US 20200167775A1

(19) **United States**

(12) **Patent Application Publication** (10) **Pub. No.: US 2020/0167775 A1**

REESE et al. (43) **Pub. Date: May 28, 2020**

(54) **VIRTUAL POS TERMINAL METHOD AND APPARATUS**

(71) Applicant: **Intel Corporation**, Santa Clara, CA (US)

(72) Inventors: **KENNETH W. REESE**, Portland, OR (US); **MARK H. PRICE**, Placitas, NM (US)

(21) Appl. No.: **16/559,413**

(22) Filed: **Sep. 3, 2019**

Related U.S. Application Data

(63) Continuation of application No. 14/739,911, filed on Jun. 15, 2015, now Pat. No. 10,410,211.

Publication Classification

(51) **Int. Cl.**

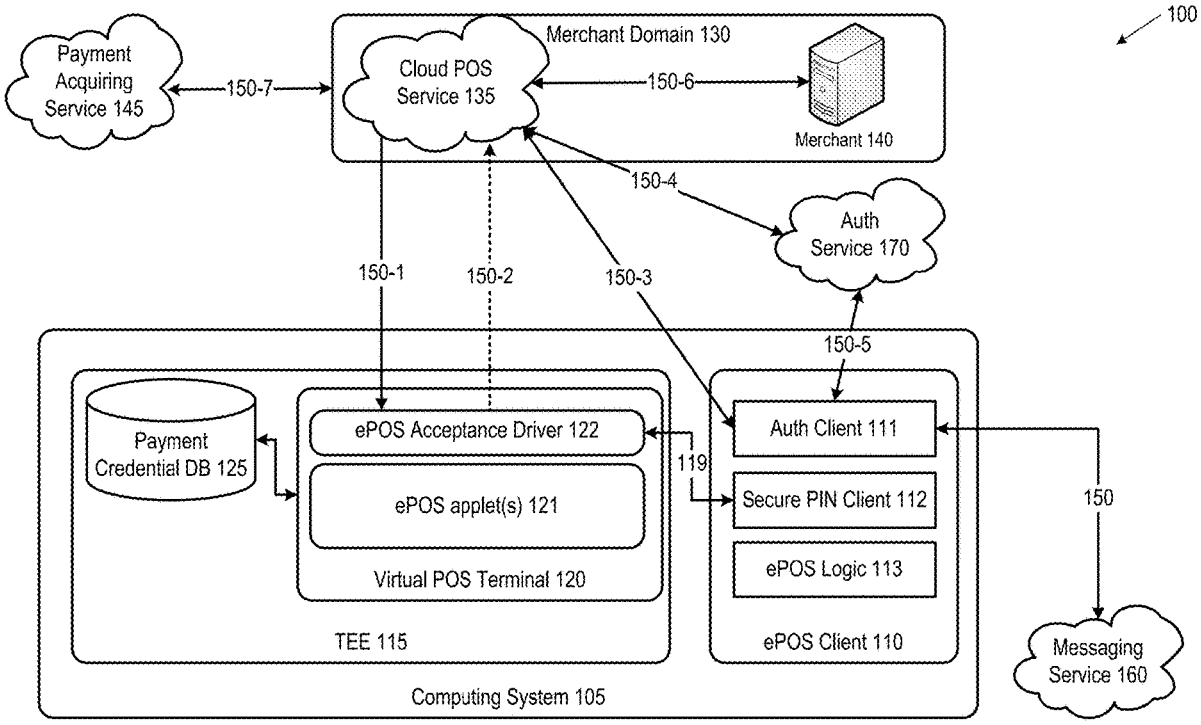
G06Q 20/38	(2006.01)
G06Q 20/20	(2006.01)
G06Q 20/40	(2006.01)

(52) **U.S. Cl.**

CPC ... ***G06Q 20/38215*** (2013.01); ***G06Q 20/4012*** (2013.01); ***G06Q 20/206*** (2013.01)

(57) **ABSTRACT**

Methods, systems, and storage media are described for processing point of sale (POS) transactions. In embodiments, a computing device may receive a transaction initiation, and provide a selection of a payment credential to be used to process a POS transaction. The computing device includes a trusted execution environment to process the POS transaction in response to the selection of the payment credential. The trusted execution environment may comprise a payment credential storage unit to store payment credentials and a virtual POS terminal that may validate a merchant terminal associated with the transaction initiation, process the POS transaction using the selected payment credential to generate payment data, and encrypt the payment data. The computing device may communicate the encrypted payment data to a cloud POS service for further processing. Other embodiments may be described and/or claimed.



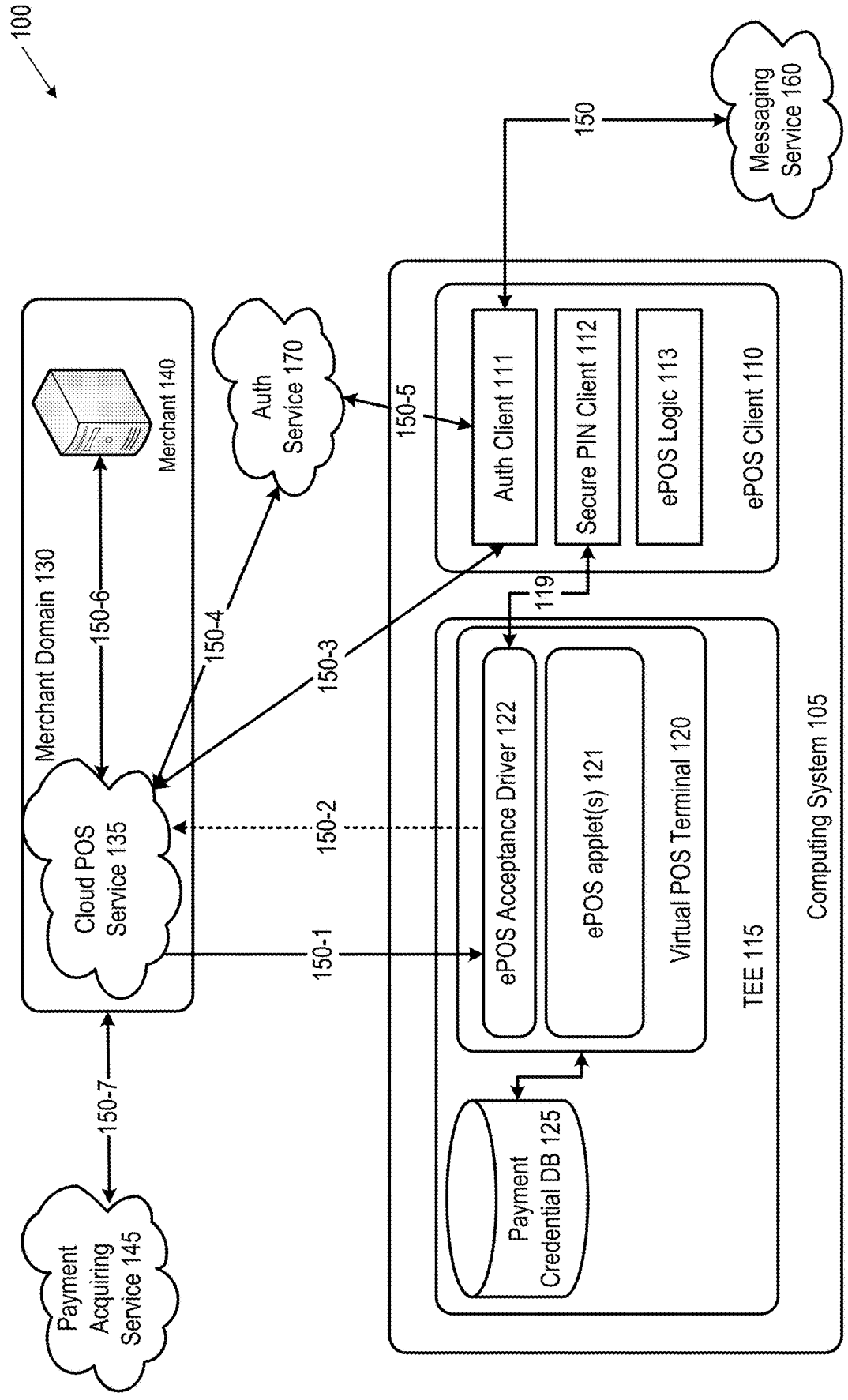

100
Messaging Service 160
150
Auth Service 170
150-5
Merchant 140
Merchant Domain 130
Auth Client 111
Secure PIN Client 112
ePOS Logic 113
ePOS Client 110
150-6
150-4
150-3
119
Cloud POS Service 135
150-2
ePOS Acceptance Driver 122
ePOS applet(s) 121
Virtual POS Terminal 120
TEE 115
Computing System 105
150-1
Payment Credential DB 125
150-7
Payment Acquiring Service 145

Figure 1

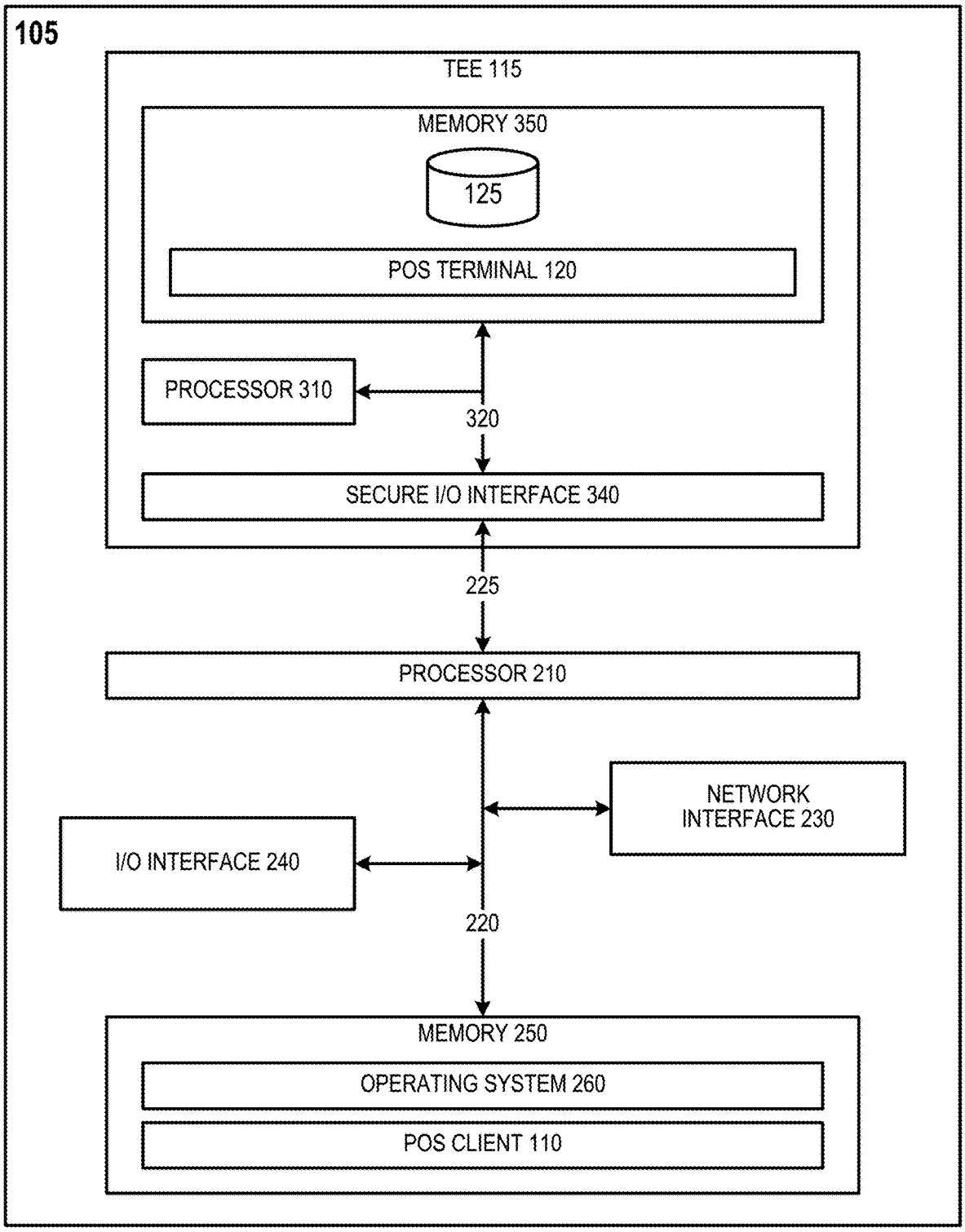

105
TEE 115
MEMORY 350
125
POS TERMINAL 120
PROCESSOR 310
320
SECURE I/O INTERFACE 340
225
PROCESSOR 210
NETWORK INTERFACE 230
I/O INTERFACE 240
220
MEMORY 250
OPERATING SYSTEM 260
POS CLIENT 110

Figure 2

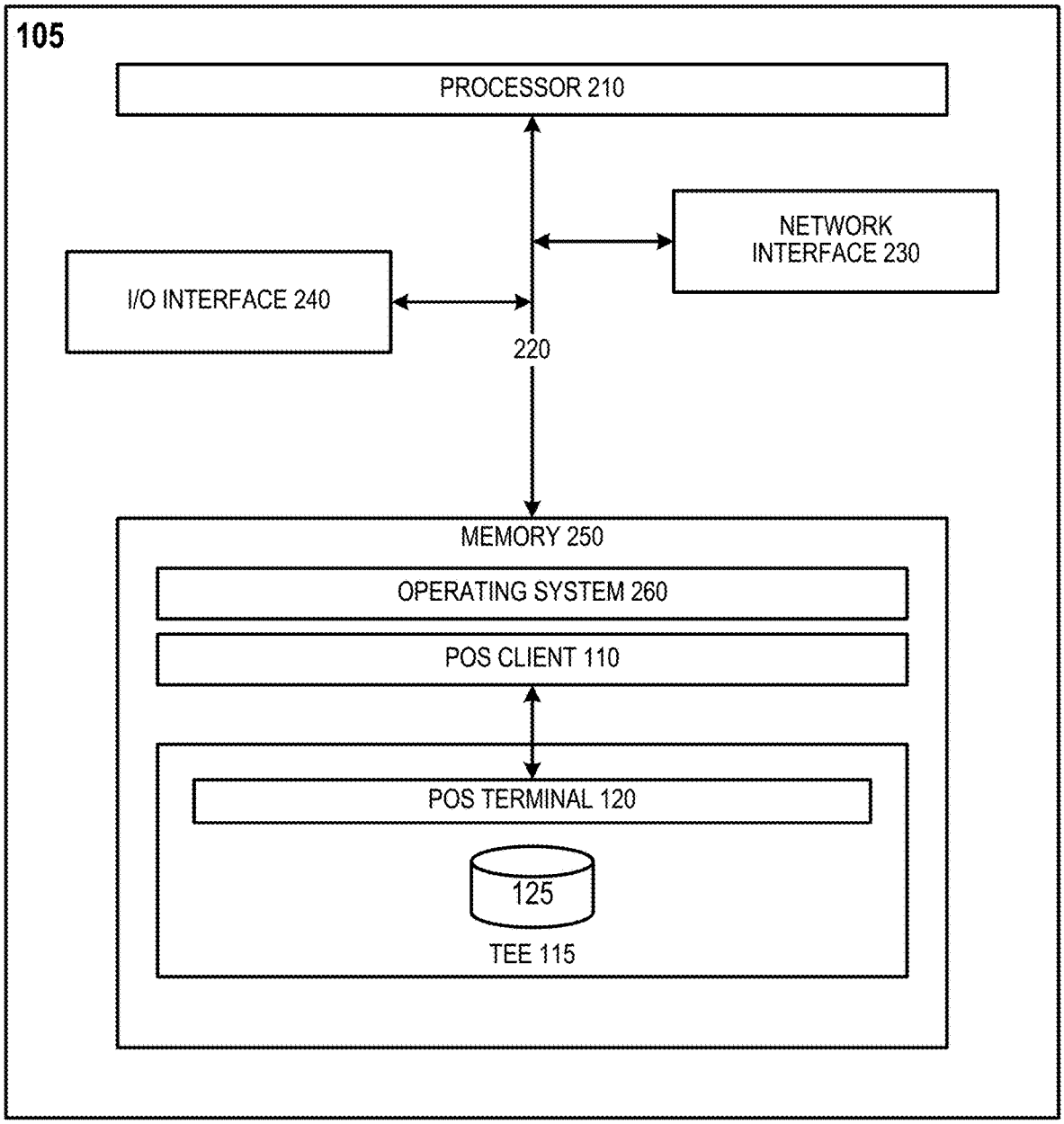

105
PROCESSOR 210
NETWORK INTERFACE 230
I/O INTERFACE 240
220
MEMORY 250
OPERATING SYSTEM 260
POS CLIENT 110
POS TERMINAL 120
125
TEE 115

Figure 3

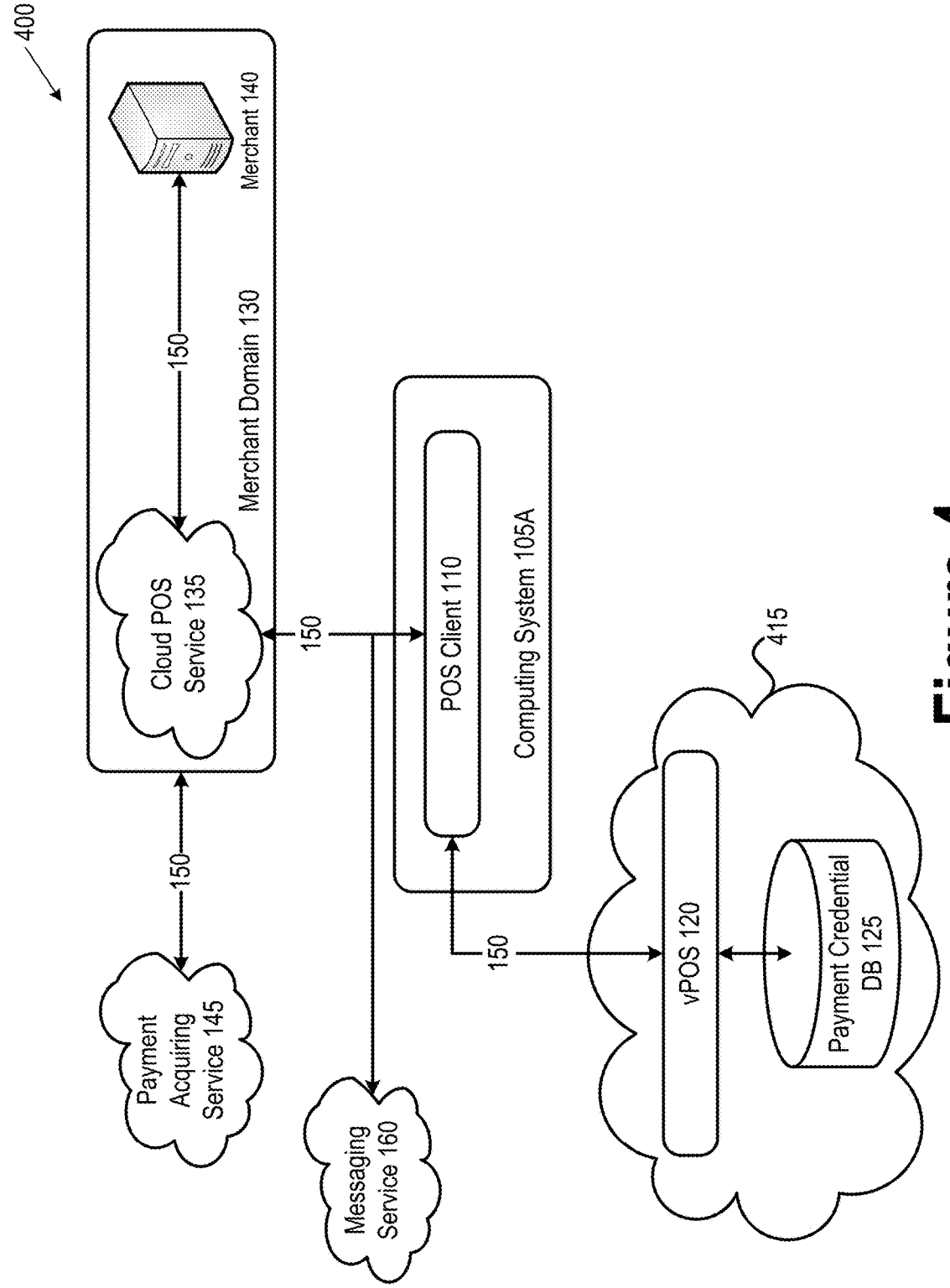

400
Merchant 140
Merchant Domain 130
150
Cloud POS Service 135
150
POS Client 110
Computing System 105A
150
Payment Acquiring Service 145
Messaging Service 160
415
vPOS 120
Payment Credential DB 125

Figure 4

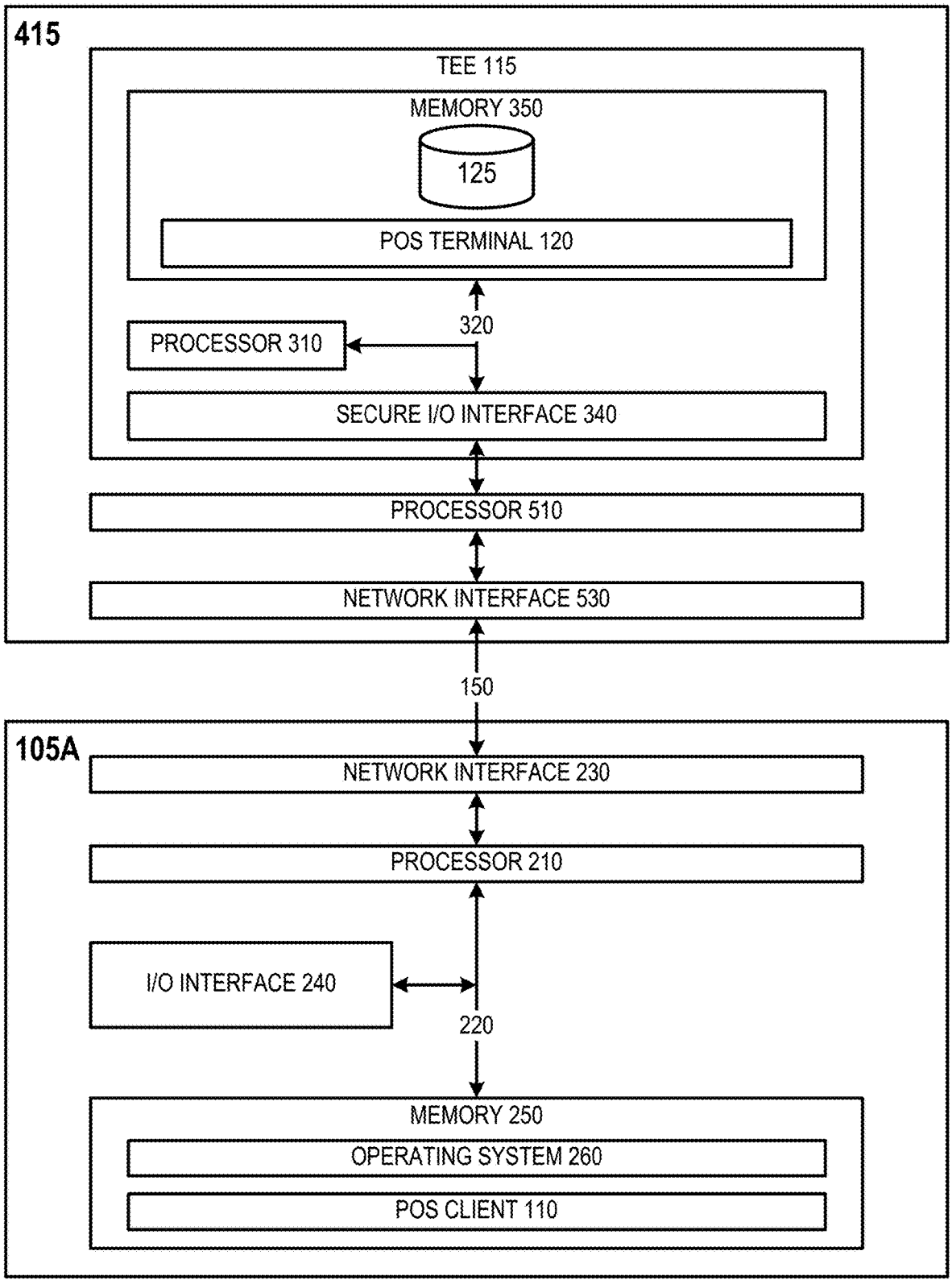

415
TEE 115
MEMORY 350
125
POS TERMINAL 120
320
PROCESSOR 310
SECURE I/O INTERFACE 340
PROCESSOR 510
NETWORK INTERFACE 530
150
105A
NETWORK INTERFACE 230
PROCESSOR 210
I/O INTERFACE 240
220
MEMORY 250
OPERATING SYSTEM 260
POS CLIENT 110

