOGRAM A
1120
1110
Retrieve Key
CARD-KEY-DEK
DEK_SESSION_KEY
1116
CRYPTOGRAM B
1118
ISS-KEY-DEK
1126

FIG. 11

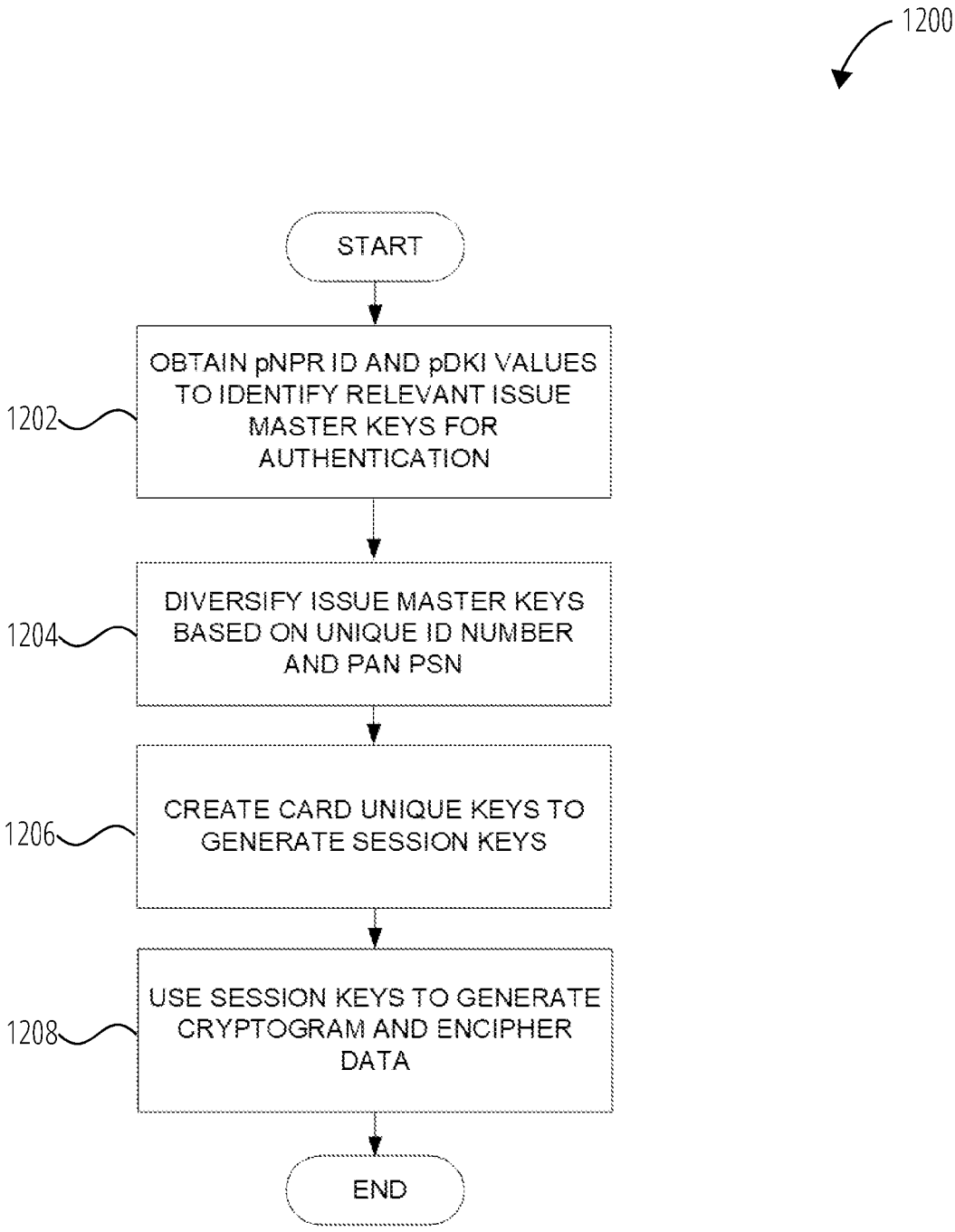
1200
START
OBTAIN pNPR ID AND pDKI VALUES TO IDENTIFY RELEVANT ISSUE MASTER KEYS FOR AUTHENTICATION
1202
DIVERSIFY ISSUE MASTER KEYS BASED ON UNIQUE ID NUMBER AND PAN PSN
1204
CREATE CARD UNIQUE KEYS TO GENERATE SESSION KEYS
1206
USE SESSION KEYS TO GENERATE CRYPTOGRAM AND ENCIPHER DATA
1208
END