Figure 5

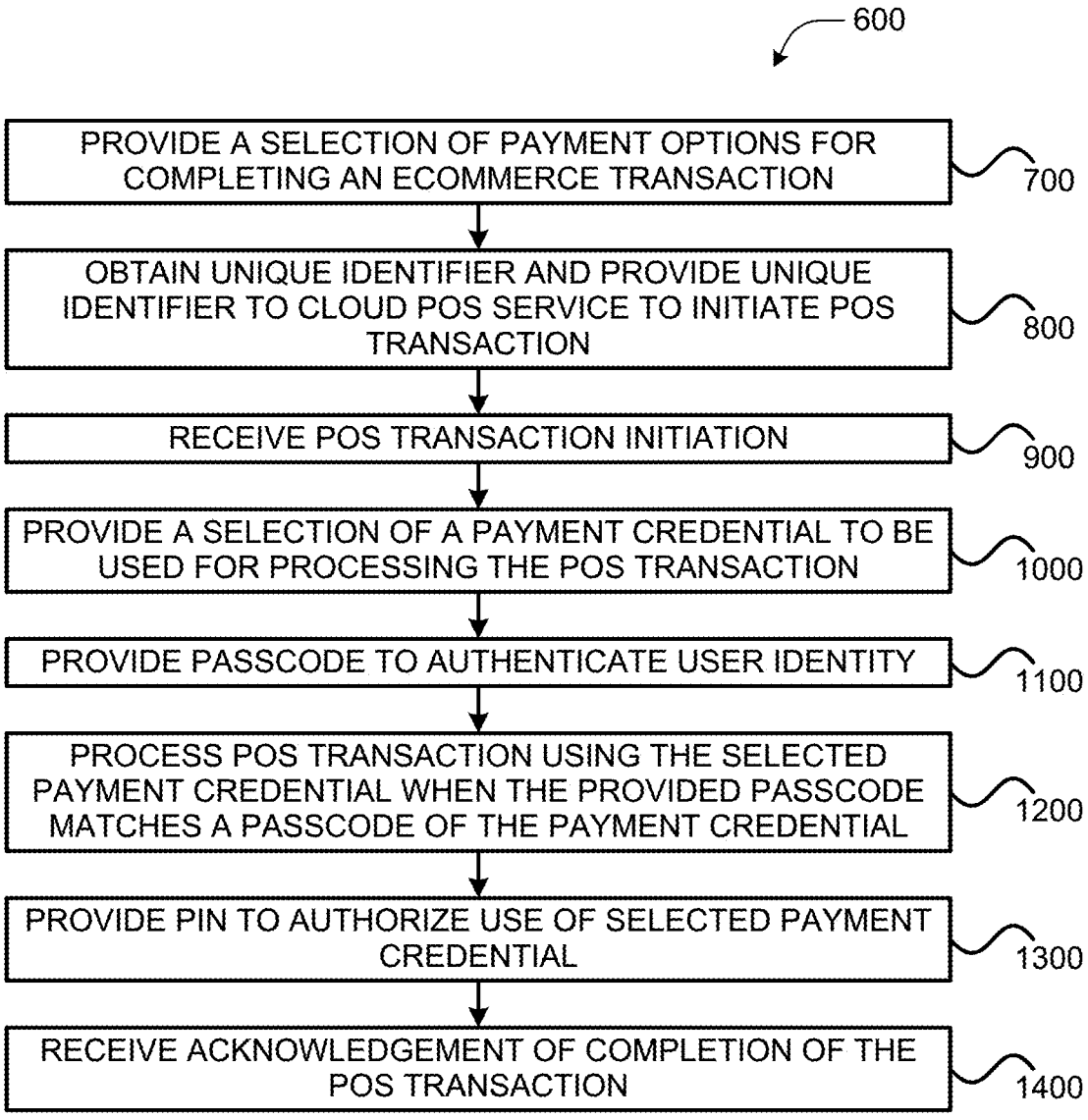

600
PROVIDE A SELECTION OF PAYMENT OPTIONS FOR COMPLETING AN ECOMMERCE TRANSACTION
700
OBTAIN UNIQUE IDENTIFIER AND PROVIDE UNIQUE IDENTIFIER TO CLOUD POS SERVICE TO INITIATE POS TRANSACTION
800
RECEIVE POS TRANSACTION INITIATION
900
PROVIDE A SELECTION OF A PAYMENT CREDENTIAL TO BE USED FOR PROCESSING THE POS TRANSACTION
1000
PROVIDE PASSCODE TO AUTHENTICATE USER IDENTITY
1100
PROCESS POS TRANSACTION USING THE SELECTED PAYMENT CREDENTIAL WHEN THE PROVIDED PASSCODE MATCHES A PASSCODE OF THE PAYMENT CREDENTIAL
1200
PROVIDE PIN TO AUTHORIZE USE OF SELECTED PAYMENT CREDENTIAL
1300
RECEIVE ACKNOWLEDGEMENT OF COMPLETION OF THE POS TRANSACTION
1400

Figure 6

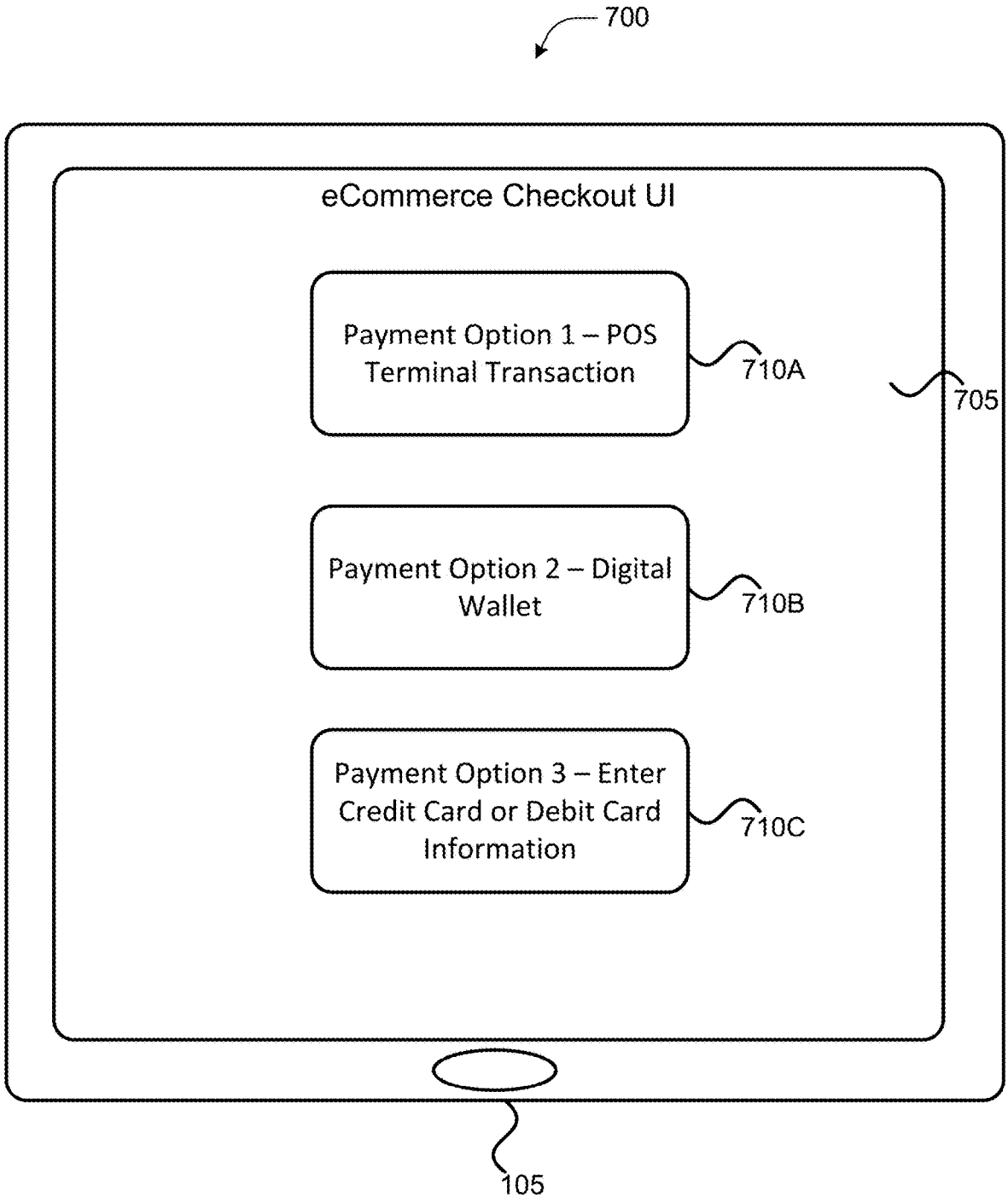

700
eCommerce Checkout UI
Payment Option 1 – POS Terminal Transaction
710A
705
Payment Option 2 – Digital Wallet
710B
Payment Option 3 – Enter Credit Card or Debit Card Information
710C
105

Figure 7

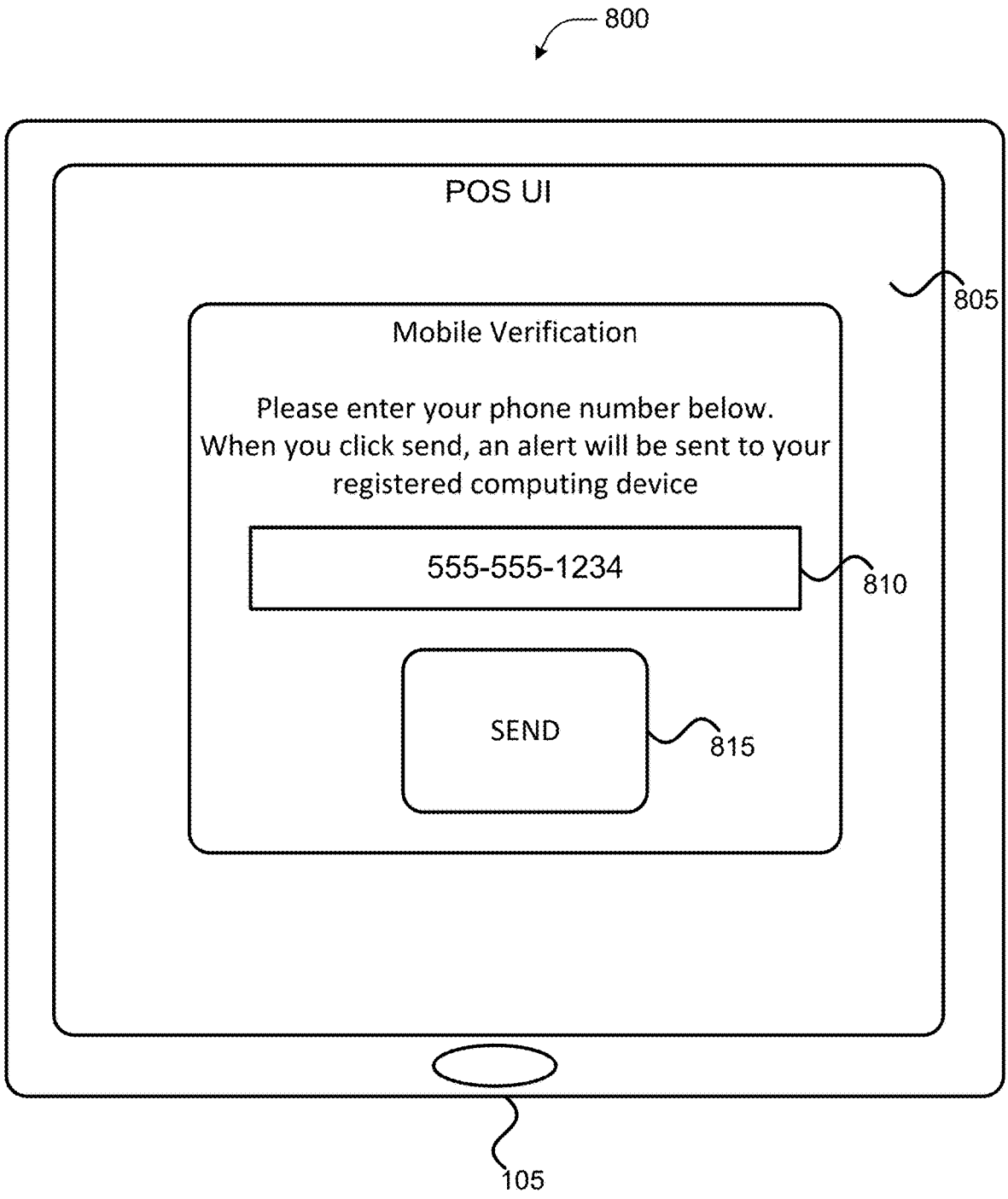

800
POS UI
805
Mobile Verification
Please enter your phone number below.
When you click send, an alert will be sent to your registered computing device
555-555-1234
810
SEND
815
105

Figure 8

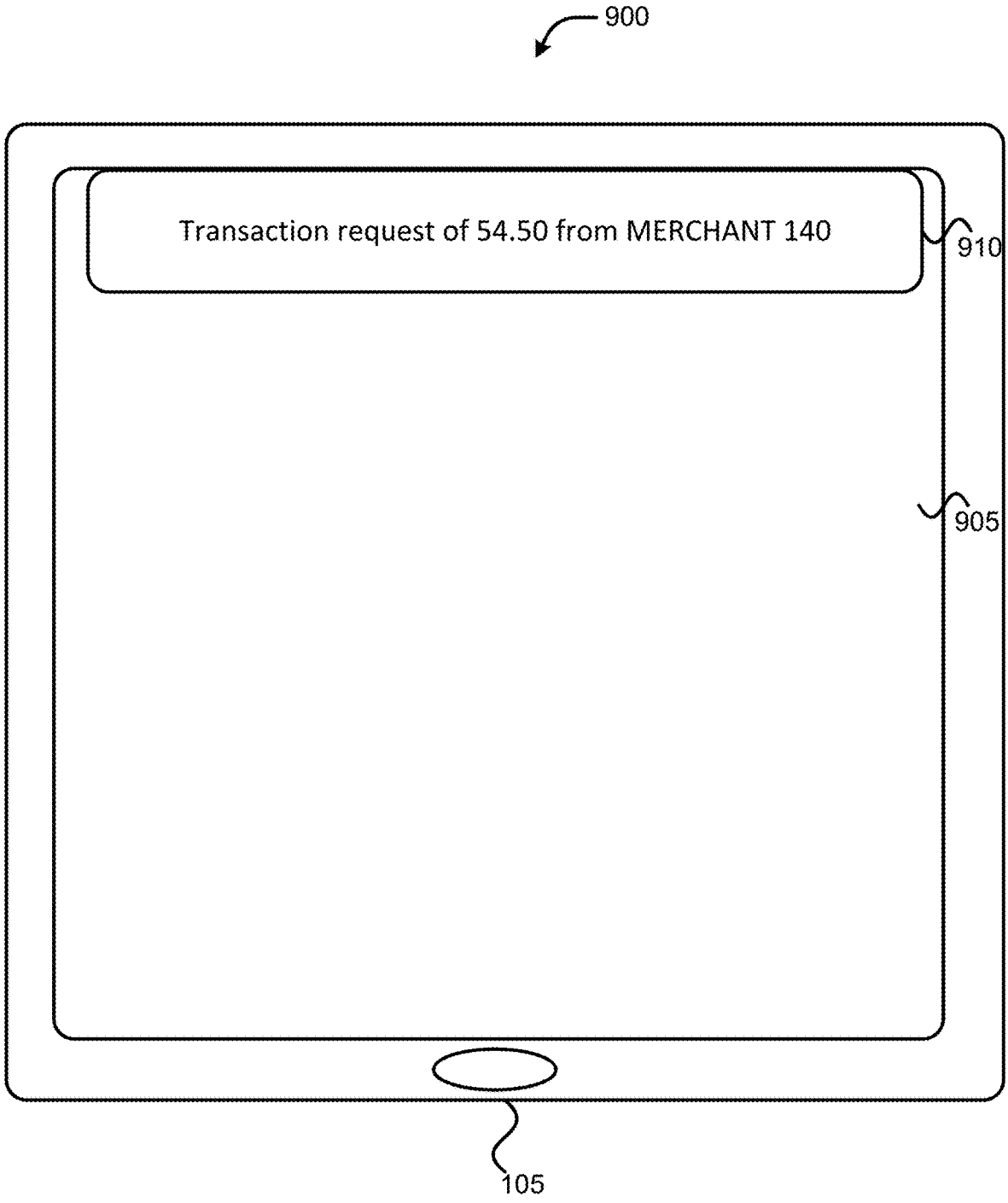

900
Transaction request of 54.50 from MERCHANT 140
910
905
105

Figure 9

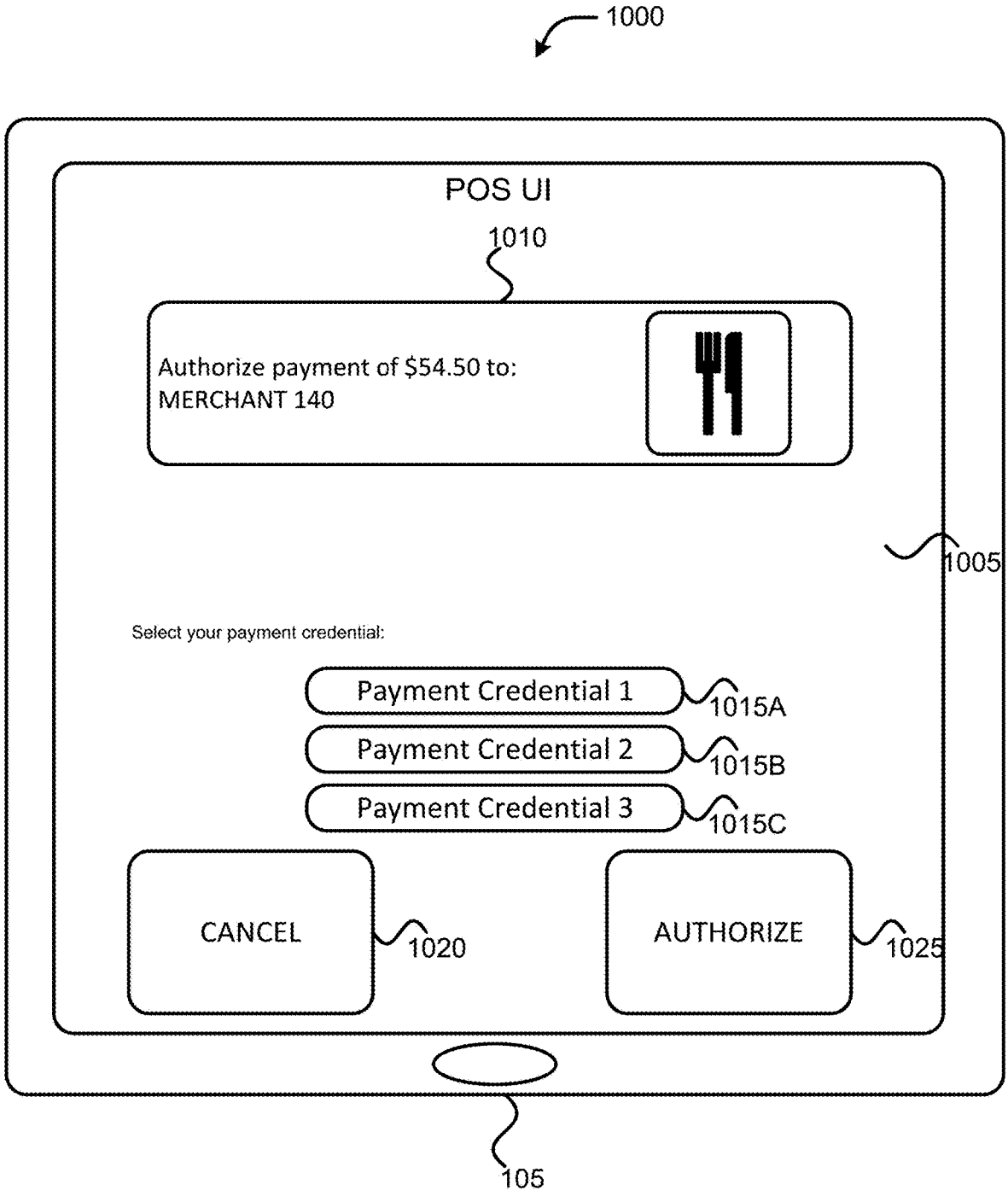

1000
POS UI
1010
Authorize payment of $54.50 to:
MERCHANT 140
1005
Select your payment credential:
Payment Credential 1
1015A
Payment Credential 2
1015B
Payment Credential 3
1015C
CANCEL
1020
AUTHORIZE
1025
105

Figure 10

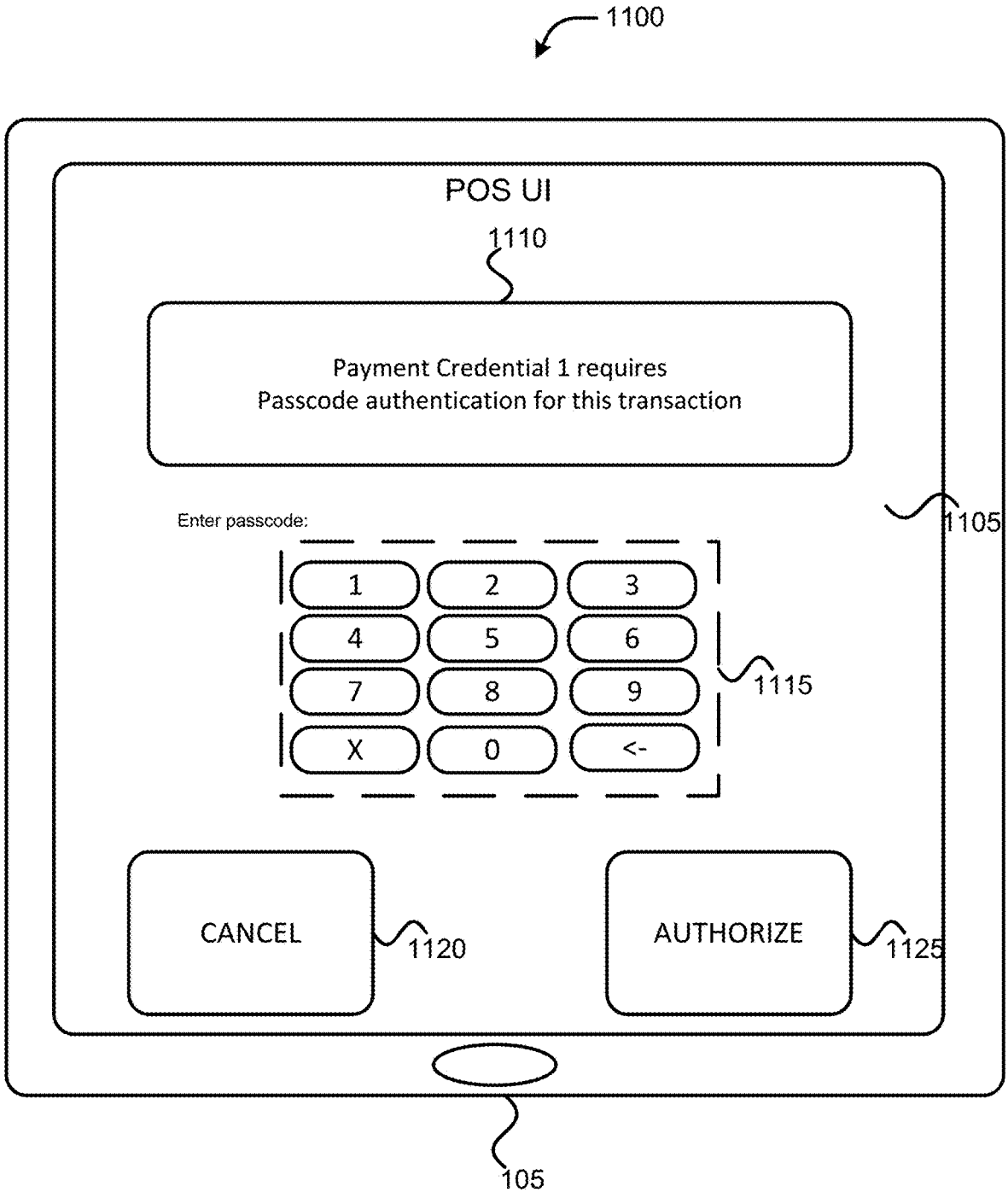

1100
POS UI
1110
Payment Credential 1 requires
Passcode authentication for this transaction
Enter passcode:
1105
1
2
3
4
5
6
7
8
9
X
0
<-
1115
CANCEL
1120
AUTHORIZE
1125
105

Figure 11

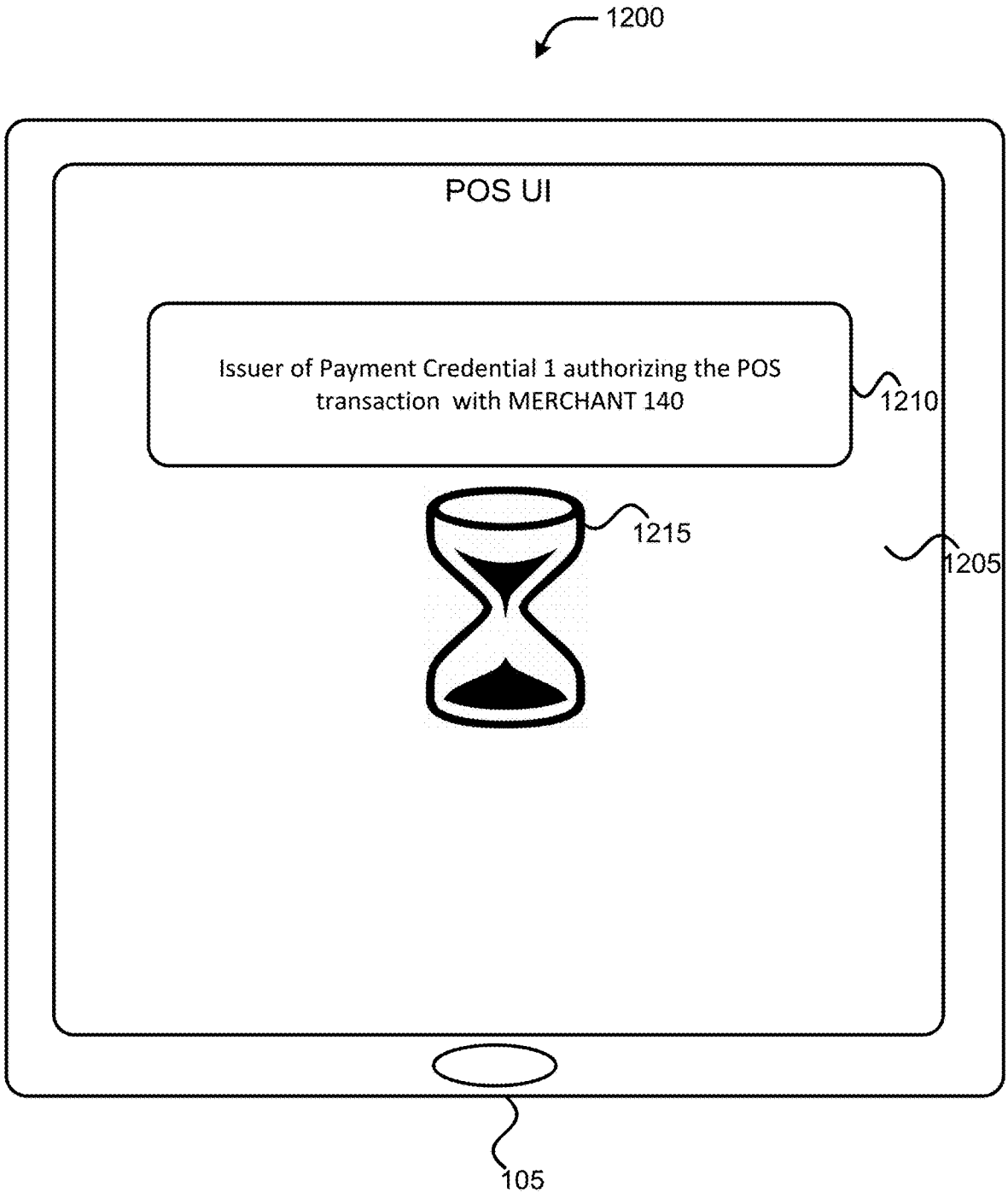

1200
POS UI
Issuer of Payment Credential 1 authorizing the POS transaction with MERCHANT 140
1210
1215
1205
105

Figure 12

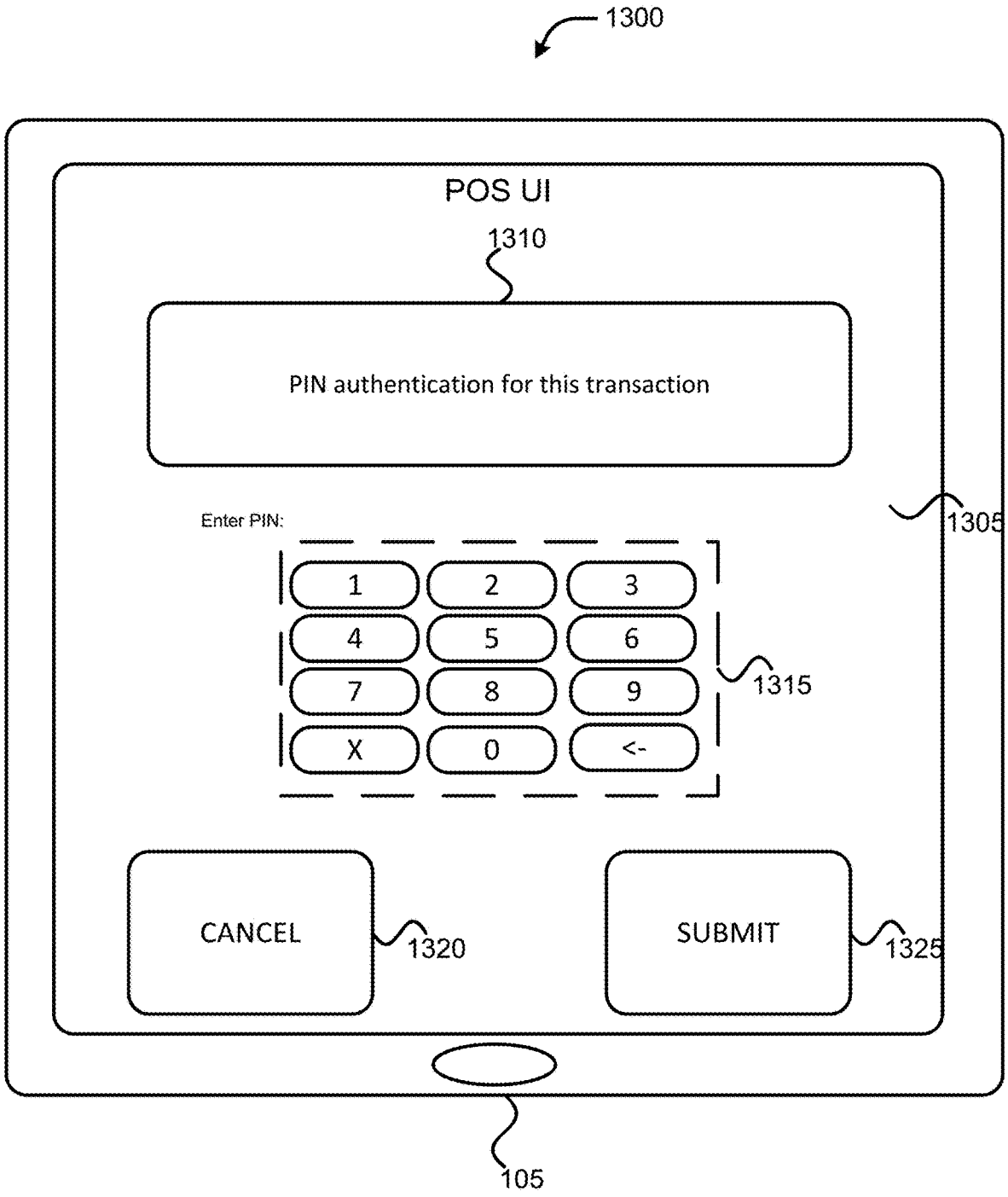

1300
POS UI
1310
PIN authentication for this transaction
Enter PIN:
1305
1
2
3
4
5
6
7
8
9
X
0
<-
1315
CANCEL
1320
SUBMIT
1325
105

Figure 13

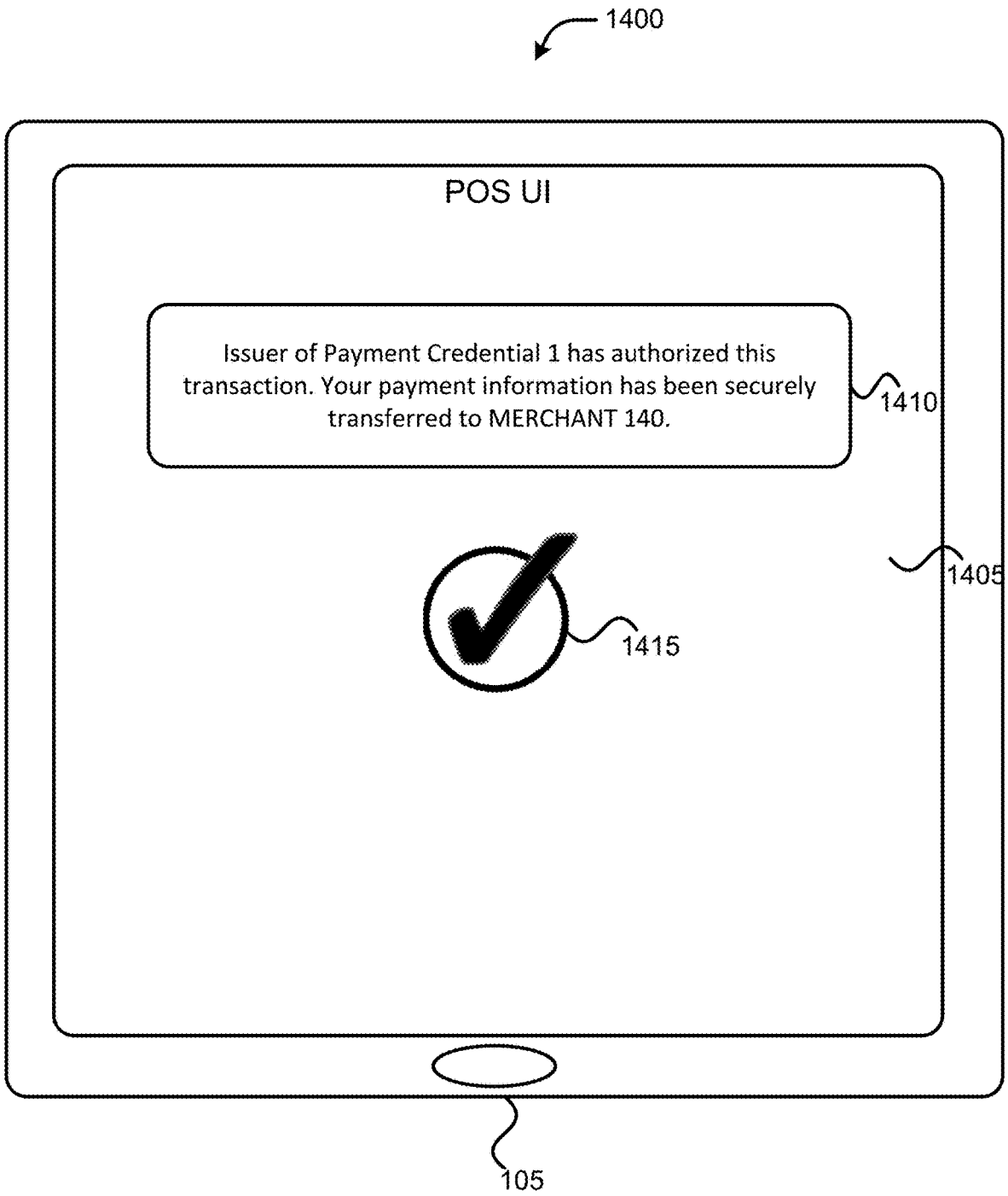

1400
POS UI
Issuer of Payment Credential 1 has authorized this transaction. Your payment information has been securely transferred to MERCHANT 140.
1410
1405
1415
105

Figure 14

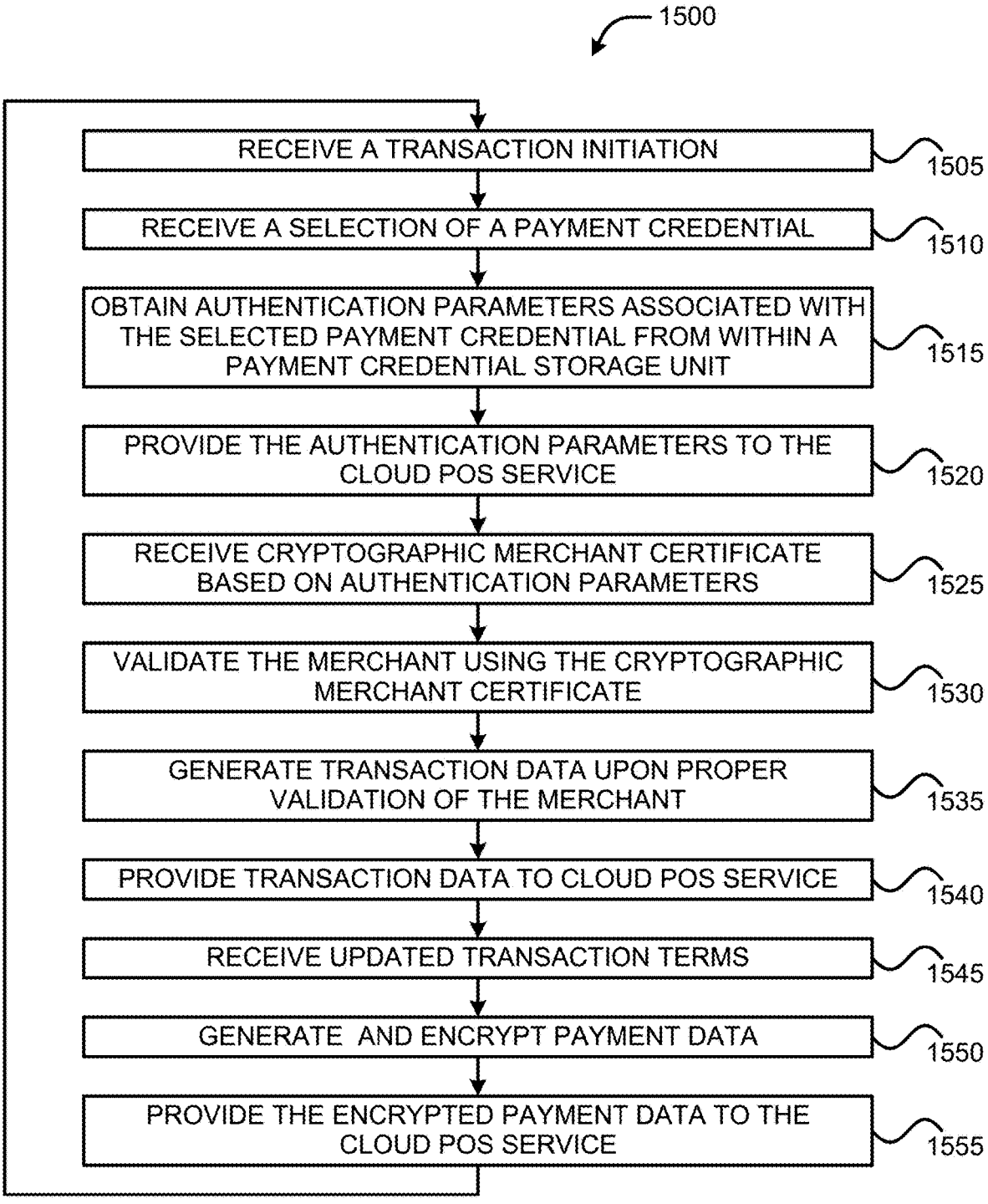

1500
RECEIVE A TRANSACTION INITIATION
1505
RECEIVE A SELECTION OF A PAYMENT CREDENTIAL
1510
OBTAIN AUTHENTICATION PARAMETERS ASSOCIATED WITH THE SELECTED PAYMENT CREDENTIAL FROM WITHIN A PAYMENT CREDENTIAL STORAGE UNIT
1515
PROVIDE THE AUTHENTICATION PARAMETERS TO THE CLOUD POS SERVICE
1520
RECEIVE CRYPTOGRAPHIC MERCHANT CERTIFICATE BASED ON AUTHENTICATION PARAMETERS
1525
VALIDATE THE MERCHANT USING THE CRYPTOGRAPHIC MERCHANT CERTIFICATE
1530
GENERATE TRANSACTION DATA UPON PROPER VALIDATION OF THE MERCHANT
1535
PROVIDE TRANSACTION DATA TO CLOUD POS SERVICE
1540
RECEIVE UPDATED TRANSACTION TERMS
1545
GENERATE AND ENCRYPT PAYMENT DATA
1550
PROVIDE THE ENCRYPTED PAYMENT DATA TO THE CLOUD POS SERVICE
1555

Figure 15

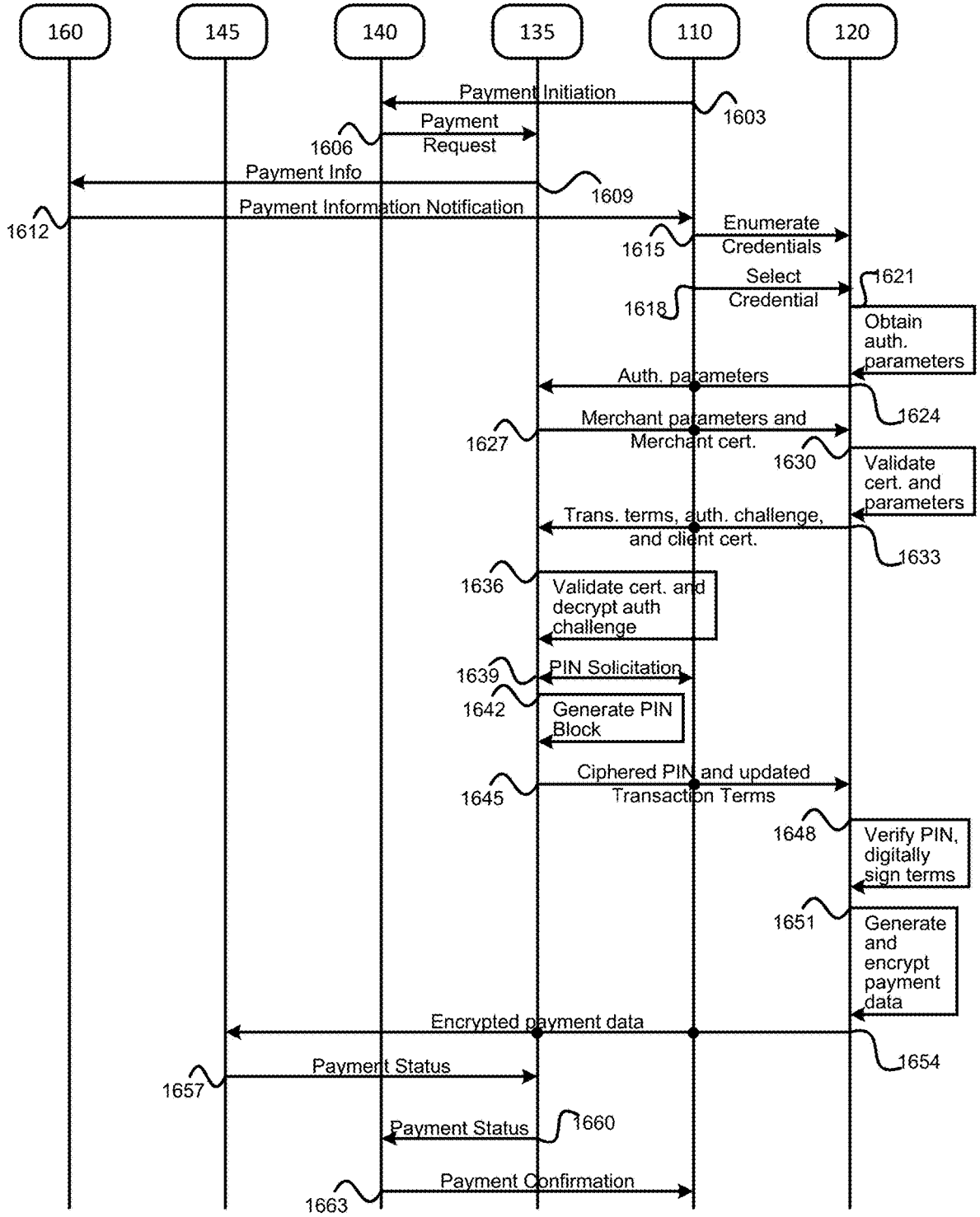

160
145
140
135
110
120
Payment Initiation
1603
Payment Request
1606
Payment Info
1609
Payment Information Notification
1612
Enumerate Credentials
1615
Select Credential
1618
1621
Obtain auth. parameters
Auth. parameters
1624
Merchant parameters and Merchant cert.
1627
1630
Validate cert. and parameters
Trans. terms, auth. challenge, and client cert.
1633
1636
Validate cert. and decrypt auth challenge
PIN Solicitation
1639
1642
Generate PIN Block
Ciphered PIN and updated Transaction Terms
1645
1648
Verify PIN, digitally sign terms
1651
Generate and encrypt payment data
Encrypted payment data
1654
Payment Status
1657
Payment Status
1660
Payment Confirmation
1663

Figure 16

VIRTUAL POS TERMINAL METHOD AND APPARATUS

CROSS REFERENCE TO RELATED APPLICATIONS

[0001] The present application is a Continuation of U.S. application Ser. No. 14/739,911, filed on Jun. 15, 2015 entitled "VIRTUAL POS TERMINAL METHOD AND APPARATUS", the contents of which is hereby incorporated by reference in its entirety.

FIELD

[0002] The present disclosure relates to the technical field of electronic transactions, and in particular, to apparatuses, methods and storage media for providing a virtual point of sale (POS) terminal.

BACKGROUND

[0003] In today's world of commerce there are two dominant methods of making purchases. One is categorized as "card-present", where the card holder and payment card are physically present at the time of transaction, and the other is "card-not-present", where the card holder is not physically present during the transaction. The former transactions typically occur at a physical establishment (e.g., store, restaurant, gas station, etc.), while the latter typically occur during making purchases over the Internet (also referred to as "ecommerce" purchases) or making purchases via phone or mail-order. Card-present transactions typically enjoy far less fraud than card-not-present transactions. This is often attributed to the ability to prove credential authenticity (i.e., authenticity of the payment card being used) and card holder legitimacy (i.e., the individual making the transaction is the card holder or an authorized agent of the card holder). The enforcement of these two attributes is typically the responsibility of a physical point of sale (POS) terminal and/or other like computing device that can verify correctness of the information contained on the card and identity authentication, which is generally a personal identification number (PIN), a card holder signature, and the like.

[0004] Most ecommerce transactions require a user to input payment card information and cardholder identification information into text fields of a web-based user interface. Additionally, most phone-order purchase require customers to recite payment card information and cardholder identification information over the phone. This is because in most card-not-present transactions, the terminal-based authentication methods are usually not available. Furthermore, most current fraud prevention measures for card-not-present transactions, such as typing/reciting the account number contained on the front of a payment card plus other identifying information like a 3-digit cardholder verification value (CVV) on the back of most payment cards, have been demonstrated to not reduce fraud to the level experienced by card-present transactions. Moreover, the attempts to strengthen cardholder presence assurances for card-not-present transactions has resulted in the introduction of online authentication protocols, such as the XML-based 3D Secure (which is also known by various trade names including Visa® Verified by Visa, MasterCard® SecureCode, American Express® SafeKey, and JCP International® J/Secure). These protocols introduce an additional authentication step in the payment checkout process that verifies cardholder identity through the card-issuing bank. However, the additional verification steps introduced by the aforementioned protocols tend to result in customer frustration in the form of online shopping cart abandonment, which may lead to a loss of revenue for ecommerce merchants.

[0005] Accordingly, it may be desirable to bring the user experience and transaction security aspects of a physical POS terminal (i.e., card-present transactions) to online, phone-order, and/or mail-order transactions (i.e., card-not-present transactions). Furthermore, it may be desirable to provide transaction security for card-not-present transactions while reducing a number of steps for providing payment card verification and cardholder authorization.