FIG. 12

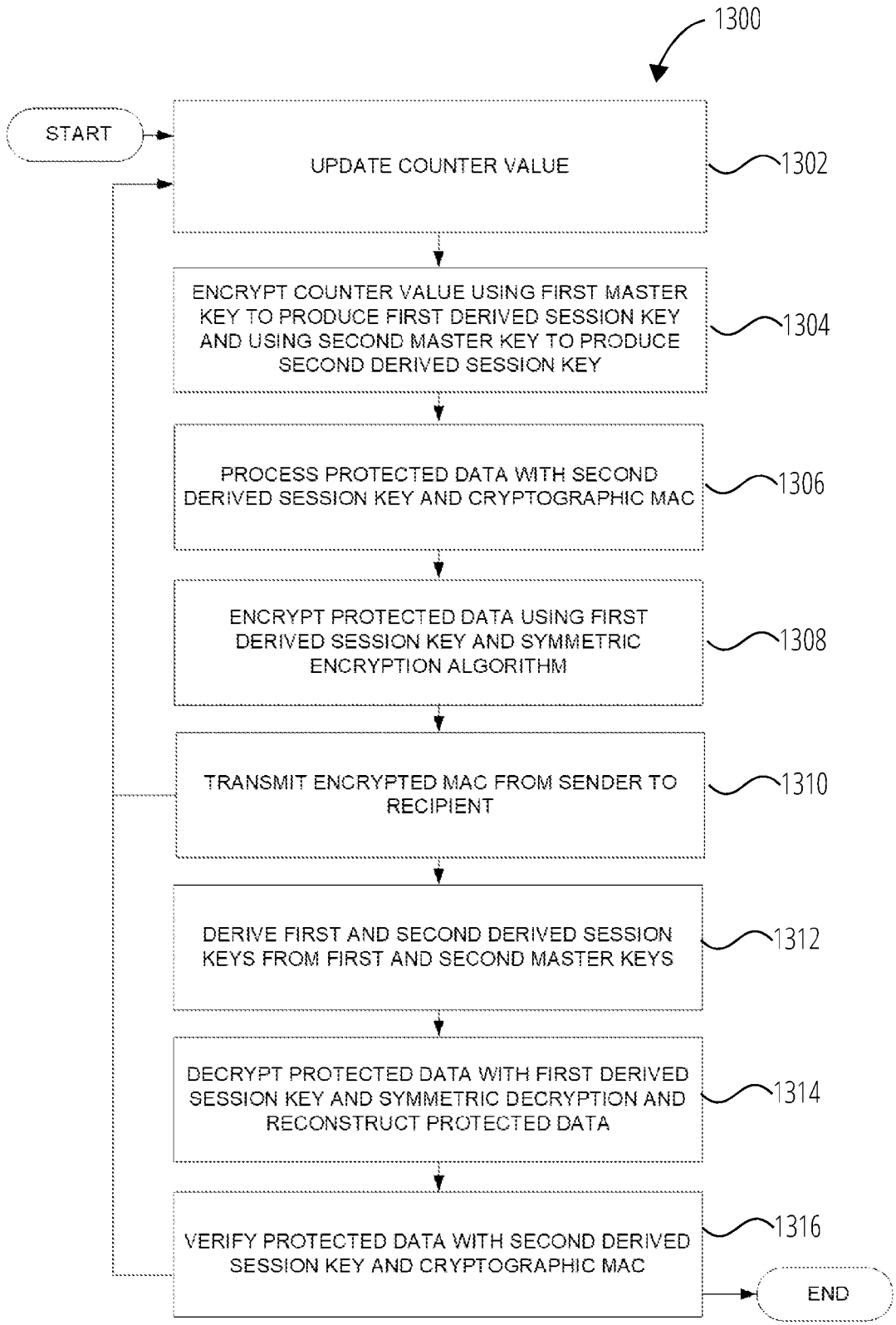
1300
START
UPDATE COUNTER VALUE
1302
ENCRYPT COUNTER VALUE USING FIRST MASTER KEY TO PRODUCE FIRST DERIVED SESSION KEY AND USING SECOND MASTER KEY TO PRODUCE SECOND DERIVED SESSION KEY
1304
PROCESS PROTECTED DATA WITH SECOND DERIVED SESSION KEY AND CRYPTOGRAPHIC MAC
1306
ENCRYPT PROTECTED DATA USING FIRST DERIVED SESSION KEY AND SYMMETRIC ENCRYPTION ALGORITHM
1308
TRANSMIT ENCRYPTED MAC FROM SENDER TO RECIPIENT
1310
DERIVE FIRST AND SECOND DERIVED SESSION KEYS FROM FIRST AND SECOND MASTER KEYS
1312
DECRYPT PROTECTED DATA WITH FIRST DERIVED SESSION KEY AND SYMMETRIC DECRYPTION AND RECONSTRUCT PROTECTED DATA
1314
VERIFY PROTECTED DATA WITH SECOND DERIVED SESSION KEY AND CRYPTOGRAPHIC MAC
1316
END