BRIEF DESCRIPTION OF THE DRAWINGS

[0006] Embodiments will be readily understood by the following detailed description in conjunction with the accompanying drawings. To facilitate this description, like reference numerals designate like structural elements. Embodiments are illustrated by way of example, and not by way of limitation, in the figures of the accompanying drawings.

[0007] FIG. **1** illustrates an arrangement for conducting electronic transaction with virtual POS terminal of the present disclosure, in accordance with various example embodiments;

[0008] FIG. **2** illustrates the components of an example computing device having the virtual POS terminal, in accordance with various example embodiments;

[0009] FIG. **3** illustrates the components of another example computing device with the virtual POS terminal, in accordance with various other example embodiments;

[0010] FIG. **4** illustrates an arrangement for conducting electronic transaction with virtual POS terminal of the present disclosure, in accordance with various other example embodiments;

[0011] FIG. **5** illustrates the components of another example computing device and a cloud trusted execution environment with the virtual POS terminal, in accordance with various other example embodiments;

[0012] FIG. **6** illustrates a method of processing a POS transaction using the virtual POS terminal, in accordance with various example embodiments;

[0013] FIGS. **7-14** illustrate stages of performing the method of FIG. **4**;

[0014] FIG. **15** illustrates an example process for processing a POS transaction using the virtual POS terminal, in accordance with various embodiments; and

[0015] FIG. **16** illustrates a flow diagram illustrating a POS transaction using the virtual POS terminal, in accordance with various embodiments.

DETAILED DESCRIPTION

[0016] In the following detailed description, reference is made to the accompanying drawings which form a part hereof wherein like numerals designate like parts throughout, and in which is shown by way of illustrated embodiments that may be practiced. It is to be understood that other embodiments may be utilized and structural and/or logical changes may be made without departing from the scope of the present disclosure. Therefore, the following detailed

description is not to be taken in a limiting sense, and the scope of embodiments is defined by the appended claims and their equivalents.

[0017] Various operations may be described as multiple discrete actions and/or operations in turn, in a manner that is most helpful in understanding the claimed subject matter. However, the order of description should not be construed to imply that the various operations are necessarily order dependent. In particular, these operations may not be performed in the order of presentation. Operations described may be performed in a different order than the described embodiments. Various additional operations may be performed and/or described operations may be omitted in additional embodiments.

[0018] For the purposes of the present disclosure, the phrase "A and/or B" means (A), (B), or (A and B). For the purposes of the present disclosure, the phrase "A, B, and/or C" means (A), (B), (C), (A and B), (A and C), (B and C), or (A, B and C). For the purposes of the present disclosure, the phrase "at least one of A and B" means (A), (B), or (A and B).

[0019] The description may use the phrases "in an embodiment", or "in embodiments", which may each refer to one or more of the same or different embodiments. Furthermore, the terms "comprising," "including," "having," and the like, as used with respect to embodiments of the present disclosure, are synonymous. The terms "coupled," "communicatively coupled," along with derivatives thereof are used herein. The term "coupled" may mean two or more elements are in direct physical or electrical contact with one another, may mean that two or more elements indirectly contact each other but still cooperate or interact with each other, and/or may mean that one or more other elements are coupled or connected between the elements that are said to be coupled with each other. The term "directly coupled" may mean that two or more elements are in direct contact with one another. The term "communicatively coupled" may mean that two or more elements may be in contact with one another by a means of communication including through a wire or other interconnect connection, through a wireless communication channel or ink, and/or the like.

[0020] As used herein, the term "logic" and "module" may refer to, be part of, or include an Application Specific Integrated Circuit (ASIC), an electronic circuit, a processor (shared, dedicated, or group) and/or memory (shared, dedicated, or group) that execute one or more software or firmware programs, a combinational logic circuit, and/or other suitable components that provide the described functionality.

[0021] Also, it is noted that example embodiments may be described as a process depicted as a flowchart, a flow diagram, a data flow diagram, a structure diagram, or a block diagram. Although a flowchart may describe the operations as a sequential process, many of the operations may be performed in parallel, concurrently, or simultaneously. In addition, the order of the operations may be re-arranged. A process may be terminated when its operations are completed, but may also have additional steps not included in the figure(s). A process may correspond to a method, a function, a procedure, a subroutine, a subprogram, and the like. When a process corresponds to a function, its termination may correspond to a return of the function to the calling function and/or the main function.

[0022] As disclosed herein, the term "memory" may represent one or more hardware devices for storing data, including random access memory (RAM), magnetic RAM, core memory, read only memory (ROM), magnetic disk storage mediums, optical storage mediums, flash memory devices and/or other machine readable mediums for storing data. The term "computer-readable medium" may include, but is not limited to, memory, portable or fixed storage devices, optical storage devices, wireless channels, and various other mediums capable of storing, containing or carrying instruction(s) and/or data.

[0023] Furthermore, example embodiments may be implemented by hardware, software, firmware, middleware, microcode, hardware description languages, or any combination thereof. When implemented in software, firmware, middleware or microcode, the program code or code segments to perform the necessary tasks may be stored in a machine or computer readable medium. A code segment may represent a procedure, a function, a subprogram, a program, a routine, a subroutine, a module, program code, a software package, a class, or any combination of instructions, data structures, program statements, and the like.

[0024] The term "computer system" as used herein refers to any type interconnected electronic devices, computer devices, or components thereof. Additionally, the term "computer system" and/or "system" may refer to various components of a computer that are communicatively coupled with one another. Furthermore, the term "computer system" and/or "system" may refer to multiple computer devices and/or multiple computing systems that are communicatively coupled with one another and configured to share computing and/or networking resources. As used herein, the term "mobile device" may be considered synonymous to, and may hereafter be occasionally referred to, as a client, mobile, mobile unit, mobile terminal, mobile station, mobile user, user equipment (UE), user terminal, subscriber, user, remote station, access agent, user agent, receiver, etc., and may describe a remote user of network resources in a communications network. Furthermore, the term "mobile device" may include any type of wireless device such as consumer electronics devices, smart phones, tablet personal computers, wearable computing devices, personal digital assistants (PDAs), laptop computers, and/or any other like physical computing device that is able to connect to a communications network.

[0025] As used herein, the term "network element", may be considered synonymous to and/or referred to as a networked computer, networking hardware, network equipment, router, switch, hub, bridge, gateway, and/or other like device. The term "network element" may describe a physical computing device of a wired or wireless communication network that is configured to host a client device and the like. Furthermore, the term "network element" may describe equipment that provides radio baseband functions for data and/or voice connectivity between a network and one or more users.

[0026] Example embodiments disclosed herein provide systems and methods for bringing a user experience and transaction security aspects of a physical point of sale (POS) terminal or other like card-present transactions to card-not-present transactions, such as online transactions, phone-order transactions, and/or mail-order transactions. Embodiments provide a trusted eCommerce payment mechanism, referred to as a virtual POS (vPOS) terminal or eCommerce

POS (ePOS) terminal that combines secure payment credentials containment with capabilities to authenticate the owner of the credentials, and asserts the credentials during an online transaction. The embodiments herein models EMV Chip and personal identification number (PIN) transactions, using the concept of an online PIN to effect release of provisioned chip data. Allows card-present information to be presented at time of transaction, which may include, for example, a Primary Account Number (PAN) (or tokenized PAN), discretionary data, one or more cryptograms (e.g., application cryptogram (AC), Application Authentication Cryptogram (AAC), Authorization Request Cryptogram (ARQC), Authorization Response Cryptogram (ARPC), etc.), and the like. The example embodiments allow users to remain in control of payment credentials during card-not-present transactions. For example, most online payment methods require users to entrust their account and/or banking information to an online service provider, such as an online payment system, online money transfer service, digital wallet service, and/or other like entities. Unlike the typical online payment methods, the example embodiments allow users to maintain their own payment credentials, whether stored locally with a trusted execution environment of their own computing device or stored within a trusted cloud computing environment, and provide the payment credentials to a merchant in encrypted form (or without having to share the payment credentials in an unencrypted form). Furthermore, unlike digital wallet services (e.g., ApplePay®, Visa Checkout®, etc.), various example embodiments, may not require a server-side digital wallet (i.e., a thin wallet) to be created and/or maintained on a server for each user.

[0027] The example embodiments provide a trusted execution environment on, or embedded in, a computing device, which includes a virtual POS terminal instead of requiring a separate standalone physical POS terminal for swiping payment cards and/or entering payment card information. In various embodiments, the trusted execution environment may include one or more processors, which are separate from an application processor of the computing device, to process POS transactions. In other embodiments, the trusted execution environment may be provided as a new mode of execution on an existing processor. In some embodiments, the trusted execution environment may be provided as a cloud computing service that is separate from the user's computing devices.

[0028] In various example embodiments, the trusted execution environment also includes the various payment credentials, which may indicate one or more methods of payment associated with a user, such as credit card information, bank account information, and the like. When a merchant initiates a payment request, such as after a user performs an online checkout process, the virtual POS terminal in the trusted execution environment may perform various tasks that a standalone physical POS terminal may perform, such as PIN authorization, transaction settlement authorization, and the like. In addition to the typical functionality of a standalone physical POS terminal, example embodiments provide that each payment credential may indicate its own requirements for authenticating entities and/or processing a POS transaction.

[0029] Furthermore, according to various example embodiments, the virtual POS terminal may be accessible through a POS user interface (UI) or POS client. Additionally, a merchant may have its own POS system that allows the user or merchant to enter the user's cellphone number to initiate a transaction. For example, a web-based store may provide a UI that allows a user to directly access the virtual POS terminal via the user's own POS UI that is rendered in the user's web browser. In this way, the user may be able to provide payment using one or more payment credentials without entering payment information and/or authentication information into a web-based UI. By way of another example, a telephone or mail-order merchant may acquire payment from a user by entering the user's cellphone number into their own POS system rather than requiring the user to dictate payment card information and/or cardholder authentication information over the phone. The merchant's POS system may either call the user's cellphone or send a text message to the cellphone, which may initiate the POS transaction, and the user may use the POS UI to select a payment credential for processing the POS transaction.

[0030] Referring now to the figures. FIG. **1** shows an arrangement **100** in which a point of sale (POS) transaction may be processed using the virtual POS terminal **120** of the present disclosure, in accordance with various example embodiments. As shown in FIG. **1**, arrangement **100** may include computing system **105**, merchant domain **130**, payment acquiring service **145**, messaging service **160**, and network connections (or "links," "channels," or the like) **150** (e.g., including links **150-1** to **150-7** in FIG. **1**). Additionally, computing system **105** may include POS user interface (UI) module **110** and Trusted Compute resources (e.g., trusted execution environment (TEE) **115**). The trusted execution environment may include virtual POS terminal **120** (also referred to as "vPOS **120**," "ePOS **120**," or the like) and payment credential database (DB) **125**. As used herein, a "POS terminal" may also be referred to as a Card Acceptance Device (CAD) or Payment Acceptance Device (PAD). Furthermore, the merchant domain may include a cloud POS service **135** and a merchant service provider platform **140** (also referred to as "merchant **140**").

[0031] Computing system **105** may be physical hardware device capable of communicating with a one or more other hardware computing devices (e.g., merchant **140**, one or more devices associated with cloud POS service **135**, one or more associated databases (not shown), and the like) via a communications interface, such that computing system **105** is able to receive one or more signals and/or data streams from the other hardware computing devices. As shown by FIG. **1**, the computing system **105** includes, inter alia, a TEE **115**. An Execution Environment (EE), as used herein, is a set of hardware and software components providing facilities that support running of client applications (CAs). As examples, an EE may include a hardware processing device; a set of connections between the processing device and other hardware resources; a physical volatile memory; a physical non-volatile memory; and peripheral interfaces. In addition to the TEE **115**, the computing system **115** also includes a Rich EE (REE), which is an execution environment comprising at least one Rich (device) operating system (OS) and one or more other components of the computing system **105** (e.g., SoCs, other discrete components/hardware elements, firmware, and software) that execute, host, and support the rich OS. The rich OS a high-level OS with a rich capability set that allows consumers/users to download and run applications. Examples of rich operating systems include Microsoft® Windows®, Apple® macOS® and iOS®, Google®

Android®, Linux®, Symbian OS®, and the like. In other words, the REE includes one or more discrete elements included in the computing system **105** excluding a secure element and/or the TEE **115**. The TEE **115** is an EE that runs alongside but is isolated from the REE. The TEE **115** has security capabilities and meet certain security related requirements. The TEE **115** protects TEE **115** assets from general software attacks, defines rigid safeguards as to data and functions that a program can access, and resists a set of defined threats.

[0032] Computing system **105** may include one or more memory devices and one or more processors (not shown). Computing system **105** may be designed to sequentially and automatically carry out a sequence of arithmetic or logical operations; equipped to record/store digital data on a machine readable medium; and transmit and receive digital data via one or more network devices. Computing system **105** may be a desktop personal computer (PC), a laptop PC, a cellphone and/or smartphone, a tablet personal computer, a wearable computing device, a video game console, a digital media player, an in-vehicle infotainment (IVI) and/or an in-car entertainment (ICE) device, a handheld messaging device, a personal data assistant, an electronic book reader, an augmented reality device, and the like. It should be noted that the computing system **105** may be any physical or logical device capable of recording, storing, and/or transferring digital data via a connection to a network element.

[0033] In various embodiments, the computing system **105** may include a network interface (e.g., network interface circuitry **230** described with regard to FIGS. **2-3**) configured to connect computing system **105** to one or more other hardware computing devices wirelessly via a transmitter and a receiver (or optionally a transceiver) and/or via a wired connection using a communications port. Computing system **105** is configurable or operable to send/receive data to/from one or more other hardware computing devices, and/or network devices, such as a router, switch, hub, or other like network devices, via the network interface using the wired connection and/or the wireless connection. Computing system **105** is configurable or operable to obtain a POS transaction initiation from a network element via the network interface, and process a POS transaction based on the POS transaction initiation according to the example embodiments described herein. In embodiments where the computing system **105** includes a transmitter/receiver (or alternatively, a transceiver), computing system **105** is configurable or operable to communicate (i.e., send/receive data to/from a network element and/or other like devices) over the network connections **150** in accordance with one or more wireless communications protocols and/or one or more cellular phone communications protocols. For example, computing system **105** is configurable or operable to operate in accordance with the Global System for Mobile Communications (GSM), Enhanced Data GSM Environment (EDGE), wideband code division multiple access (WCDMA), code division multiple access (CDMA), time division multiple access (TDMA), Bluetooth and/or Bluetooth Low Energy (BLE), Wireless Fidelity (Wi-Fi) such as the Institute of Electrical and Electronics Engineers (IEEE) 802.11a, IEEE 802.11b, IEEE 802.11g, IEEE 802.11ac, IEEE 802.11ad, and/or IEEE 802.11n, voice over Internet Protocol (VoIP), Wi-MAX, Long Term Evolution (LTE), and/or any other wireless communication protocols, including RF-based, optical, and so forth.

[0034] Computing system **105** may include one or more sensors, such as an accelerometer, gyroscope, gravimeter, magnetometer, and/or another like devices. The one or more sensors is configurable or operable to detect the one or more gestures. The one or more gestures may include various touches (or combination of touches) applied to a touchscreen of the computing system **105**, one or more spatial coordinates (or changes in spatial coordinates) of positions and/or orientations of the computing system **105**. In such embodiments, the computing system **105** may track a timing and/or sequence that one or more gestures are performed, which may be used as a gesture-based passcode or password for authenticating the user's identity. In some embodiments, the one or more sensors may include a microphone configured to obtain one or more voice commands issued by a user of the computing system **105**, wherein the voice commands are used to authenticate the user's identity. In such embodiments, the verbal commands may be a passcode and/or the computing system **105** may perform voice recognition to authenticate the user. In some embodiments, the one or more sensors may include one or more biometric sensors, such as an infrared heart rate monitoring device, a fingerprint or handprint scanning device, a face and/or an eye scanning device (e.g., a camera or other like image sensor), an electromyography (EMG) device for detecting electrical patterns associated with a user's muscular contractions, an electroencephalograph (EEG) device for measuring and/or recording electrical signals produced by a user's brain, and the like. In such embodiments, biometric data detected or sensed by the one or more biometric sensors may be used to authenticate a user's identity.

[0035] Computing system **105** is configurable or operable to run, execute, or otherwise operate one or more applications. According to various example embodiments, the one or more applications may include the ePOS client **110**, the modules within the TEE **115**, and/or the TEE **115** itself. The one or more applications may include native applications, web applications, and hybrid applications. The native applications may be used for operating the computing system **105**, such as using a camera or other like sensor of the computing system **105**, GPS functionality of the computing system **105**, an accelerometer of the computing system **105**, cellular phone functionality of the computing system **105**, and other like functions of the computing system **105**. Native applications may be platform or operating system (OS) specific. Here, the term "platform" may refer to an EE within a device or computing system, such as the REE, a secure element (SE), or TEE **115**. Native applications may be developed for a specific platform using platform-specific development tools, programming languages, and the like. Such platform-specific development tools and/or programming languages may be provided by a platform vendor. Native applications may be pre-installed on computing system **105** during manufacturing, or provided to the computing system **105** by an application server (not shown) via a network. Web applications are applications that load into a web browser of the computing system **105** in response to requesting the web application from a service provider (e.g., web server **135**). The web applications may be websites that are designed or customized to run on a mobile device by taking into account various mobile device parameters, such as resource availability, display size, touchscreen input, and the like. In this way, web applications may provide an experience that is similar to a native application within a

web browser. Web applications may be any server-side application that is developed with any server-side development tools and/or programming languages, such as PHP, Node.js, ASP.NET, and/or any other like technology that renders HTML. Hybrid applications may be a hybrid between native applications and web applications. Hybrid applications may be a standalone, skeletons, or other like application containers that may load a website within the application container. Hybrid applications may be written using website development tools and/or programming languages, such as HTML5, CSS, JavaScript, and the like. Hybrid applications use browser engine of the computing system **105**, without using a web browser of the computing system **105**, to render a website's services locally. Hybrid applications may also access mobile device capabilities that are not accessible in web applications, such as the accelerometer, camera, local storage, and the like. According to various embodiments, the ePOS client **110** may be a native application, web application, or a hybrid application. In many embodiments, the TEE **115** may be a native application, which may operate in conjunction with a specialized computer processing device.

[0036] There are two client-side peers to the cloud ePOS service **135**. One peer is the ePOS client **110**, which includes or provides the ePOS user interface (UI) that renders the online ePOS experience and directs user interactions with the ePOS system. The other peer is a secure vPOS terminal **120** that interfaces with the embedded payment credentials to obtain their contents and interacts with the cloud ePOS service **135** to interpret signed authentication assertions (e.g., over connections **150-1** and **150-2**). Once the user has been successfully authenticated, the vPOS terminal **120** encrypts the obtained payment credentials and transaction signatures for subsequent transmission to upstream payment processors, such as the payment acquirer service **145** (e.g., via connections **150-2** and **150-7**).

[0037] The client system **105** may operate the ePOS client **110** may be one or more software modules that operate in conjunction with one or more hardware devices (e.g., processor **210** as described with regard to FIGS. **2-3**) to provide a user of the computing system **105** with the ability to process a POS transaction using the computing system **105**. In this way, the user of the computing system **105** may use the ePOS client **110** to interact with the vPOS **120** within the TEE **115**. The ePOS client **110** may be a client application (or "client") capable of accessing dynamic content, for example, by sending appropriate HTTP messages or the like, and in response, one or more server-side application(s) may dynamically generate and provide code, scripts, markup language documents, etc., to the ePOS client **110** to render and display graphical objects within the ePOS client **110**. The ePOS client **110** may be implemented using one or more graphical control elements, graphical icons, visual indicators, and/or text based commands. A collection of some or all of the graphical objects may be a webpage or application (app) comprising GUIs including GCEs for accessing and/or interacting with a service provider system/platform such as messaging service **160**, cloud POS service **135**, and the like. Additionally or alternatively, the collection of some or all of the graphical objects may comprise GUIs including GCEs for accessing and/or interacting with the vPOS **120** within the TEE **115**. The aforementioned server-side applications may be developed with any suitable server-side programming languages or technologies, such as PHP; Java™ based technologies such as Java Servlets, JavaServer Pages (JSP), JavaServer Faces (JSF), etc.; ASP.NET; Ruby or Ruby on Rails; and/or any other like technology that renders HyperText Markup Language (HTML). The applications may be built using a platform-specific and/or proprietary development tool and/or programming languages. UIs and/or GUIs, and their typical functionality are generally well-known, and thus, a further detailed description of the typical functionality of ePOS client **110** is omitted. However, it should be noted that according to various embodiments, the ePOS client **110** may be the only element that is outside of the trusted execution environment that is capable of communicating with elements within the TEE **115**. For example, the ePOS client **110** is configurable or operable to communicate with the vPOS **120** as shown in FIG. **1** and as described with regards to FIGS. **13-14**. In such embodiments, the ePOS client **110** may communicate with elements within the TEE **115** according to a security application programming interface (API), one or more software guard extension (SGX) instructions, and the like. Furthermore, the ePOS client **110** is configurable or operable to communicate POS transaction related data with the cloud POS service **135** as described with regards to FIGS. **13-14**.

[0038] In one example implementation, the ePOS client **110** may be an HTTP client, such as a "web browser" (or simply a "browser") capable of sending and receiving HTTP messages to and from web and/or application servers. In another example implementation, the ePOS client **110** may be a browser extension or plug-in that operates and/or interacts with an existing browser. Example browsers include WebKit-based browsers, Microsoft's Internet Explorer browser, Microsoft's Edge browser, Apple's Safari, Google's Chrome, Opera's browser, Mozilla's Firefox browser, and/or the like. In another example implementation, the ePOS client **110** may be a desktop or mobile application that runs directly on the computing system **105** without a browser.

[0039] In another example implementation, the ePOS client **110** may be an individual (e.g., stand-alone) application object that operates independently of other applications and/or interacts with other applications. In these implementations, the ePOS client **110** may operate within a security sandbox, VM, container, wrapper, or some other means for isolating the ePOS client **110** code runs in isolation from other applications. One or more data containers may also be created within the sandbox, VM, container, wrapper, etc.

[0040] In another example implementation, the ePOS client **110** is a digital wallet application (app), which is an app used to store a user's payment credentials (e.g., cryptographic private keys and/or public keys) that are associated with the user's payment instrument and/or payment credentials. In embodiments, the wallet app may store the user's credentials in the PCDB **125**. In some implementations, the wallet app may be a blockchain-based app that stores the user's credentials, and the credentials are associated with a state of a user's account in the blockchain. The wallet app also allows the user to make transactions, where the public key of the public/private key pair allows other wallets to make payments to the wallet app (e.g., using the wallet's app network address, app/wallet identifier, or the like) and the private key of the public/private key pair allows the wallet app spend currency or cryptocurrency stored by the wallet app and/or in the blockchain. When the wallet app is a blockchain app or service, the wallet app enables the user to

interact with a blockchain or other decentralized ledger or database. The blockchain app may store public and/or private keys, hash generators, encryption/decryption algorithms, and/or other like elements that can be used to generate blocks to be added to one or more blockchains. The blockchain app may also include modules, logic, program code, etc., that allow the blockchain app to validate other blocks generated by other user systems before appending those blocks to the blockchain. Additionally, where the Trusted Compute resources is/are implemented using secure enclaves (discussed infra), the wallet app can also interface directly with the secure enclave of the ePOS applet **121** via the ePOS acceptance driver **122** (discussed infra), a private transaction manager, or other like entity, and/or interface with on-chain or off-chain enclaves. The private transaction manager may be a subsystem of a specific blockchain platform/system that implements privacy and permissioning, and/or validates blocks for inclusion in a blockchain. Off-chain transactions involve the movement of value outside of a blockchain, while on-chain transactions modify the blockchain and depend on the blockchain to validate transactions. Off-chain transactions rely on other mechanisms to record and validate transactions, such as payment channels (e.g., Hashed Timelock Contracts (HTLCs) or Lightning Network), sidechains, credit/debit based solutions, trusted third parties, and the like.

[0041] The computing system **105** includes Trusted Compute resources that preserve data confidentiality, execution integrity and enforces data access policies. The Trusted Compute resources may be used to store cryptographic keys, digital certificates, credentials, payment credentials, and/or other sensitive information (e.g., personally identifying information (PII)), and could be used to operate some aspects of one or more applications. The Trusted Compute resources can be implemented using software-based cryptographic security guarantees (e.g., Zero-Knowledge Proofs), user-level or OS-level isolated instances (e.g., "containerization") or virtualization (e.g., using VMs), Trusted Multi-Party-Compute (MPC) resources, or using TEE **115**. In either implementation, applications running on the REE/host platform may be capable of interfacing with the Trusted Compute resources using a suitable (secure) application programming interface (API). Where the Trusted Compute resources is/are implemented using secure enclaves, the applications running on the REE can also interface directly with the enclave of a secure application or other like entity, and/or interface with other enclaves.

[0042] In some embodiments, the TEE **115** may be a physical hardware device that are separate from other components of the computing system **105** (e.g., as described with regard to FIG. **2**). In these embodiments, the TEE **115** is a hardware-based technology that executes only validated tasks, produces attested results, provides protection from malicious host software, and ensures confidentiality of shared encrypted data. In various embodiments, TEE **115** may include one or more physically tamper-resistant embedded chipset including processors (e.g., trusted processing core(s), trusted crypto accelerators, etc.) and memory devices (e.g., trusted RAM, trusted ROM, etc.) that communicate with an REE (e.g., host platform, application circuitry, etc.) of the computing system **105**. In such embodiments, applications that are not within the TEE **115** (e.g., ePOS client **110**) may communicate with the physically tamper-resistant embedded processors and memory devices via a security (secure) API or some other suitable interface such as those discussed in various GlobalPlatform® TEE API standards such as GlobalPlatform, *TEE Client API Specification v*1.0 (July 2010); GlobalPlatform, *TEE Internal Core API Specification v*1.2.1 (May 2019); GlobalPlatform, TEE Secure Element API v1.1.1 (November 2016); and the like. In various embodiments, the TEE **115** may include a secure cryptoprocessor, which is a dedicated computer on a chip or a microprocessor designed to secure hardware by integrating cryptographic keys into devices and/or components of the computing system **105**. In such embodiments, the TEE **115** may operate in accordance with the International Organization for Standardization (ISO) and International Electrotechnical Commission (IEC) specification ISO/IEC 11889:2009. In some embodiments, the TEE **115** may have an architecture in accordance with the GlobalPlatform, TEE System Architecture standard version 1.2 (November 2018). Examples TEEs **115** may include a Desktop and mobile Architecture Hardware (DASH) compliant Network Interface Card (NIC), Intel® Management/Manageability Engine, Intel® Converged Security Engine (CSE) or a Converged Security Management/Manageability Engine (CSME), a financial-grade Secure Element microcontroller, Trusted Execution Engine (TXE) provided by Intel® each of which may operate in conjunction with Intel® Active Management Technology (AMT) and/or Intel® vPro™ Technology; AMD® Platform Security coProcessor (PSP), AMD® PRO A-Series Accelerated Processing Unit (APU) with DASH manageability, Apple® Secure Enclave coprocessor; IBM® Crypto Express3®, IBM® 4807, 4808, 4809, and/or 4765 Cryptographic Coprocessors, IBM® Baseboard Management Controller (BMC) with Intelligent Platform Management Interface (IPMI), Dell™ Remote Assistant Card II (DRAC II), integrated Dell™ Remote Assistant Card (iDRAC), and the like. An example of such embodiments is shown and described with regards to FIG. **2**.

[0043] The TEE **115** is not constrained to an embedded platform or subsystem, and in various embodiments, the TEE **115** may be implemented as access card circuitry including, for example, an integrated circuit card (ICC) (also referred to as a "chip card" or "smart card"), universal ICC (UICC), Subscriber Identity Module (SIM) card, embedded UICC (eUICC), a form-factor secure element (e.g. SD cards, smart microSDs, USB tokens, etc.), and/or the like. In these embodiments, the access card circuitry contains the ePOS acceptance driver **122** and ePOS applet **121** (discussed infra). Such embodiments allow the same payment instrument to be used in multiple platforms such as desktop computers, laptops, tablets, smartphones, and the like. In some implementations, the access card is removable or detachable security circuitry (e.g., smart card, UICC, SIM card, etc.), and the client system **105** may include a card reader or card slot that is an input device configured to accept the access card (e.g., usually card shaped) and reads data from the access card's storage medium. Such access cards are usually implemented as a plastic (e.g., PVC) card containing semiconductors and embedded contacts, which can be inserted into and communicatively coupled with corresponding contacts in the card reader. In some implementations, the access card circuitry is non-removable access card (e.g., eUICC, SoC-based secure element, etc.), where the access card circuitry is embedded on a motherboard of the computing system **105** or on some other chip or

component in the computing system **105**. In either implementation, the access card may perform various access card functions, such as those defined by ISO/IEC 7816, European Telecommunications Standards Institute (ETSI) technical standard (TS) 102 221, ETSI TS 102 225, ETSI TS 102 226, ETSI TS 102 241, ETSI TS 102 600, 3GPP TS 31.101, 3GPP TS 31.116, 3GPP TS 31.121, 3GPP TS 31.220, 3GPP TS 31.828, Groupe Speciale Mobile Association (GSMA), "Remote Provisioning Architecture for Embedded UICC," TS version 3.0, 30 Sep. 2015, and/or any other like standards. Additionally, the access card (e.g., SIM card) may securely stores access network information (e.g., an international mobile subscriber identity (IMSI) number and related keys or access credentials) which is used to identify and authenticate subscribers using radio equipment (e.g., smartphones, satellite phones, tablets, wearables, etc.). The access network information (or SIM) is registered and activated by a mobile network operator (MNO), and the access network information is then used to authenticate the subscriber during a connection or session establishment procedure.

[0044] In other embodiments, the TEE **115** may be one or more software modules that operate in conjunction with one or more hardware devices (e.g., as described with regard to FIG. **3**) to carry out secure operations and/or control the storage and use of personal and/or confidential data. In these embodiments, the TEE **115** may be one or more "enclaves" (also referred to as "secure enclaves") within the memory of the computing system **105**. An "enclave" is an instantiation of trusted compute resources within a hardware based environment, such as the REE (e.g., host platform, application processor circuitry) or within a hardware-based TEE **115**. For purposes of the present disclosure, the terms "TEE" and "enclave" may be used interchangeably. Secure enclaves may be isolated regions of code and/or data within the address space of a secure application. In most embodiments, only code executed within a secure enclave may access data within the same secure enclave, and the secure enclave may only be accessible using the secure application, and other applications may access the secure enclave via the secure application. Certain hardware based enclaves allow multiple instances of enclaves to execute concurrently. Examples of the secure enclave-based TEE **115** may include secure enclaves defined using the Intel® Software Guard Extensions (Intel® SGX), Keystone Enclaves provided by Oasis Labs™ secure/trusted zones using ARM® TrustZone® hardware security extensions, and/or the like. An example of such embodiments is shown and described with regards to FIG. **3**. Other aspects of security hardening, hardware roots-of-trust, and trusted or protected operations may also be implemented in the device **105** through the TEE **115** and/or the processor circuitry of the device **105**.

[0045] The TEE **115** operates a Trusted OS, which is an OS running in the TEE **115** that is designed using security based design techniques to enable the TEE **115**. The trusted OS provides TEE Internal APIs to Trusted Applications (TAs) and a proprietary methods to enable a TEE Client API to interface with applications operating within other EEs, such as the POS client **110** operating within the REEs. TAs are applications that run inside the TEE **115** that provide security related functionality to CAs outside of the TEE **115** or to other TAs inside the TEE **115**. The TAs communicate with the rest of the system via APIs exposed by Trusted OS components. TEE internal APIs define the fundamental software capabilities of a TEE **115** and TEE client APIs define interfaces for client applications to access or communicate with TAs. When a CA creates a session with a TA, the CA connects to an instance of the TA. A TA instance has physical memory address space which is separated from the physical memory address space of all other TA instances. A session is used to logically connect multiple commands invoked in a TA. Each session has its own state, which contains the session context and the context(s) of the task(s) executing the session.

[0046] As shown in FIG. **1**, the TEE **115** includes vPOS **120** and payment credential database (DB) **125**, which are TAs within the TEE **115**. The vPOS **120** aggregates payment instrument(s) and certain acquiring elements of POS terminals (e.g., CADs, PADs, etc.) within a single platform (e.g., the TEE **115**). The vPOS **120** may be one or more software element, engine, applet, modules, and/or other like collection of functions that operate in conjunction with one or more hardware devices (e.g., processor circuitry **310** as described with regard to FIG. **2**, processor circuitry **210** as described with regard to FIG. **3**, and the like) to interface with embedded payment credentials to obtain payment credential information, and interact with the ePOS client **110** and/or other like secure applications to obtain user inputs and information from the cloud POS service **135**. The vPOS **120** may also validate and/or decrypt the user inputs and the information received from the cloud POS service **135** in order to authenticate a user of the computing system **105** and/or the merchant **140**. Once a user has been successfully authenticated, the vPOS **120** may encrypt the obtained payment credential information and/or transaction signatures for subsequent transmission to an upstream payment processor (e.g., payment acquiring service **145**). Furthermore, the vPOS **120** may generate POS transaction messages (which may be communicated via the ePOS client **110**) in accordance with ISO 8583, which defines a standard for systems that exchange electronic transactions made by cardholders using payment cards.

[0047] Payment credential database (PCDB) **125** may be a hardware device or system for storing payment credentials and payment credential-related information for a plurality of payment credentials. The PCDB **125** may also be referred to as a "payment credential storage unit," "payment system environment" (PSE), or the like. In various embodiments, the payment credentials may be associated with a payment card issued to users for paying for goods and services (e.g., a credit card, charge card, debit card, electronic benefits transfer (EBT) cards, etc.). The payment credentials may be a digital/electronic version of a physical payment card, or the payment credentials may be digital/electronic payment credentials. The PCDB **125** may also store cardholder data or other like sensitive information belonging to an authorized user of a payment card, which may include a cardholder name, billing address, shipping address, payment card number, PAN, PIN, verification codes, security questions and answers (e.g., cardholder birthdate, mother's maiden name, etc.), and the like. Furthermore, the PCDB **125** may store payment credentials and authorization information in accordance with EMV (Europay, MasterCard, and Visa) standards, which define worldwide interoperability and acceptance standards for secure card-present and card-not-present transactions. Moreover, in some embodiments, the PCDB **125** may store virtual currency information (e.g., Linden Dollars traded in the virtual online community

Second Life®, one or more currencies exchanged in the massively multiplayer online role-playing game (MMORPG) World of Warcraft®, Amazon Coins®, Nintendo Points®, Facebook® Credits, frequent flyer miles, brick-and-mortar store rewards points, etc.), cryptocurrency information (e.g., bitcoins, litecoins, etc.) and/or other like information used as a medium of exchange. In such embodiments, the payment credentials stored in the PCDB **125** may include one or more digital currency wallets (e.g., a bitcoin wallet, etc.) and/or any other like mechanism for storing information necessary to account for digital currency transactions. In some embodiments, the PCDB **125** may store a Data Object List (DOL) for each payment credential. The DOLs specify the data that a corresponding credential applet **121** expects as inputs and will provide as (signed) outputs during each step of the transaction processing procedure/protocol, as well as the format for the data. The DOLs indicate the various data elements for each part of a transaction, the format and/or arrangement of the data elements to be used for each part of the transaction, and which data elements should be signed or encrypted. Example data elements include the Application Transaction Counter (ATC), transaction amount(s), currency type(s), country code, financial institution information, and payment credential or credential applet-chosen nonces.

[0048] PCDB **125** may include one or more relational database management systems (RDBMS) one or more object database management systems (ODBMS), a column-oriented DBMS, correlation database DBMS, and the like. According to various example embodiments, the PCDB **125** may be stored on or otherwise associated with one or more data storage devices. These data storage devices may include at least one of a non-volatile random-access memory (NVRAM) device, a read-only memory (ROM) device, a flash memory device, a solid state drive (SSD), and/or other like data storage devices. In various embodiments, PCDB **125** may be accessed by the vPOS **120** such that the vPOS **120** may query the PCDB **125** to obtain payment credential information and/or store payment credential information in the PCDB **125**. In some embodiments, PCDB **125** may be physically and/or logically divided into multiple databases, while in other embodiments, the PCDB **125** may be a single database that resides on one physical hardware data storage device.

[0049] The TEE **115** may also include one or more Security Domains (SDs) (not shown by FIG. **1**). An SD is an on-device representative entity of an Authority in the TEE **115**. SDs are responsible for the control of administration operations, and are used to perform the provisioning of TEE properties and manage the life cycle of TAs. An Authority is an entity or actor that grants permission to perform a specific set of actions, such as the cloud POS service **135**, the merchant **140**, the payment acquiring service **145**, or the auth service **170**. The SDs allow the device **105** to be managed by the associated Authority and can then be placed in in either a secured state (e.g., "TEE_SECURED") or a locked state (e.g., "TEE_LOCKED"). In the secured state, the TEE **115** has been configured with at least one SD including personalization, keys, and data in order to perform administration operations. The locked state disables new accesses from CAs to any TAs within the TEE **115**. In the locked state, any attempt by a CA to open a new session with a TA is rejected and provided with an error message/code. Once the TEE **115** is locked, only an SD having the necessary privileges is allowed to unlock the TEE **115**. The unlock operation switches the TEE **115** from the locked state back to the secure state. Similarly, the SD may be in an accessible state or a block state. In the accessible state, the SD is fully operational, and can be used by CAs (active state) or is temporarily suspended to perform some maintenance operations (restricted state). In the blocked state, the SD can neither be used nor administrated until it is unblocked. The TAs may be in an active state or inactive state. In the active state, a TA can be accessed and used by CAs, and if the TEE is in the locked state, the TA is temporarily suspended to perform some maintenance operations. In the inactive state, the TA is directly controlled by an associated SD and can neither be used nor administrated until its SD is unblocked.

[0050] Merchant domain **130** may be a collection of devices, systems, and/or services that are utilized by a merchant (e.g., merchant **140**) to engage with ecommerce-related activities. As shown merchant domain **130** may include merchant server **140** and cloud POS service **135**.

[0051] Merchant server **140** (also referred to as "merchant **140**") may be one or more hardware computing devices that may include one or more servers or other like network elements for providing one or more services. The one or more services may include selling products and/or services, providing an online market place for the purchase and/or sale of products and services, mining demographic data, online marketing, and/or other like ecommerce-related services. In this regard, the merchant **140** may engage with a user of the computing system **105** to sell products and/or services to the user, and utilize the cloud POS service **135** to initiate a POS transaction with the computing system **105**. The merchant server **140** may include an operating system that may provide executable program instructions for the general administration and operation of merchant server **140**, and may include a computer-readable medium storing instructions that, when executed by a processor of the merchant server **140**, may allow the merchant server **140** to perform its intended functions. Suitable implementations for the operating system and general functionality of the servers are known or commercially available, and are readily implemented by persons having ordinary skill in the art, particularly in light of the disclosure herein. Furthermore, it should be understood that the merchant **140** may not be required and the applications and software components discussed herein may be executed on any appropriate device or host machine.

[0052] Cloud POS service **135** may be one or more hardware computing devices that may include one or more servers or other like network elements for providing one or more services. The one or more services provided by the cloud POS service **135** may include initiating a POS transaction with a client device that is enabled to process POS transactions (e.g., computing system **105**), authenticating payment information and/or user identity information, and providing POS transaction-related information to the computing system **105** to process a POS transaction. In this regard, cloud POS service **135** may communicate data (i.e., transmit and receive) with ePOS client **110**, and thus, may communicate to vPOS **120** securely and indirectly through the ePOS client **110**. Additionally, the cloud POS service **135** may communicate data (i.e., transmit and receive) with the merchant **140** and/or the payment acquiring service **145** purposes of processing POS transactions. In order to com-

municate with the ePOS client **110**, the merchant **140**, and/or the payment acquiring service **145**, the cloud POS service **135** may generate POS transaction messages (which may be communicated to the ePOS client **110**) in accordance with ISO 8583, which defines a standard for systems that exchange electronic transactions made by cardholders using payment cards.

[0053] The cloud POS service **135** may be integrated with a payment system that is implemented by the merchant **140**, and may capture POS transaction details for analysis and reporting to the merchant **140**. In some embodiments, the cloud POS service **135** may be a standalone service such that the cloud POS service **135** is implemented separately from the merchant domain **130**. In this way, the cloud POS service **135** may be hosted by an independent entity that may provide the cloud POS service **135** to a plurality of merchant services simultaneously. It should be noted that typical digital wallet services and/or online money transfer services may be hosted by an independent entity that provides their services to a plurality of merchants simultaneously. However, unlike the typical digital wallet services and/or online money transfer services, in various embodiments the cloud POS service **135** may not have to retain user payment information and user identification information. Instead, the cloud POS service **135** may allow users to retain possession and control of their personal account information and personally identifying information. Therefore, the cloud POS service **135** may provide network and computational resource efficiencies while also providing user privacy protections. Furthermore, because typical card-not-present transactions usually have a higher rate of fraud, merchants are usually charged higher bank processing fees and/or are required to absorb costs associated with data breaches and/or fraudulent transactions. Therefore, merchants are usually responsible for processing payments and providing security measures for handling payment information for typical card-not-present transaction. Accordingly, merchants that utilize the cloud POS service **135** could end up paying a lower transaction fee to process payments through the cloud POS service **135** (including the vPOS **120**) rather than processing payments on their own.

[0054] The cloud POS service **135** may include an operating system that may provide executable program instructions for the general administration and operation of payment acquiring service **145**, and may include a computer-readable medium storing instructions that, when executed by a processor of the application server **130**, may allow the payment acquiring service **145** to perform its intended functions. Suitable implementations for the operating system and general functionality of the servers are known or commercially available, and are readily implemented by persons having ordinary skill in the art, particularly in light of the disclosure herein. Furthermore, it should be understood that the payment acquiring service **145** may not be required and the applications and software components discussed herein may be executed on any appropriate device or host machine.

[0055] In embodiments, a payment kernel (e.g., the algorithm dictating the protocol interaction with the payment instrument) may run in the cloud **135** (not shown by FIG. **1**) while the acceptance endpoint (e.g., ePOS applet(s) **121**) runs within the confines of the secure TEE **115** platform. In some embodiments, the payment kernel maintains trusted state synchronization with the acceptance endpoint. This may be done over a network connection **150** or with an associated SD in the TEE **115** (e.g., the TEE_SECURED state, SD blocked state, or TA inactive state). The TEE **115** includes one or more payment credentials (e.g., in PCDB **125**), and also one or more ePOS applets **121** (also referred to as "credential applets" or the like). The ePOS applets **121** are responsible for providing internal and external access to the secure element instance (e.g., Trusted OS instance), processing account data, performing cardholder verification, and the like. The ePOS applets **121** may be implemented as widgets such as plug-ins, applets, desk accessories, or other like software component(s). A widget (applet) is a relatively small application that performs a specific task and runs within a secure OS, dedicated widget engine, or as part of a larger application. A widget engine is the software platform on which desktop or web widgets run. In some embodiments, the ePOS applet **121** may be implemented as a smartcard payment applet compatible with EMV standards, GlobalPlatform standards, and the like. The TEE **115** may include and/or operate a Trusted OS such as a secure OS, real-time OS (RTOS), widget engine, VM manager, or other like software element that that controls and manages access to the TEE **115** and/or the ePOS applet **121**. A Trusted OS is an operating system running in the TEE providing a TEE internal core API to TAs. In one example implementation, the Trusted OS may be the Java Card™ OpenPlatform (JCOP), which is a standardized smart card OS. The JCOP may include a Java Card™ Virtual Machine (JCVM) and/or implement a Java Card™ Runtime Environment (JCRE), which allow Java applets to run on UICCs and eUICCs. In other embodiments, the Trusted OS may be a native OS, which is a proprietary vendor-specific OS. In either embodiment, the Trusted OS may implement a profile loader (also referred to as a "profile installer" and/or a "profile manager") for installing and managing provisioned profiles, applets, and/or payment credentials. In various embodiments, vPOS **120** may include multiple ePOS applets **121** that are selectable and applied at the time of committing to a transaction (e.g., making payment).

[0056] The ePOS applet(s) **121** is/are personalized to contain the payment credentials in a same or similar manner to an applet contained on a standard payment card or standard access card. The personalization may be provided using, for example, Store Data data grouping identifiers (DGIs) and/or secure channel protocols. Personalization may also take place according to the EMV Card Personalization Specification version 1.0 (June 2003) and/or other like specifications. When there is a request for payment, the cloud payment kernel, interacting with the platform payment acceptance endpoint (e.g., TEE **115**), communicates with the embedded ePOS applet **121** to consummate the payment transaction. From the perspective of the payment acquiring service **145**, the transaction is identical to a payment transaction initiated between a conventional POS terminal interacting with a customer payment card. This provides assurance that the actual card is present at the time of transaction rather than a falsified card or malicious code.

[0057] Another aspect of a secure transactions is to prove that the authorized cardholder is present at the time of transaction. In general face-to-face payment card transactions, there are four levels of transaction security associated with cardholder presence. In increasing confidence of cardholder authenticity these include: No Verification, Signature, Local PIN, and Online PIN. In various embodiments, indi-

vidual confidence levels of user authentication (e.g., depending on platform capabilities) are mapped to each of the cardholder authenticity four levels, providing compatibility with existing EMV user verification methods. In other embodiments, multiple confidence levels may be mapped to the same cardholder authenticity level to provide a more granular approach to verifying user authenticity locally at the computing system **105**. The combination of card authenticity coupled with credential authentication yields assurances that are compliant with traditional card-not-present situations. When such assurances are successfully asserted, merchants **130** benefit from decreased fraud and lower cost of payment acceptance, thereby reducing overall network and computational resource consumption. Payment card issuers and processing networks also benefit from increased transaction volume since there are significant numbers of people who refuse to participate in eCommerce due to fears surrounding fraud and/or identity theft. Moreover, these embodiments may be applied to other scenarios, such as preventing SIM swapping and/or SIM hijacking.

[0058] In various embodiments, the ePOS acceptance driver **122** provides a bridge interface between the cloud ePOS kernel in cloud **135** and the ePOS applet **121** executing inside of the TEE **115**. In some embodiments, the ePOS driver **122** includes terminal logic and/or device pairing logic to maintain trusted state synchronization with the POS cloud **135**. Security of the transaction is provided by the ePOS acceptance driver **122**, which uses transaction-specific encryption to provide confidentiality of sensitive transaction elements. One example approach combines format-preserving-encryption with transaction-specific keying available with standards such as ANSI X9.24-1 (DUKPT) to provide transaction data confidentiality. Other encryption schemes may be used in other embodiments.

[0059] FIG. **1** also shows elements of the ePOS client **110**, which supports both user authentication mechanisms, plus ePOS logic **113** to support payment interactions and user interfaces. The ePOS logic **113** is factored between the merchant **140** and the credential hosting environment (e.g., vPOS **120**). The authentication mechanisms may be referred to as "cardholder verification methods" or "CVMs" in EMV terminology. there are three options for PIN-based cardholder verification: online PIN where the PIN is provided to the cloud POS service **135** and/or the auth service **170** for authentication; offline unencrypted PIN where the PIN is sent to the ePOS applet **121** without being encrypted; and offline encrypted PIN where the PIN is encrypted before being sent to the ePOS applet **121**. In conventional CVM, the access card is only involved in cardholder verification when offline PIN is used, and online PIN verification is performed using a protocol between the conventional POS terminal and the financial institution (e.g. service **145**). If an access card supports both online and offline PIN CVMs, the issuing financial institution usually has to have mechanisms in place to ensure that the two PINs are synchronized. Synchronization is important, because when cardholders are asked to enter a PIN, they do not know whether they should enter their offline PIN or online PIN. By contrast, in various embodiments, the vPOS **120** is involved in online PIN verification at least to securely pass the PIN to the cloud **135** and/or auth service **170**, which is more secure than conventional CVM. Although the present disclosure discusses using a PIN for user authentication/verification, which is usually four, six, or twelve digits, it should be noted that a passcode, password, biometric data, physical hardware mechanism, and/or any other type of authentication means may be used in place of a PIN in various embodiments.

[0060] Locally controlled (offline) user authentication is provided through the secure PIN client **112**, which provides a PIN (e.g., input by the user) to the ePOS driver **122** over interface **119**. In these embodiments, the PIN are verified directly by credential applet **121**. For plaintext (unencrypted) PIN verification, the PIN code is sent to the ePOS applet **121** via the ePOS driver **122**, and the ePOS applet **121** returns an unauthenticated response to the secure PIN client **112** via the ePOS driver **122**. The unauthenticated response indicates whether the PIN was correct or how many failed PIN attempts there are left before the ePOS applet **121** blocks access to the selected payment credential. Additional or alternative security measures may be taken when the maximum number of attempts is reached. If encrypted PIN verification is used, the ePOS applet **121** requests a nonce from the ePOS driver **122** and/or the secure PIN client **112**. The ePOS applet **121** uses a public key of associated with the selected payment credential, encrypts the PIN together with the nonce and some random padding created by the ePOS applet **121** (e.g., using a suitable random number generator, hash function, etc.). A result of the verification may be returned to the PIN client **112** and displayed to the user in some embodiments.