FIG. 13

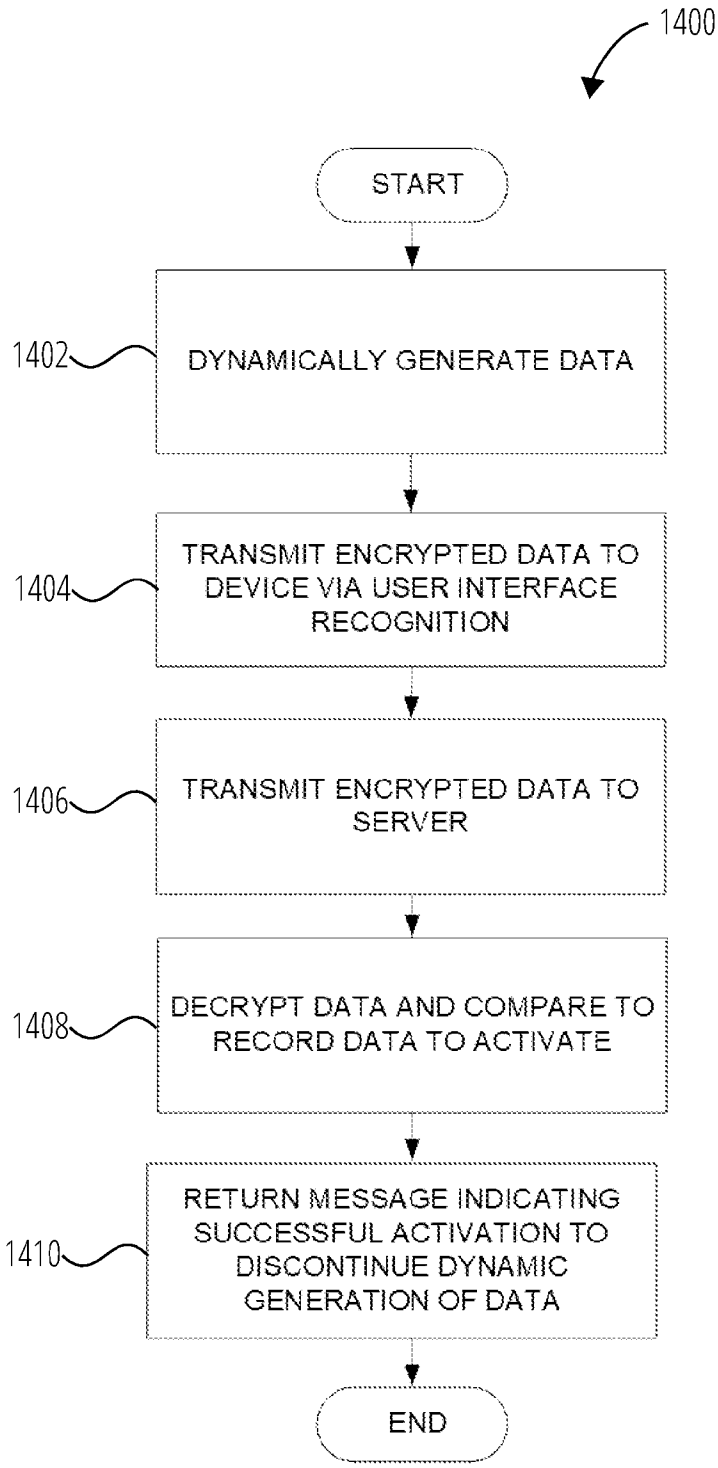
1400
START
1402
DYNAMICALLY GENERATE DATA
1404
TRANSMIT ENCRYPTED DATA TO DEVICE VIA USER INTERFACE RECOGNITION
1406
TRANSMIT ENCRYPTED DATA TO SERVER
1408
DECRYPT DATA AND COMPARE TO RECORD DATA TO ACTIVATE
1410
RETURN MESSAGE INDICATING SUCCESSFUL ACTIVATION TO DISCONTINUE DYNAMIC GENERATION OF DATA
END

FIG. 14