[0061] Online user authentication is managed through an external authentication (auth) service **170** via the auth client **111**. In these embodiments, the online PIN is verified by auth service **170**, and online service authentication assertions are received via connection **150-1** for verification by the ePOS **121**. In some embodiments, the merchant **130** that hosts the ePOS payment kernel (e.g., within cloud POS service **135** or in some other merchant **130** system or platform), and who is also a relying party to the auth service **170** is responsible for providing levels of user authentication relevant to the value of the transaction. In some embodiments, the auth client **111** provides the online PIN to the auth service **170** via connection **150-5**. In some embodiments, the auth client **111** provides the online PIN to the cloud POS **135** over connection **150-3**, which is then relayed to the auth service **170** over connection **150-4**. In one implementation, the 'what you see is what you get' (WYSIWYS) digital signature techniques may be used to capture and protect the online PIN for verification by the ePOS applet **121**. In one example implementation, security of the online PIN entry may be achieved by using Intel® Identity Protection Technology (IPT), which uses Protected Transaction Display (PTD) to protect Public Key Infrastructure (PKI) certificates and RSA keys with a PIN. The PTD may be used to confirm transaction amounts and/or other transaction information.

[0062] The user authentication mechanisms (e.g., auth client **111** and/or secure PIN client **112**) can include multiple and varying factors, including, but not limited to user name/ID and password, PIN, location-based, network-based (e.g., virtual private network (VPN), etc.) security tokens, physical authentication devices (e.g., wearable continuous authentication using a smartwatch or the like, YubiKey® provided by Yubico® or the like), and/or biometric data. Additionally or alternatively, authentication strategies based on NIST SP800-63 Electronic Authentication Guidelines may be used.

[0063] The cloud ePOS kernel is also interconnected to a payment processing network **145** via connection **150-7**

between the cloud ePOS service **135** and payment acquirer service **145**. This element understands standard payment protocols such as ISO 8583 and the like, common to the payment processing ecosystems. Interactions between the ePOS cloud **135** and client ePOS elements ensure that data elements are present in the transaction such that the payment networks **145** comprehend ePOS transactions to be equivalent to card-present payments and/or the like.

[0064] In some embodiments, the use of a contactless payment kernel hosted in the TEE **115** (e.g., a secure enclave) may communicate with the ePOS Acceptance Driver **122**, and the ePOS Acceptance Driver **122** may be communicatively coupled with an NFC controller implemented in the computing system **105** (not shown). Such embodiments allow the payment instrument to be, or act as, an external contactless card such as an Apple Pay® contactless device. In these embodiments, the cloud ePOS **135** directs the actions of the client-hosted payment kernel **121** to facilitate acceptance of existing contactless payment instruments.

[0065] As alluded to previously, the strength of user authentication is mapped to an appropriate value of the transaction. This capability is provided by auth service **170**. In one example, transaction inputs (e.g., EMV commands and/or auth service **170** assertions) are conveyed to the vPOS **120** via connection **150-1**, and transaction outputs (e.g., cardholder authentication data, ISO 8583 transaction data, EMV responses, payment credentials, online PIN block, etc.) are provided to the ePOS cloud **135** via connection **150-2**. In some implementations, the auth service **170** may be provided by Intel® True Key (or "You Are the Password" or "YAP"). For example, YAP provides scoring levels associated with the strength of a given authentication. Mapping YAP scoring levels to EMV assurance categories of No Verification, Signature, Local PIN, or Online PIN, allows ePOS authentication/verification to be used with existing payment processing attributes designed to assert these various levels of user/cardholder verification. When combined with EMV metadata elements that bind required authentication strength to transaction value, the ePOS system **100** provides both card issuers and merchants **130** the controls necessary to dynamically manage risk.

[0066] The various entities, elements, systems, devices, etc., in arrangement **100** are communicatively coupled via various network connections **150** (e.g., including connections **150-1** to **150-7** in FIG. **1**), which may be part of any of one or more networks (referred to as "network **150**") that allow computers to exchange data. Network **150** may include one or more network elements (not shown) capable of physically or logically connecting computers. The network **150** may include any appropriate network, including an intranet, the Internet, a cellular network, a local area network (LAN), a wide area network (WAN), a personal network or any other such network or combination thereof. Components used for such a system can depend at least in part upon the type of network and/or environment selected. Protocols and components for communicating via such a network are well known and will not be discussed herein in detail. Communication over the network **150** may be enabled by wired or wireless connections, and combinations thereof.

[0067] Payment acquiring service **145** may be include one or more physical and/or virtual computing systems such as one or more servers or other like network elements for providing one or more services. The payment acquiring service **145** may be a financial institution, bank, and/or other like entity or group of entities that is able to process payment credential purchases on behalf of a merchant platform (e.g., merchant **140**). In some embodiments, the payment acquiring service **145** may be referred to as a "merchant bank", "acquirer", and the like. The payment acquiring service **145** may be contacted by the cloud POS service **135** to authorize a payment card purchase amount. The payment acquiring service **145** may be able to approve or decline a payment credential purchase amount, and if approved, the payment acquiring service **145** may settle the transaction by placing the funds into an account associated with the merchant **140**. The payment acquiring service **145** may include an operating system that may provide executable program instructions for the general administration and operation of payment acquiring service **145**, and may include a computer-readable medium storing instructions that, when executed by a processor of the application server **130**, may allow the payment acquiring service **145** to perform its intended functions. Suitable implementations for the operating system and general functionality of the servers are known or commercially available, and are readily implemented by persons having ordinary skill in the art, particularly in light of the disclosure herein.

[0068] Messaging service **160** may be one or more hardware computing devices that may include one or more servers or other like network elements for providing one or more services. The messaging service **160** may be a service that forwards or otherwise provides notifications and/or other like messages from third parties to client devices (e.g., computing system **105**). In such embodiments, the notifications or messages may be push notifications communicated in accordance with the Push Access Protocol (PAP), simple mail transfer protocol (SMTP), Extensible Messaging and Presence Protocol (XMPP), and/or other like push communication protocols. Such notifications or messages may include audio/video messages, text alerts, and/or the like. The messaging service **160** may be contacted by the cloud POS service **135** to provide a POS transaction initiation message to the computing system **105** on behalf of the cloud POS service **135**. In various embodiments, the messaging service **160** may be a mobile messaging service, such as Google Cloud Messaging (GCM), Apple Push Notification Services and/or Notification Center, Windows Push Notification Service (WNS), and/or other like messaging services. In other embodiments, the messaging service may be a Short Message Service (SMS) Function (SMSF) that sends SMS messages to cellular network subscribers (e.g., computing system **105**) on behalf of cloud POS service **135**.

[0069] The arrangement **100** may be implemented according to one or more of the following use cases. There are multiple use cases that can benefit from arrangement **100** and several of these are listed below, although these use cases is not an exhaustive list, and the embodiments herein can be employed for various other use cases.

[0070] Use case 1: ecommerce or web-based purchases. According to use case 1, the vPOS **120** may be accessible by rendering the ePOS client **110** in a standard web browser. When the user of the computing system **105** goes through an ecommerce checkout process, the user may be asked to make an ecommerce payment. At this point, the cloud POS service **135**, operating in conjunction with the ecommerce checkout process provided by the merchant **140**, may solicit

the POS transaction through the ePOS client **110** rendered in the web browser. A variant of use case 1 may include out-of-band ecommerce transactions, such as those performed on untrusted or otherwise unsecure computing devices. In such instances, when the user is asked to provide payment, the user may provide a unique identifier (e.g., e.g., mobile phone number, an email address, and/or any other user-related identifier) associated with a computing device enable to process POS transactions (e.g., computing system **105**) instead of entering payment information directly into a web browser of an untrusted computing device. Once the user enters the phone number, email address, etc. of the computing device enabled to process POS transactions the transaction initiation message may be sent to the computing system **105** via the cloud POS service **135** and/or messaging service **160**, instructing computing system **105** to process the POS transaction.

[0071] Use case 2: mail-order and/or telephone-order purchases. Telephone-order transactions traditionally require customers to dictate payment information over the phone to an agent of a merchant in order to purchase a product or service. According to use case 2, a user of the computing system **105** may order a product or service over the phone. However, instead of requiring the user to dictate payment information over the phone, the user may provide a unique identifier (e.g., mobile phone number, an email address, and/or any other user-related identifier) to the merchant **140** (or an agent of the merchant **140**). The merchant **140** (or an agent of the merchant **140**) may enter the unique identifier into a merchant terminal that is enabled to interact with the cloud POS service **135**, and the cloud POS service **135** and/or messaging service **160** may send a transaction initiation to the computing system **105**. The POS transaction may then be processed according to the various example embodiments disclosed herein. For mail-order purchases, which typically require a customer to write down payment information on a paper order form, the customer may manually write the unique identifier on the paper order form instead of writing payment information and/or authentication information into the order form. When the order form reaches the merchant **140**, the unique identifier may be entered into a merchant terminal enabled to interact with the cloud POS service **135**, which may immediately send a transaction initiation to the computing system **105** via the cloud POS service **135**. When the merchant receives payment confirmation from the computing system **105**, the merchant **140** may fulfill (e.g., ship) the ordered product or service.

[0072] Use case 3: brick-and-mortar purchases. Most brick-and-mortar transactions require customers to enter payment information into a standalone physical POS terminal (also referred to as a credit card terminal, payment terminal, electronic funds transfer (EFT) POS terminal, etc.) to purchase a product or service. In most cases, the standalone physical POS terminals allow customers or merchants to insert, swipe, or manually enter the required payment information. According to use case 3, the user of the computing system **105** may provide a unique identifier to a brick-and-mortar store employee, who may then enter the unique identifier into a merchant terminal enabled to interact with the cloud POS service **135**.

[0073] A variant of use case 3 may involve other physical outlets that may not have a typical standalone physical POS terminal. These other physical outlets may include merchants at farmers' markets, trade shows, swap meets, flea markets, bazaars, conventions, concerts, and the like. Typically, merchants of these other physical outlets may have a computing device, such as a smartphone or tablet PC, which may mimic or replace the functionality of the standalone physical POS terminal hardware using a terminal application running on a the computing device. These terminal applications usually require manual entry of payment information, or may operate in conjunction with a peripheral hardware payment card reader that can transfer payment information to the terminal application. Such peripheral payment card readers may connect to the merchant's computing device through an audio jack of the computing device. Since these terminal applications typically mimic the standalone physical POS terminals, they often present the same or similar security concerns that apply to the standalone physical POS terminals. According to the variant of use case 3, the merchant's computing device may include its own POS UI that may interact with the cloud POS service **135**, and a user of the computing system **105** may manually enter or otherwise provide a unique identifier to the merchant's POS UI to initiate the POS transaction via the cloud POS service **135**.

[0074] Use case 4: user-initiated donations. In some instances, a user may wish to make a donation or tithe to an entity, such as a non-profit organization. Rather than requiring the donation recipients to manually process card requests using standalone physical POS terminals or a terminal application as discussed with regard to use case 3, according to use case 4, the ePOS client **110** may allow a user of the computing system **105** to initiate and complete payment requests on their own. In various embodiments, the donation recipient may deploy a Bluetooth low-energy (BLE) beacon that broadcasts one or more signals, which indicate that a POS payment may be made at a physical location. When the computing system **105** obtains the one or more BLE signals, the computing system **105** may indicate that a POS payment point associated with the donation recipient is nearby, and the user of the computing system **105** may execute the ePOS client **110** to initiate a POS transaction according to the example embodiments disclosed herein. It should be noted that a variant of use case 4 may involve deploying BLE beacons in brick-and-mortar stores or other like physical outlets to facilitate the purchase of products or services, rather than requiring an employee to perform a checkout process and the like.

[0075] Use case 5: Out-of-band eCommerce payments are possible using the ePOS mechanisms described in use case 2, and where a web checkout form offers an out-of-band ePOS payment capability. By entering a mobile phone number in the web checkout form, the merchant eCommerce system **140** provides a mechanism to redirect the eCommerce payment operation to the computing system **105** (e.g., using messaging service **160** or the like). This option is more secure from the perspective of not providing payment information through a computer that may not have secure payment acceptance capabilities (e.g. a public computer).

[0076] Use case 6: protection against SIM hijacking or SIM swapping scams. At its most basic level, a SIM swap or hijack is a type of account takeover fraud where an attacker convinces a victim's mobile carrier to switch the victim's phone number over to a SIM card own by the attacker. This allows the attacker to divert incoming messages (e.g., SMS messages) to the attacker's device so the attacker can easily

complete text/SMS-based two-factor authentication to gain access to the victim's online accounts and/or other protected sensitive data or applications. It is conceivable that this type of fraud could be attempted for the vPOS **120**, where an attacker may convince a financial institution to switch or re-provision a victim's payment credentials to a SIM card or TEE **115** in a device owned by the attacker, such as when, the TEE **115** is a UICC, SIM card, or other access card that includes the vPOS **120** and/or PCDB **125**. Usually, the contents/data stored by an access card (e.g., SIM card) cannot be perfectly copied/cloned, and therefore, swapping an access card with a different physical or virtualized access card would cause an authentication failure in the backend of the cloud system **135** and/or auth system **170**. This is because the original vPOS **120** would maintain a trusted state synchronization (e.g., secured state, blocked state, or inactive state) with the cloud system **135** and/or auth system **170** (or an associated SD in the TEE **115**) that cannot be duplicated with to a new or different access card or TEE **115**. In this use case, when the trusted state synchronization with the cloud system **135** and/or auth system **170** is broken, the TEE **115** may deny access to the vPOS **120** (e.g., by placing the TEE **115** in the locked state) until the trusted (secure) state is reestablished. Additionally or alternatively, a PIN may be set to access the vPOS **120**, which may be the same or different than the PINs set to access individual payment credentials within the vPOS **120**. The victim's PIN(s) may still be maintained in the emulated or stolen SIM. If the access card were cloned or stolen and placed in another device, then the attacker attempting to use it would also have to know the victim's PIN in order to utilize the stolen SIM and/or vPOS **120**. Without the PIN, the overall authentication of a transaction would fail.

[0077] FIG. **2** illustrates the components of the computing system **105**, in accordance with various example embodiments. As shown, computing system **105** may include processor circuitry **210**, interconnect (IX) **220**, network interface circuitry (NIC) **230**, input/output (I/O) interface circuitry **240**, memory circuitry **250**, and TEE **115**. As shown, TEE **115** may include memory **350** and processor circuitry **310**. In various embodiments, one or more of the elements/components not included in the TEE **115** may be referred to as the REE or "host platform". The TEE **115** is an execution environment that runs alongside but is isolated from the REE. The TEE **115** has security capabilities and meet certain security related requirements. The TEE **115** protects TEE **115** assets from general software attacks, defines rigid safeguards as to data and functions that a program can access, and resists a set of defined threats.

[0078] Memory circuitry **250** may be a hardware device configured to store an operating system **260** and program code for one or more software components, such as ePOS client **110** and/or one or more other applications. Memory circuitry **250** may be a computer readable storage medium that generally includes a random access memory (RAM), read only memory (ROM), a flash memory device, a solid state disk (SSD), a secure digital (SD) card, and/or other like storage media capable of storing and recording data. The program code and/or software components may also be loaded from a separate computer readable storage medium into memory circuitry **250** using a drive mechanism (not shown). Such separate computer readable storage medium may include a memory card, memory stick, removable flash drive, SIM card, and/or other like computer readable storage medium (not shown). In some embodiments, software components may be loaded into memory circuitry **250** via NIC **230**, rather than via a computer readable storage medium.

[0079] Any number of memory devices may be used to provide for a given amount of system memory **250**. As examples, the memory may be random access memory (RAM) in accordance with a Joint Electron Devices Engineering Council (JEDEC) design such as the DDR or mobile DDR standards (e.g., LPDDR, LPDDR2, LPDDR3, or LPDDR4). In particular examples, a memory component may comply with a DRAM standard promulgated by JEDEC, such as JESD79F for DDR SDRAM, JESD79-2F for DDR2 SDRAM, JESD79-3F for DDR3 SDRAM, JESD79-4A for DDR4 SDRAM, JESD209 for Low Power DDR (LPDDR), JESD209-2 for LPDDR2, JESD209-3 for LPDDR3, and JESD209-4 for LPDDR4. Other types of RAM, such as dynamic RAM (DRAM), synchronous DRAM (SDRAM), and/or the like may also be included. Such standards (and similar standards) may be referred to as DDR-based standards and communication interfaces of the storage devices that implement such standards may be referred to as DDR-based interfaces. In various implementations, the individual memory devices may be of any number of different package types such as single die package (SDP), dual die package (DDP) or quad die package (Q17P). These devices, in some examples, may be directly soldered onto a motherboard to provide a lower profile solution, while in other examples the devices are configured as one or more memory modules that in turn couple to the motherboard by a given connector. Any number of other memory implementations may be used, such as other types of memory modules, e.g., dual inline memory modules (DIMMs) of different varieties including but not limited to microDIMMs or MiniDIMMs.

[0080] To provide for persistent storage of information such as data, applications, operating systems and so forth, storage device(s) may also be included in or coupled with the system **105**. In an example, the storage device(s) may be implemented via a solid-state disk drive (SSDD) and/or high-speed electrically erasable memory (commonly referred to as "flash memory"). Other devices that may be used for the storage device(s) include flash memory cards, such as SD cards, microSD cards, XD picture cards, and the like, and USB flash drives. In an example, the memory device may be or may include memory devices that use chalcogenide glass, multi-threshold level NAND flash memory, NOR flash memory, single or multi-level Phase Change Memory (PCM), a resistive memory, nanowire memory, ferroelectric transistor random access memory (FeTRAM), anti-ferroelectric memory, magnetoresistive random access memory (MRAM) memory that incorporates memristor technology, phase change RAM (PRAM), resistive memory including the metal oxide base, the oxygen vacancy base and the conductive bridge Random Access Memory (CB-RAM), or spin transfer torque (STT)-MRAM, a spintronic magnetic junction memory based device, a magnetic tunneling junction (MTJ) based device, a Domain Wall (DW) and Spin Orbit Transfer (SOT) based device, a thyristor based memory device, or a combination of any of the above, or other memory. The memory circuitry **250** and/or storage circuitry may also incorporate three-dimensional (3D) cross-point (XPOINT) memories from Intel® and Micron®. In low power implementations, the storage device(s) may be on-die memory or registers associated with

the processor circuitry **210**. However, in some examples, the storage device(s) may be implemented using a micro hard disk drive (HDD). Further, any number of new technologies may be used for the storage device(s) in addition to, or instead of, the technologies described, such resistance change memories, phase change memories, holographic memories, or chemical memories, among others

[0081] During operation, memory circuitry **250** may include operating system **260** and ePOS client **110**. Operating system **260** may manage computer hardware and software resources and provide common services for computer programs. Operating system **260** may include one or more drivers, such as a display driver, sensor drivers (e.g., a camera driver, etc.), audio drivers, and/or any other like drivers that provide an interface to hardware devices without needing to know the details of the hardware itself. According to various embodiments, the operating system **260** may include one or more drivers for accessing the TEE **115** thereby enabling ePOS client **110** to access hardware functions inside the TEE **115**, such as the vPOS **120**, without needing to know the details of the TEE **115** itself. The operating system **260** may be a general purpose operating system or an operating system specifically written for and tailored to the computing system **105**.

[0082] ePOS client **110** may be a collection of software modules and/or program code that enables the computing system **105** to access or obtain POS transaction data and/or other like data from the vPOS **120** within the TEE **115**, and/or operate according to the various example embodiments as disclosed herein. ePOS client **110** may be a native application, a web application, or a hybrid application. In some embodiments, ePOS client **110** may be a web application that may be rendered in a web browser of the computing system **105**.

[0083] Processor circuitry **210** is configurable or operable to carry out instructions of a computer program by performing the basic arithmetical, logical, and input/output operations of the system. The processor circuitry **210** includes circuitry such as, but not limited to one or more processor cores and one or more of cache memory, low drop-out voltage regulators (LDOs), interrupt controllers, serial interfaces such as SPI, I^2C or universal programmable serial interface circuit, real time clock (RTC), timer-counters including interval and watchdog timers, general purpose I/O, memory card controllers such as secure digital/multi-media card (SD/MMC) or similar, interfaces, mobile industry processor interface (MIPI) interfaces and Joint Test Access Group (JTAG) test access ports. In some implementations, the processor circuitry **210** may include one or more hardware accelerators, which may be microprocessors, programmable processing devices (e.g., FPGA, ASIC, etc.), or the like. The one or more accelerators may include, for example, computer vision and/or deep learning accelerators. In some implementations, the processor circuitry **210** may include on-chip memory circuitry, which may include any suitable volatile and/or non-volatile memory, such as DRAM, SRAM, EPROM, EEPROM, Flash memory, solid-state memory, and/or any other type of memory device technology, such as those discussed herein.

[0084] The processor circuitry **210** may include, for example, one or more processor cores (CPUs), application processors, GPUs, RISC processors, Acorn RISC Machine (ARM) processors, CISC processors, one or more DSPs, one or more FPGAs, one or more PLDs, one or more ASICs, one or more baseband processors, one or more radio-frequency integrated circuits (RFIC), one or more microprocessors or controllers, a multi-core processor, a multithreaded processor, an ultra-low voltage processor, an embedded processor, or any other known processing elements, or any suitable combination thereof. The processors (or cores) **210** may be coupled with or may include memory/storage and may be configured to execute instructions stored in the memory/storage to enable various applications or operating systems to run on the system **105**. The processors (or cores) **210** is configured to operate application software to provide a specific service to a user of the system **105**. In some embodiments, the processor(s) **210** may be a special-purpose processor(s)/controller(s) configured (or configurable) to operate according to the various embodiments herein.

[0085] As examples, the processor(s) **210** may include an Intel® Architecture Core™ based processor such as an i3, an i5, an i7, an i9 based processor; an Intel® microcontroller-based processor such as a Quark™, an Atom™, or other MCU-based processor; Pentium® processor(s), Xeon® processor(s), or another such processor available from Intel® Corporation, Santa Clara, Calif. However, any number other processors may be used, such as one or more of Advanced Micro Devices (AMD) Zen® Architecture such as Ryzen® or EPYC® processor(s), Accelerated Processing Units (APUs), MxGPUs, Epyc® processor(s), or the like; A5-A12 and/or S1-S4 processor(s) from Apple® Inc., Snapdragon™ or Centrig™ processor(s) from Qualcomm® Technologies, Inc., Texas Instruments, Inc.® Open Multimedia Applications Platform (OMAP)™ processor(s); a MIPS-based design from MIPS Technologies, Inc. such as MIPS Warrior M-class, Warrior I-class, and Warrior P-class processors; an ARM-based design licensed from ARM Holdings, Ltd., such as the ARM Cortex-A, Cortex-R, and Cortex-M family of processors; the ThunderX2® provided by Cavium™, Inc.; or the like. In some implementations, the processor(s) **210** may be a part of a system on a chip (SoC), System-in-Package (SiP), a multi-chip package (MCP), and/or the like, in which the processor(s) **210** and other components are formed into a single integrated circuit, or a single package, such as the Edison™ or Galileo™ SoC boards from Intel® Corporation. Other examples of the processor(s) **210** are mentioned elsewhere in the present disclosure.

[0086] The processor circuitry **210** may perform a variety of functions for the computing system **105** and may process data by executing program code, one or more software modules, firmware, middleware, microcode, hardware description languages, and/or any other like set of instructions stored in the memory circuitry **250**. The program code may be provided to processor circuitry **210** by memory circuitry **250** via IX **220**, one or more drive mechanisms (not shown), and/or via NIC **230**. In order to perform the variety of functions and data processing operations, the program code and/or software components may be executed by the processor circuitry **210**. On execution by the processor circuitry **210**, the processor circuitry **210** may cause computing system **105** to perform the various operations and functions delineated by the program code.

[0087] For example, in various embodiments, the ePOS client **110** may include various modules and/or program code configured to operate (using hardware and/or software components) to obtain POS transaction data from the vPOS **120** and communicate various POS transaction-related data with the vPOS **120** and/or the cloud POS service **135** as

described herein. Once the various modules and/or program code of the ePOS client **110** are loaded into memory circuitry **250** and executed by the processor circuitry **210**, the processor circuitry **210** is configurable or operable to cause computing system **105** to receive a transaction initiation from the cloud POS service **135** or other merchant-related terminal; provide a selected payment credential to be used to process a POS transaction; provide authentication parameters to the cloud POS service **135** or other merchant-related terminal; receive a cryptographic merchant certificate from the cloud POS service **135** or other merchant-related terminal, and provide the cryptographic merchant certificate to the vPOS **120**; receive a personal identification (PIN) solicitation from the cloud POS service **135** or other merchant-related terminal; provide a user interface to allow a user of the computing system **105** to enter a PIN; receive the input PIN and provide the inputted PIN to the cloud POS service **135** or other merchant-related terminal; receive updated transaction terms from the cloud POS service **135** or other merchant-related terminal; receive encrypted payment data from the vPOS **120**; communicate the encrypted payment data to the cloud POS service **135** or other merchant-related terminal; and/or perform various other functions or combinations of functions as disclosed herein. While specific modules are described herein, it should be recognized that, in various embodiments, various modules may be combined, separated into separate modules, and/or omitted. Additionally, in various embodiments, one or more modules may be implemented on separate devices, in separate locations, or distributed, individually or in sets, across multiple processors, devices, locations, and/or in cloud-computing implementations.

[0088] IX **220** is configurable or operable to enable the communication and data transfer between the processor circuitry **210** and memory circuitry **250**. The IX **220** may include any number of technologies, including Industry Standard Architecture (ISA), extended ISA, inter-integrated circuit (I^2C) IX, Serial Peripheral Interface (SPI), point-to-point interfaces, power management bus (PMBus), Peripheral Component Interconnect (PCI), PCI express (PCIe), PCI extended (PCIx), a Small Computer System Interface (SCSI) IX, Intel® Ultra Path Interconnect (UPI), Intel® Accelerator Link, Intel® Common Express Link (CXL), Coherent Accelerator Processor Interface (CAPI), OpenCAPI, Intel® QuickPath Interconnect (QPI), Intel® Omni-Path Architecture (OPA) IX, RapidIO™ system IXs, Cache Coherent Interconnect for Accelerators (CCIX), Gen-Z Consortium IXs, a HyperTransport interconnect, NVLink provided by NVIDIA®, a Time-Trigger Protocol (TTP) system, a FlexRay system, and/or any number of other IX technologies. The IX **220** may be a proprietary bus, for example, used in a SoC based system. In some implementations, the IX **220** may comprise a high-speed serial bus, parallel bus, internal universal serial bus (USB), Front-Side-Bus (FSB), and/or other suitable communication technology for transferring data between components within computing system **105** and other like devices.

[0089] NIC **230** may be a computer hardware component that connects computing system **105** to a computer network (e.g., network **150**). NIC **230** may connect computing system **105** to a computer network via a wired or wireless connection. NIC **230** may operate in conjunction with a wireless transmitter/receiver and/or transceiver (not shown) that is configured to operate in accordance with one or more wireless standards. The wireless transmitter/receiver and/or transceiver is configurable or operable to operate in accordance with a wireless communications standard, such as the IEEE 802.11-2007 standard (802.11), the Bluetooth standard, and/or any other like wireless standards. The communications port is configurable or operable to operate in accordance with a wired communications protocol, such as a serial communications protocol (e.g., the Universal Serial Bus (USB), FireWire, Serial Digital Interface (SDI), and/or other like serial communications protocols), a parallel communications protocol (e.g., IEEE 1284, Computer Automated Measurement And Control (CAMAC), and/or other like parallel communications protocols), and/or a network communications protocol (e.g., Ethernet, token ring, Fiber Distributed Data Interface (FDDI), and/or other like network communications protocols). The NIC **230** may also include one or more virtual network interfaces configured to operate with ePOS client **110** and/or other like applications.

[0090] I/O interface circuitry **240** may be a computer hardware component that provides communication between the computing system **105** and one or more other devices. The I/O interface circuitry **240** may include one or more user interfaces designed to enable user interaction with the computing system **105** and/or peripheral component interfaces designed to provide interaction between the computing system **105** and one or more peripheral components. User interfaces may include, but are not limited to a physical keyboard or keypad, a touchpad, a speaker, a microphone, etc. Peripheral component interfaces may include, but are not limited to, a non-volatile memory port, a USB port, an audio jack, and a power supply interface.

[0091] As discussed above, computing system **105** may also include a transmitter and receiver or a transceiver (not shown). The transmitter may be any type of hardware device that generates or otherwise produces radio waves in order to communicate with one or more other devices. The transmitter may be coupled with an antenna (not shown) in order to transmit data to one or more other devices. The transmitter is configurable or operable to receive digital data from one or more components of computing system **105** via IX **220**, and convert the received digital data into an analog signal for transmission over an air interface. The receiver may be any type of hardware device that can receive and convert a signal from a modulated radio wave into usable information, such as digital data. The receiver may be coupled with the antenna (not shown) in order to capture radio waves. The receiver is configurable or operable to send digital data converted from a captured radio wave to one or more other components of computing system **105** via IX **220**. In embodiments where a transceiver (not shown) is included with computing system **105**, the transceiver may be a single component configured to provide the functionality of a transmitter and a receiver as discussed above.

[0092] TEE **115** may be one or more hardware devices and software modules configured to carry out secure operations and/or control the storage and use of personal and/or confidential data. In various embodiments, the TEE **115** may be a single integrated circuit (IC) containing a processing device, memory, an internal bus/IX, and/or an I/O interface. As shown by FIG. **2**, the TEE **115** includes memory **350**, processor circuitry **310**, IX **320** and **325**, and secure I/O interface **340**.

[0093] Memory **350** may be a same type of device or similar hardware device as memory circuitry **250**, such as

RAM, ROM, a flash memory device, a SSD, and/or other like storage media capable of storing and recording data. However, according to various embodiments, the memory **350** may only be accessed (i.e., read and/or written to) by processor circuitry **310**. In such embodiments, components that are external to trusted execution environment may access the memory **350** via secure I/O interface **340**. Memory **350** is configurable or operable to program code for one or more software components, such as vPOS **120**, PCDB **125**, and/or one or more other like applications. The program code and/or software components may be loaded into memory **350** from a separate computer readable storage medium, from memory circuitry **250**, using a drive mechanism (not shown), NIC **230**, and/or I/O interface circuitry **240** via the secure I/O interface **340**.

[0094] The vPOS **120** may be a collection of software elements, engines, modules, applet(s), and/or program code that enables the computing system **105** to process POS transactions by validating and/or encrypting/decrypting POS transaction data, and/or otherwise operate according to the various example embodiments as disclosed herein. The vPOS **120** may be a native application, an applet (e.g., Java® applet), or a firmware application. PCDB **125** may be one or more databases that stores payment credentials in association with other payment credential-related information. As discussed elsewhere, the PCDB **125** may be one or more relational databases, one or more object databases, one or more column-oriented databases, one or more correlation databases, and/or the like.

[0095] IX **320** may be a same type of device or similar to IX **220** and/or IX **225** and/or other suitable IX/bus technology for transferring data between components within TEE **115** and with components outside of TEE **115**. In various embodiments, the processor circuitry **310** may be the same or similar as processor circuitry **210**.

[0096] Processor circuitry **310** is configurable or operable to carry out instructions of a computer program (e.g., vPOS **120**) by performing the basic arithmetical, logical, and input/output operations. The processor circuitry **310** may include a single-core processor, a dual-core processor, a triple-core processor, a quad-core processor, one or more digital signal processor (DSP), an application-specific integrated circuit (ASIC) device, and/or the like. The processor circuitry **310** may perform a variety of functions for the computing system **105** and may process data by executing program code, one or more software modules, firmware, middleware, microcode, hardware description languages, and/or any other like set of instructions stored in the memory **350**. The program code may be provided to processor circuitry **310** by memory **350** via bus **320**, and/or by processor circuitry **210** via secure I/O interface **340**. In order to perform the variety of functions and data processing operations, the program code and/or software components may be executed by the processor circuitry **310**. On execution by the processor circuitry **310**, the processor circuitry **310** may cause computing system **105** to perform the various operations and functions delineated by the program code.

[0097] For example, in various embodiments, the vPOS **120** may include various modules and/or program code configured to operate (using hardware and/or software components) to process POS transactions and communicate various POS transaction-related data with the cloud POS service **135** via the ePOS client **110** as described herein. Once the various modules and/or program code of the vPOS **120** are loaded into memory **350** and executed by the processor circuitry **310**, the processor circuitry **310** is configurable or operable to cause computing system **105** to obtain payment credentials and/or other like information from the PCDB **125**; validate merchant terminals associated with a POS transaction by decrypting, deciphering, or otherwise decode merchant certificates; process POS transactions using the selected payment credentials; generate and encrypt payment data; and/or perform various other functions or combinations of functions as disclosed herein. While specific modules are described herein, it should be recognized that, in various embodiments, various modules may be combined, separated into separate modules, and/or omitted. Additionally, in various embodiments, one or more modules may be implemented on separate devices, in separate locations, or distributed, individually or in sets, across multiple processors, devices, locations, and/or in cloud-computing implementations.

[0098] In various embodiments, the TEE **115** may be a graphics and memory controller hub and the processor circuitry **310** and memory **350** may be part of microcontroller IC that includes the Management Engine firmware provided by Intel®. In some embodiments, the TEE **115** may be a Trusted Platform Module (TPM) provided by Intel® that operates in accordance with ISO/IEC 11889:2009, wherein the processor circuitry **310** is a secure cryptoprocessor. However, it should be noted that the TEE **115** may be any secure region of a computing device that may protect data that is stored and/or processed within the TEE **115**.

[0099] In some embodiments, computing system **105** may include many more components than those shown in FIG. **2**, such as a display device (e.g., a computer monitor, a touchscreen, etc.), one or more input devices (e.g., a physical keyboard, a touch screen, etc.), one or more image sensors, a transmitter/receiver (or alternatively, a transceiver), a mobile video card and/or graphics processing unit (GPU), and other like components.

[0100] FIG. **3** illustrates the components of the computing system **105**, in accordance with other various example embodiments. As shown by FIG. **3**, computing system **105** may include similar components as shown by FIG. **2**, such as processor circuitry **210**, bus **220**, NIC **230**, input/output (I/O) interface circuitry **240**, and memory circuitry **250**, each of which may operate in a same or similar manner as discussed with regard to FIG. **2**. However, according to the example embodiment shown by FIG. **3**, the TEE **115** may be within memory circuitry **250**. In some embodiments, computing system **105** may include many more components than those shown in FIG. **3**, such as a display device (e.g., a computer monitor, a touchscreen, etc.), one or more input devices (e.g., a physical keyboard, a touch screen, etc.), one or more image sensors, a transmitter/receiver (or alternatively, a transceiver), a mobile video card and/or graphics processing unit (GPU), and other like components.

[0101] According to the example embodiment shown by FIG. **3**, the TEE **115** may be a hardware enforceable container called an enclave, a secure enclave, and the like. The enclaves may be used to store security critical code and/or data, such as payment credentials and/or other POS transaction data. The enclaves may be one or more isolated regions of memory circuitry **250** that are encrypted within the memory circuitry **250**. In some embodiments, the enclaves may be managed by a virtual machine monitor (WM) of a virtual machine running as a guest operating

system. The memory location of the enclaves may be protected from access by all hardware components and/or software components of computing system **105**, regardless of a privilege level assigned to the hardware components and/or software components. In such embodiments, only the processor circuitry **210** executing a secure application (e.g., ePOS client **110**) may access an enclave. In such embodiments, the operating system **260** may include a secure driver that provides an interface for hardware devices and/or software components to create secure applications; access enclaves created by the secure applications; create secure application certificates (e.g., an "application cryptogram" or "transaction certificate") that allow one or more software components; hardware devices; or external computing devices to attest to validate their identity; create secure channels between multiple enclaves; and/or other like secure operations. In embodiments where the processor circuitry **210** is a multi-core processor, the processor circuitry **210** may execute program code from an enclave in parallel with unsecured program code. In various embodiments, the TEE **115** may be one or more secure enclaves defined using the Intel® Software Guard Extensions (Intel® SGX).

[0102] FIG. **4** shows an arrangement **400** in which a POS transaction may be processed using the vPOS of the present disclosure, in accordance with various other example embodiments. As shown in FIG. **4**, arrangement **400** may include computing system **105**A, merchant domain **130**, payment acquiring service **145**, messaging service **160**, network **150**, and cloud trusted execution environment **415**. Additionally, computing system **105**A may include ePOS client **110**, and the cloud trusted execution environment **415** may include the vPOS **120** and PCDB **125**. Furthermore, the merchant domain may include a cloud POS service **135** and the merchant server **140** (also referred to as "merchant **140**").

[0103] The cloud POS service **135**, the merchant server **140**, payment acquiring service **145**, messaging service **160**, and the network **150** as shown in FIG. **4** may be the same or similar to the cloud POS service **135**, the merchant server **140**, payment acquiring service **145**, messaging service **160**, and the network **150** as shown in FIG. **1**. Additionally, the ePOS client **110**, the vPOS **120**, and the PCDB **125** of arrangement **400** may operate in a same or similar fashion as the ePOS client **110**, the vPOS **120**, and the PCDB **125** of arrangement **100**. In contrast to computing system **105** of FIG. **1**, as shown in FIG. **4**, the computing system **105**A does not include a TEE **115**, but instead, the TEE **115** is replaced with the cloud TEE **415**.

[0104] The cloud TEE **415** may be one or more hardware computing devices that may include one or more servers or other like network elements for providing one or more services. The one or more services provided by the cloud trusted execution environment **415** may include processing POS transactions on behalf of the computing system **105**. In this regard, cloud trusted execution environment **415** may communicate data (i.e., transmit and receive) with ePOS client **110** located in the computing device **105**A via network **150**. In order to communicate with the ePOS client **110**, the cloud trusted execution environment **415** may generate POS transaction messages (which may be communicated to the ePOS client **110**) in accordance with ISO 8583 and/or any other like standards for encrypting and/or encapsulating data for exchanging electronic transactions made by cardholders using payment cards. In such embodiments, the ePOS client **110** of the computing system **105**A may also communicate with the cloud trusted execution environment **415** via network **150**. The communications between the computing system **105**A and cloud trusted execution environment **415** may be carried out in a same or similar fashion as the communications between the ePOS client **110** and the vPOS **120** as discussed previously with regard to FIGS. **1-3**. However, in the arrangement **400** the communications between the ePOS client **110** and the vPOS **120** may additionally include encrypting and/or enciphering the POS transaction messages according to one or more encryption or enciphering algorithms discussed herein prior to communicating such messages over the network connections **150**.

[0105] In cloud-based embodiments, the ePOS applet(s) **121** may be web-based widgets or web-based applets that run in a browser, or within a native or hybrid application, operated by the computing system **105**. In these embodiments, the POS cloud **135** or some other cloud service also hosts or otherwise stores user credentials on behalf of the user in a same or similar manner as discussed with respect to the PCDB **125**. For cloud-hosted credentials, the systems that are granted credential access establish and maintain a strong pairing relationship with the cloud POS service **135** (e.g., a secure session or the like). In one example implementation, a time-based OTP (TOTP) cryptographic method may be used for this purpose.

[0106] FIG. **5** illustrates the components of example computing system **105**A and cloud trusted execution environment **415** with the vPOS **120**, in accordance with various example embodiments. As shown, computing system **105**A may include processor circuitry **210**, bus **220**, NIC **230**, input/output (I/O) interface circuitry **240**, and memory circuitry **250**. As shown, cloud trusted execution environment **415** may include a processor **510**, a network interface **530**, and the TEE **115**. The trusted execution environment may include memory **350** and processor circuitry **310**. In some embodiments, computing system **105**A and/or cloud trusted execution environment **415** may include many more components than those shown in FIG. **5**, such as a display device (e.g., a computer monitor, a touchscreen, etc.), one or more input devices (e.g., a physical keyboard, a touch screen, etc.), one or more image sensors, a transmitter/receiver (or alternatively, a transceiver), a mobile video card and/or graphics processing unit (GPU), and other like components.

[0107] The processor circuitry **210**, the bus **220**, the NIC **230**, the I/O interface circuitry **240**, and the memory circuitry **250** of the computing system **105**A may be the same or similar as the processor circuitry **210**, the bus **220**, the NIC **230**, the I/O interface circuitry **240**, and the memory circuitry **250** of the computing device **10** discussed previously with regard to FIGS. **2-3**. Additionally, the TEE **115**, the memory **350**, the processor circuitry **310**, the bus **320** and the secure I/O interface **340** of the cloud trusted execution environment **415** may be the same or similar to the TEE **115**, the memory **350**, the processor circuitry **310**, the bus **320** and the secure I/O interface **340** of computing system **105** as discussed previously with regard to FIG. **2**. Furthermore, the processor **510** may be a same or similar device that operates in a same or similar fashion as the processor circuitry **210**, and the network interface **530** may be a same or similar device that operates in a same or similar fashion as the NIC **230**.

[0108] In contrast to computing system **105** discussed with regard to FIGS. **1-3**, as shown in FIG. **5**, the ePOS

client **110** of computing system **105**A may access the vPOS **120** via the network **150** using the NIC **230** to communicate POS transaction messages to the cloud trusted execution environment **415**. The network interface **530** may receive the communicated POS transaction messages and provide the POS transaction messages in their encrypted form to the processor **510**, which may then route the POS transaction messages to the secure I/O interface **340** in a same or similar fashion as discussed previously with regard to FIG. **2**. The cloud trusted execution environment **415** may also communicate POS transaction messages to the computing system **105**A via the network **150** by using the network interface **530**.

[0109] Furthermore, although not shown, in various embodiments, the cloud trusted execution environment **415** may have a component configuration that is similar to the configuration shown by FIG. **3**. In such embodiments, computing system **105**A may still include similar components as shown by FIGS. **2** and **5**, such as processor circuitry **210**, bus **220**, NIC **230**, input/output (I/O) interface circuitry **240**, and memory circuitry **250**, each of which may operate in a same or similar manner as discussed with regard to FIG. **2**. Additionally, in such embodiments, the TEE **115** may be within a memory device of the cloud trusted execution environment **415** (not shown) and may operate in a same or similar fashion as discussed with regard to FIG. **3**, such as the TEE **115** of the cloud trusted execution environment **415** including one or more secure enclaves defined using the Intel® SGX.

[0110] FIG. **6** illustrates a method **400** for processing a POS transaction, in accordance with various example embodiments. FIGS. **7-14** illustrate various user interface stages, each of which corresponds to an operation of method **600** of FIG. **6** having a same numeral. In particular, each of FIGS. **7-14** illustrate an example user interface that may be displayed on a display device of computing system **105** as each operation of method **600** is performed using computing system **105**. However, it should be noted that the process **600** may be performed by computing system **105**A, wherein the example user interfaces may be displayed on a display device of computing system **105**A as each operation of method **600** is performed using computing system **105**A. While particular examples and orders of operations are illustrated in FIGS. **6-14**, in various embodiments, these operations may be re-ordered, broken into additional operations, combined, and/or omitted altogether. Furthermore, the user interfaces illustrated by FIGS. **7-14** may be generated or otherwise formed in various artistic representations without departing from the example embodiments disclosed herein.

[0111] Referring to FIG. **6**, at operation **700** the computing system **105** may provide a selection of payment options for completing a POS transaction. Referring to FIG. **7**, the computing system **105** may display a screen **505** that includes three payment option buttons **710**A-C. In various embodiments, the screen **505** may be displayed during an ecommerce checkout process for purchasing products and/or services from an online store or a merchant website (e.g., a website associated with merchant **140**). As shown by FIG. **7**, the computing system **105** may be a mobile device including a touchscreen display device, wherein a user may provide an input to the computing system **105** by performing one or more gestures on the touchscreen display device using a finger, a stylus, etc. The payment option buttons **710**A-C may be graphical control elements that may be used to provide a selection of one of a plurality of payment options. In such embodiments, the user may press one of the payment option buttons **710**A-C to select a desired payment option. When a user of computing system **105** selects a desired payment option (e.g., payment option 1 **710**A), the computing system **105** may activate the ePOS client **110** so that the user may access the vPOS **120** to complete the POS transaction.

[0112] Referring back to FIG. **6**, at operation **800** the computing system **105** may obtain a unique identifier associated with the computing system **105** and provide the unique identifier to initiate the POS transaction. Referring to FIG. **8**, when the ePOS client **110** is activated, the ePOS client **110** may display a screen **605** that allows a user to enter or input a unique identifier associated with the computing system **105**. In embodiments where the computing system **105** includes a touchscreen display device, the screen **805** may include a keypad that may be overlaid on top of the ePOS client **110** (not shown) so as to enable the user of the computing device to enter the unique identifier into the text box **810**. Text box **810** may be any type of graphical control element that enables a user of the computing system **105** to input text information to be used for initiating a POS transaction. The send button **815** may be a graphical control element that may operate in a same or similar fashion as the payment option buttons **710**A-C, which may be used to submit the unique identifier to the cloud POS service **135**.

[0113] In the embodiment shown by FIG. **8**, the unique identifier may be a phone number associated with the computing system **105**. As shown, a user of the computing system **105** may input the phone number into text box **810**, and when the user selects the send button **815**, the ePOS client **110** may provide the phone number to the cloud POS service **135** to initiate the POS transaction. It should be noted that in some embodiments, the unique identifier may be an email address, a media access control (MAC) address, a domain name, an IP address, and/or any other like unique identifier.

[0114] Referring back to FIG. **6**, at operation **900**, the ePOS client **110** of the computing system **105** may receive a transaction initiation from the cloud POS service **135**. Referring to FIG. **9**, when the computing system **105** receives the transaction initiation, the ePOS client **110** may display screen **905**, which includes message **910** that indicates that the computing system **105** has received a POS transaction initiation. In embodiments where the unique identifier is a phone number associated with the computing system **105**, the transaction initiation may be received via a text message. In such embodiments, the text message may be a short message service (SMS) message, a multimedia messaging service (MMS) message, or any other type of message that based in cellular network technology. It should be noted that in embodiments where the unique identifier is a MAC address or an IP address, the message **910** may be a push notification sent by a messaging service (e.g., messaging service **160**), a notification in response to a client pull communication, and/or the like. In various embodiments, the message **910** may be a graphical control element that allows a user of the computing system **105** to initiate processing the POS transaction indicated by the message **910**. For example, when the user selects the message **910**, the ePOS client **110** may provide a user interface that allows

the user to select one or more payment credentials to be used for processing the POS transaction.

[0115] Referring back to FIG. **6**, at operation **1000**, the ePOS client **110** of the computing system **105** may provide a selection of a payment credential to be used for processing the POS transaction. Referring to FIG. **10**, the ePOS client **110** may display screen **1005**, which includes message **1010**, payment credential buttons **1015**A-C, cancel button **1020**, and authorize button **1025**.

[0116] In the embodiment shown by FIG. **10**, the message **1010** may indicate a POS transaction amount (e.g., $54.50) and a merchant payee (e.g., merchant **140**). The message **1010** may also include a trademark, symbol, emblem, or other like identifying image associated with the merchant payee. In some embodiments the message **1010** may include the same or similar information as message **910**.

[0117] The payment credential buttons **1015**A-C may be graphical control elements that may operate in a same or similar fashion as buttons **710**A-C and **815**. The payment credential buttons **1015**A-C may allow a user of the computing system **105** to select a desired payment credential stored in the PCDB **125**. Each of the payment credentials represented by payment credential buttons **1015**A-C may include payment information that may be accepted by a merchant payee. In most embodiments, each of the payment credentials may be associated with a physical credit card, a charge card, a debit card, a prepaid card, a gift card, an automated teller machine (ATM) card, an EBT card, a stored-value card, a fleet card, an electronic check, digital currency wallet, and/or other like physical payment credentials. In such embodiments, each payment credential may include payment information, such as card number, expiration date, card verification code (CVC) code, digital currency public-private key pairs, cardholder name, billing address, security questions and answers, and the like. In some embodiments, each payment credential may be associated with an electronic form of payment, without a physical counterpart, such as an electronic script or any other type of substitute for legal tender. In such embodiments, the payment credentials may include the same or similar information as discussed previously with regard to physical payment cards. Additionally, each payment credential may also include additional information that is required or desired to complete a POS transaction, such as authentication terms and/or transaction terms required by a payment credential, a merchant authentication challenge, a cryptographic client certificate that is unique to a payment credential, and/or other like POS transaction-related data.

[0118] Furthermore, FIG. **10** shows three payment credential buttons **1015**A-C, each of which is associated with a different payment credential. However, according to various embodiments, the number of payment credential buttons that are displayed in screen **805** may vary according to the number and types of payment credentials accepted by the merchant **140** and/or the number and types of payment credentials that are stored in the PCDB **125**. Therefore, many more (or fewer) payment credentials may be displayed in screen **805**. For example, the merchant **140** may only accept payment credentials associated with payment credential buttons **1015**A-C even though the PCDB **125** stores many more payment credentials. By way of another example, the PCDB **125** may only include the payment credentials associated with payment credential buttons **1015**A-C, even though the merchant **140** may accept many more payment credentials. Additionally, in some embodiments, the POS transaction initiation may indicate a geographic location of the merchant **140**, and the ePOS client **110** may obtain payment credentials from within the PCDB **125** that may be accepted within a jurisdiction or other like venue containing the geographic location of the merchant **140**. By way of example, the PCDB **125** may include payment credentials that may be accepted in the United States and payment credentials that may be accepted in France, and if the merchant **140** is located within the United States, the payment credential buttons **1015**A-C may be associated with payment credentials that are issued by issuing banks located within the United States or otherwise accepted by merchants located within the United States, even though the computing system **105** may be located within France. In various embodiments, the POS transaction initiation may indicate the types of payment credentials accepted by the merchant **140**, and the ePOS client **110** may generate screen **1005** to include a number of payment credential buttons **1015** based on a query of the PCDB **125** using the information contained in the POS transaction initiation. In some embodiments, the ePOS client **110** may submit the query to the vPOS **120**, which may conduct the actual search through the PCDB **125**, while in other embodiments the ePOS client **110** may search through the payment credential DB module **110** independent of the vPOS **120**. For example, the ePOS client **110** may submit the query through a secure I/O interface **340** (see e.g., the example embodiments described with regard to FIGS. **2** and **5**) or may issue one or more SGX commands to query the PCDB **125** (see e.g., the example embodiments described with regard to FIGS. **3** and **5**).

[0119] Once the user of the computing system **105** selects one of the payment credentials to be used for processing the POS transaction, such by pressing the payment credential button **1015**A to select payment credential **1**, the user may then select the authorize button **1025** to initiate the vPOS **120** to process the POS transaction using the selected payment credential **1**, or the user of the computing system **105** may select the cancel button **1020** to cancel the POS transaction.

[0120] Referring back to FIG. **6**, at operation **1100**, the ePOS client **110** of the computing system **105** may provide a passcode to authorize use of the selected payment credential. Referring to FIG. **11**, the screen **1105** may include message **1110**, keypad **1115**, cancel button **1120**, and authorize button **1125**, which may operate in a same or similar manner as message **1010**, keypad **1015**, cancel button **1020**, and submit button **1025** of FIG. **10**, respectively. In various embodiments, each of the payment credentials **1015**A-C may require a passcode, password, or other like string of numerals or string of characters to authenticate or otherwise prove an identity of the user of the computing system **105**. As will become apparent later, the passcode used at operation **1100** may be different than a personal identification number (PIN) used to authorize use of a payment credential. In some embodiments, the passcode may be specific to each payment credential, while in other embodiments, the passcode may be a same passcode used to access the computing system **105**, such as from a lockscreen or login screen of the computing system **105**. Accordingly, in various embodiments, the PCDB **125** may store a passcode in association with each payment credential.

[0121] The keypad **1115** may include a set of buttons or other like graphical control elements arranged in a block or pad that bear a digit, symbol, or character that enables the user of the computing system **105** to enter the passcode. Each of the graphical control elements within the keypad **915** may operate in a same or similar manner as discussed with regard to buttons **710**A-C, **815**, **1015**A-C, **1020**, **1025**, etc. As shown by FIG. **11**, the computing system **105** may be a mobile device including a touchscreen display device, wherein the user may press one or more of the graphical control elements of the keypad **1115** using a finger or stylus to enter or input a passcode. In some embodiments where the computing system **105** includes a touchscreen display device, a gesture-based passcode may be used instead of using keypad **1115** to enter a numeric-based and/or character-based passcode. In such embodiments, a user may be required to perform one or more touch-gestures in a predetermined sequence to authenticate or otherwise prove an identity of the user. In embodiments where the computing system **105** does not include a touchscreen display device, the passcode may be entered using a physical keyboard or using mouse point-and-click actions. Once the user of the computing system **105** inputs a passcode using the keypad **1115**, the user may then select the authorize button **1125** to submit the passcode to the vPOS **120** to continue processing the POS transaction using the selected one of the payment credentials **1015**A-C. Additionally, the user of the computing system **105** may select the cancel button **1120** to cancel the POS transaction. In various embodiments, the ePOS client **110** may be operate in conjunction with the Protected Transaction Display (PTD) provided by Intel®, such that the screen **1105** is generated by an application running within the TEE **115** or otherwise outside of the operating system **260**. In such embodiments, the PTD may provide a graphics overlay and randomly placed input options (e.g., buttons of keypad **1115**), which may protect the selected inputs from malware scraping and/or other like malicious attempts to obtain payment credential-related information. Furthermore, it should be noted that in various embodiments, instead of requiring a passcode to be entered by a user of computing system **105**, at operation **1100**, biometric information (e.g., user voice recognition, facial recognition, eye scan, fingerprint, etc.) may be used as an authenticating factor for proving an identity of the user of the computing system **105**. In such embodiments, instead of providing the keypad **1115**, the computing system **105** may execute an application that utilizes one or more biometric sensors, or other like sensors to obtain the biometric information.

[0122] Referring back to FIG. **6**, at operation **1200**, the vPOS **120** of the computing system **105** may begin processing the POS transaction using the selected payment credential when the provided passcode matches a passcode associated with the payment credential. In various embodiments, the ePOS client **110** may provide the input passcode from operation **1100** to the vPOS **120**, wherein the vPOS **120** compares the input passcode with a passcode associated with the selected payment credential. If the input passcode matches the passcode associated with the selected payment credential, the vPOS **120** may provide the ePOS client **110** with various POS transaction-related data associated with the selected payment credential to be communicated with the cloud POS service **135**. In such embodiments, while the computing system **105** is communicating with the cloud POS service **135**, the ePOS client **110** may display screen **1205** to indicate to a user of the computing system **105** that the POS transaction is being processed. Message **1210** may indicate one or more operations that the computing system **105** is undertaking to process the POS transaction. Accordingly, in various embodiments, the message **1210** may display information based on the operations that the computing system **105** is performing. As shown, the screen **1205** may include progress image **1215**, which may be a graphical control element used to visualize a progression of one or more computer operations. In some embodiments, the progress image **1215** may include a progress bar, a textual representation of the progress in a percent format, a throbber, and/or other like an animated graphical control element that visualizes that one or more computer operations are being performed.

[0123] Referring back to FIG. **6**, at operation module **1300**, the ePOS client **110** of the computing system **105** may provide a PIN to authorize use of the selected payment credential. Referring to FIG. **13**, the screen **1305** may include message **1310**, keypad **1315**, cancel button **1320**, and submit button **1325**, which may operate in a same or similar manner as message **1010**, keypad **1015**, cancel button **1020**, and submit button **1025** of FIG. **10**, respectively. In various embodiments, the PIN may be a numeric password that may be used to indicate whether the user of the computing system **105** is authorized to use the selected payment credential. In various embodiments, the PIN that is input at operation module **1300** may be used to authenticate the user with the cloud POS service **135** and/or the vPOS **120**, whereas the passcode discussed with regard to operation **1000** may be used to access the payment credential located within the TEE **115**. It should be noted that, in some embodiments, the requirement to enter a PIN may be dependent on the purchase price, merchant requirements and/or payment credential requirements. For example, some payment credentials or merchants may not require a user to enter a PIN when a purchase price is under a threshold amount or when the purchase price is not above a threshold amount. Once the user of the computing system **105** inputs a PIN using the keypad **1315**, the user may then select the authorize button **1325** to submit the PIN to the cloud POS service **135** to continue processing the POS transaction using the selected payment credential. Additionally, the user of the computing system **105** may select the cancel button **1320** to cancel the POS transaction. Furthermore, in various embodiments, after operation module **1100** the ePOS client **110** may display screen **1205** as shown by FIG. **12** to indicate to the user of the computing system **105** that the POS transaction is currently being processed.

[0124] In various embodiments, the ePOS client **110** may be operate in conjunction with the PTD provided by Intel®, such that the screen **1105** is generated by an application running within the TEE **115** or otherwise outside of the operating system **260**. In such embodiments, the PTD may provide a graphics overlay and randomly placed input options (e.g., buttons of keypad **1115**), which may protect the selected inputs from malware scraping and/or other like malicious attempts to obtain payment credential-related information. Furthermore, it should be noted that in various embodiments, instead of requiring a PIN to be entered by a user of computing system **105**, at operation **1300**, biometric information (e.g., user voice recognition, facial recognition, eye scan, fingerprint, etc.) may be used as an authenticating factor for authorizing use of the selected payment credential.

In such embodiments, instead of providing the keypad **1315**, the computing system **105** may execute an application that utilizes one or more biometric sensors, or other like sensors to obtain the biometric information.

[0125] Referring back to FIG. **6**, at operation **1400**, the ePOS client **110** of the computing system **105** may receive acknowledgment of completion of the POS transaction. Referring to FIG. **14**, the ePOS client **110** may display screen **1410**, which includes message **1410** and image **1415**. The message **1410** and the image **1415** may indicate that the POS transaction was successful (as shown), or may indicate that the POS transaction was unsuccessful (not shown).

[0126] FIG. **15** is a flowchart illustrating an example process **1500** of processing a POS transaction, in accordance with various embodiments. For illustrative purposes, the operations of process **1300** will be described as being performed by the computing system **105** utilizing the various modules, as described with respect to FIGS. **1-14**. However, it should be noted that other similar devices, such as computing system **105**A as discussed with regard to FIGS. **4-5**, may operate the process **1500** as described below. While particular examples and orders of operations are illustrated in FIG. **15**, in various embodiments, these operations may be re-ordered, broken into additional operations, combined, and/or omitted altogether.

[0127] Referring to FIG. **15**, at operation **1505**, the vPOS **120** may receive, via the ePOS client **110**, a transaction initiation from a cloud POS service **135**, messaging service **160**, or other device within the merchant domain **130**. In various embodiments, the transaction initiation may be a SMS message, a MMS message, an email message (or a file attached or otherwise included in an email message), a push notification, an OTT message, and/or any other like message that may be received by the computing system **105** over a wired or wireless network connection.

[0128] At operation **1510**, the vPOS **120** may receive a selection of a payment credential from the ePOS client **110**. The selection of the payment credential may obtained by the ePOS client **110** in a same or similar fashion as discussed with regard to FIGS. **6** and **10**. At operation **1515**, the vPOS **120** may obtain authentication parameters associated with the selected payment credential from within the PCDB **125**. According to various embodiments, the vPOS **120** may obtain the authentication parameters by querying the PCDB **125** according to known database querying methods. At operation **1520**, the vPOS **120** may provide, via the ePOS client **110**, authentication parameters associated with the selected payment credential to the cloud POS service **135** or other device within the merchant domain **130**. In various embodiments, the authentication parameters may be security measures, defined by the selected payment credential, required for authenticating entities and/or communicating data between entities.

[0129] At operation **1525**, the vPOS **120** may receive from the cloud POS service **135**, via the ePOS client **110**, a cryptographic merchant certificate based on the authentication parameters. The cryptographic merchant certificate may be an application cryptogram or a transaction cryptogram (e.g., referred to herein as a "merchant cryptogram"). Additionally or alternatively, the cryptographic merchant certificate may be a public key certificate, digital certificate, identity certificate, etc. that is unique to the merchant **140**, which may be used to prove that the merchant **140** owns or otherwise possesses a public key. The cryptographic merchant certificate may include information indicating the identity of the merchant **140**, such as an IP address and/or domain name of the merchant **140**, a serial number that is unique to the cryptographic merchant certificate, a digital signature used to verify that the cryptographic merchant certificate originated from the merchant **140**, a signature algorithm used to create the digital signature, an issuer identifier such as an IP address and/or domain name, generation date, expiration date, key-usage information, a public key, an algorithm used to hash the public key (also referred to as a "thumbprint algorithm"), a hashed public key (also referred to as a "thumbprint"), and/or any other like information. In various embodiments, the cryptographic merchant certificate may be generated and issued to the merchant **140** by a certificate authority in accordance with EMV standards 4.3 and/or other like standards. However, in various embodiments, the authentication parameters may define one or more criteria required by the selected payment credential to verify the identity of the merchant **140**. Thus, in various embodiments, the cryptographic merchant certificate may be altered to fulfill one or more of the authentication parameters, or may include additional information required by the authentication parameters to validate the identity of the merchant **140**.

[0130] At operation **1530**, the vPOS **120** may validate the merchant **140** identity using the cryptographic merchant certificate. The vPOS **120** may validate the merchant identity according to known methods, such as by using a public key contained in the cryptographic merchant certificate, obtaining a public key associated with the merchant **140** from a certificate authority (either directly or via the cloud POS service **135**), validating the public key signature from the cryptographic merchant certificate with the public key retrieved from the certificate authority, confirming the IP address and/or domain name of the merchant **140** listing in the cryptographic merchant certificate, generating a shared symmetric key to be used to encrypt POS transaction data, encrypting the symmetric key with the public key, and sending the encrypted symmetric key and public key back to ensure that only the cloud POS service **135** can decrypt the encrypted symmetric key and public key using a private key. In some embodiments, the vPOS **120** may validate a digital signature included with the cryptographic merchant certificate according to known methods, which may include obtaining a session key, obtaining a private key associated with the merchant **140**, decrypting the session key using the private key, obtaining an encrypted hash value associated with the merchant **140**, obtaining a public key associated with the merchant **140**, decrypting the encrypted hash value using the public key, comparing the decrypted hash value with the calculated hash value, and validating the identity of the merchant **140** if the decrypted hash value matches the calculated hash value. It should be noted that the merchant **140** identity may be validated using one or more other known methods in addition to, or alternative to the aforementioned methods.

[0131] At operation **1535**, the vPOS **120** may generate transaction data upon properly validating the merchant **140**. In various embodiments, the transaction data may include transaction terms required by the selected payment credential, a cryptographic client certificate, and an authentication challenge. The cryptographic client certificate may be an application cryptogram or a transaction cryptogram (e.g., referred to herein as a "client cryptogram"). Additionally or

alternatively, the transaction terms may include authorization information required by the selected payment credential to authorize payment for the POS transaction. For example, authorization information may indicate whether the user of the computing system **105** is required to enter a PIN, and in some embodiments, the requirement to enter a PIN may be dependent on the purchase price or other like criteria. For example, some payment credentials may not require a user to enter a PIN when a purchase price is under a threshold amount or when the purchase price is not above a threshold amount.

[0132] At operation **1540**, the vPOS **120** may provide, via the ePOS client **110**, the transaction data to the cloud POS service **135**. Prior to providing the transaction data to the ePOS client **110**, the vPOS **120** may encrypt the transaction data using an RSA encryption algorithm, a triple data encryption algorithm (3DES), a secure hash algorithm (SHA), a format preserving encryption algorithm, elliptic curve cryptography, an Advanced Encryption Standard (AES) algorithm, and/or according to any other type of encryption method. In response to properly validating the cryptographic client certificate and decrypting the authentication challenge, the cloud POS service **135** may solicit a PIN from the ePOS client **110**. The cryptographic client certificate may be validated in a same or similar manner as discussed previously with regard to validating the merchant **140** identity using the cryptographic merchant certificate.

[0133] At operation **1545**, the vPOS **120** may receive, via the ePOS client **110**, updated transaction terms from the cloud POS server **135**. In various embodiments, the updated transaction terms may be a stricter version of the transaction terms required by the merchant **140** and/or the payment acquiring service **145** and the transaction terms provided by the vPOS **120** at operation **1540**. In such embodiments, the updated transaction terms may include transaction terms that are common to both the transaction terms required by the selected payment credential and the transaction terms required by the merchant **140** (or required by the payment acquiring service **145**), as well as any transaction terms that are required by the selected payment credential or the merchant **140**/payment acquiring service **145** that are not included in the transaction terms of the other entity. For example, if the selected payment credential requires a user to enter a PIN for POS transactions that are more than $50, and the merchant **140** requires a user to enter a PIN for POS transactions that are more than $25, then the updated transaction terms may require a user to enter a PIN for a POS transaction having a payment amount that is more than $25. By way of another example, if the selected payment credential requires a user to enter a PIN and a cardholder birthdate for POS transactions that are more than $50, and the merchant **140** only requires a user to enter a PIN for POS transactions that are more than $50, then the updated transaction terms may require a user to enter a PIN and a cardholder birthdate for transaction that are more than $50. In various embodiments, the vPOS **120** may accept or deny the updated transaction terms, wherein the vPOS **120** may process the POS transaction according to the transaction terms delineated by the selected payment credential if the vPOS **120** denies the updated transaction terms, or the vPOS **120** may process the POS transaction using the updated transaction terms if the vPOS **120** accepts the updated transaction terms.

[0134] At operation **1550**, the vPOS **120** may generate and encrypt payment data. In various embodiments, the payment data may be generated and encrypted in response to properly validating a PIN and/or other like security information as required by the updated transaction terms. In various embodiments, the payment data may include transaction settlement authorization information (e.g., billing information such as payment amount, currency type, account number, billing/payment address, shipping address, and/or other like cardholder data) a digital signature associated with the payment credential and/or the computing system **105**, and/or other like information. Once generated, the payment data may be encrypted using an RSA encryption algorithm, a triple data encryption algorithm (3DES), a secure hash algorithm (SHA), a format preserving encryption algorithm, elliptic curve cryptography, an Advanced Encryption Standard (AES) algorithm, and/or according to any other type of encryption method. At operation **1555**, the vPOS **120** may provide, via the ePOS client **110**, the encrypted payment data to the cloud POS service **135**. Once the vPOS **120** provides the encrypted payment data, the computing system **105** may proceed back to operation **1505** to receive, via the ePOS client **110**, a transaction initiation from the cloud POS service **135**.

[0135] FIG. **16** is a flowchart illustrating an example process **1400** of processing a POS transaction, in accordance with various embodiments. For illustrative purposes, the operations of process **1600** will be described as being performed by the various elements as described with respect to FIGS. **1-3**. However, it should be noted that other similar devices, such as the computing system **105**A as described with regard to FIGS. **4-5**, may operate the process **1600** as described below. While particular examples and orders of operations are illustrated in FIG. **16**, in various embodiments, these operations may be re-ordered, broken into additional operations, combined, and/or omitted altogether.

[0136] Referring to FIG. **16**, at operation **1603**, the ePOS client **110** may transmit a payment initiation to the merchant **140**. In various embodiments, operation **1603** may be performed during an ecommerce checkout procedure on a website associated with the merchant **140**. In response to the payment initiation, at operation **1606**, the merchant **140** may transmit a payment request to the cloud POS service **135**.

[0137] At operation **1609**, the cloud POS service **135** may transmit payment information to the messaging service **160**. The payment information may include client device identification information (e.g., MAC address of the computing system **105**, IP address of the computing system **105**, and/or any other type of identifier), merchant identification information (e.g., MAC address of the merchant **140**, IP address of the merchant **140**, and/or any other like identifier), a POS transaction amount (e.g., a purchase price), a currency type for the POS transaction, merchant-accepted payment credentials, and/or other like POS transaction-related data. At operation **1612**, the messaging service **160** may transmit the payment information notification to the ePOS client **110**.

[0138] At operation **1615**, the ePOS client **110** may enumerate the payment credentials. In various embodiments, the ePOS client **110** may query the PCDB **125** via the vPOS **120** to obtain a list of one or more payment credentials stored in the PCDB **125** that match the merchant-accepted payment credentials contained in the payment information notification.

[0139] At operation **1618**, the ePOS client **110** may receive a selection of the one or more enumerated payment credentials, and may provide the selection of the one or more enumerated payment credentials to the vPOS **120**. At operation **1621**, the vPOS **120** may obtain one or more authentication parameters and/or other like information associated with the selected payment credential from the PCDB **125**. As discussed previously, the authentication parameters may be security measures, defined by the selected payment credential, that are required for authenticating entities and/or communicating data between entities. At operation **1624**, the vPOS **120** may provide authentication parameters to the ePOS client **110**, which in turn may transmit the authentication parameters to the cloud POS service **135**.

[0140] At operation **1627**, the cloud POS service **135** may transmit merchant parameters and a merchant certificate to the ePOS client **110**, which in turn may provide the merchant parameters and the merchant certificate to the vPOS **120**. The merchant parameters may be security measures, defined by the merchant **140** and/or the payment acquiring service **145**, that are required for authenticating entities and/or communicating data between entities. In various embodiments, the stricter security requirements between the authentication parameters and the merchant parameters may be used to authenticate entities, encrypt payment data, encode data packets containing the payment data, and/or the like.

[0141] At operation **1630**, the vPOS **1430** may validate the merchant certificate and the merchant parameters. Upon properly validating the merchant certificate and the merchant parameters, at operation **1633**, the vPOS may provide transaction terms, an authentication challenge and a cryptographic client certificate to the ePOS client **110**, which in turn may transmit the transaction terms, the authentication challenge and the cryptographic client certificate to the cloud POS server **135**.

[0142] At operation **1636**, the cloud POS service **135** may validate the cryptographic client certificate and decrypt the authentication challenge. Upon properly decrypting the authentication challenge and properly validating the cryptographic client certificate, at operation **1639**, the cloud POS service **135** may transmit a PIN solicitation to the ePOS client **110**, and in response, the ePOS client **110** may transmit an inputted PIN to the cloud POS service **135**. At operation **1642**, the cloud POS service **135** may generate a PIN block based on the PIN received during the PIN solicitation. In various embodiments, the user may be required to enter a PIN in order to authorize use of the selected payment credential for the POS transaction. The PIN may be issued to the user by an issuer of the payment credential and may be used to authenticate the user's identity. In typical POS transactions, the cardholder is usually required to enter their PIN into a standalone physical POS terminal, the standalone physical POS terminal encodes the PIN into a PIN block and sends the PIN block to the merchant domain **130** for payment processing. In typical POS transactions where the payment credential is a smart card that includes a EMV chip, the cardholder is usually required to enter their PIN into the standalone physical POS terminal, and the standalone physical POS terminal sends the PIN to the to the smart card to check if the PIN is correct, wherein the smart card sends the result to the terminal so that the transaction continues if the PIN is correct. According to various example embodiments, when the user of the computing system **105** enters a PIN, the PIN may be encoded into a PIN block to protect the PIN during transmission between the ePOS client **110** and the cloud POS service **135**. In such embodiments, the PIN may be enciphered/deciphered and communicated according to ISO 9564, one or more EMV standards, and/or any other like standard. According to other embodiments, the PCDB **125** may store a PIN associated with the selected payment credential, and when the user of the computing system **105** enters a PIN, the entered PIN may be checked against the stored PIN. In some embodiments, the PIN may not be required to be enciphered when shared between the ePOS client **110** and the vPOS **120**, which may reduce the amount of computational resources used for processing the POS transaction. It should be noted that in other embodiments, other types of authentication methods may be used instead of, or in addition to using a PIN to authorize use payment credentials.

[0143] At operation **1645**, the cloud POS service **135** may transmit the ciphered PIN and updated transaction terms to the ePOS client **110**, which in turn, may provide the ciphered PIN and the updated transaction terms to the vPOS **120**. At operation **1648**, the vPOS **120** may verify the PIN and digitally sign the updated transaction terms.

[0144] At operation **1651**, the vPOS **120** may generate and encrypt payment data. In various embodiments, the payment data may be encrypted according to EMV standards. At operation **1654**, the vPOS **120** may provide the encrypted payment data to the POS UI, which in turn, may transmit the encrypted payment data to the cloud vPOS **135**, which in turn, may transmit the encrypted payment data to the acquirer **145**. At operation **1657**, the acquirer **145** may transmit a payment status message to the cloud POS service **135**. At operation **1660**, the cloud POS service **135** may transmit a payment status message to the merchant **140**. At operation **1663**, the merchant **140** may transmit a payment confirmation to the ePOS client **110**.

[0145] Some non-limiting Examples are provided below.

[0146] Example A01 includes a computer-implemented method for processing a point of sale (POS) transaction. The method may comprise receiving, by a virtual POS terminal within a trusted execution environment of a cloud trusted execution environment, a transaction initiation from a network element via a POS user interface (UI) module (or "POS client") wherein the transaction initiation indicates the POS transaction and one or more payment options to be used for processing the POS transaction; receiving, by the virtual POS terminal from the ePOS client, a selection of a payment credential matching one of the one or more payment options; obtaining, by the virtual POS terminal, the selected payment credential from within a payment credential storage unit located in the trusted execution environment; validating, by the virtual POS terminal, a merchant domain associated with the transaction initiation; encrypting, by the virtual POS terminal, payment data when the merchant domain is properly validated; and providing, by the virtual POS terminal, the encrypted payment data to the ePOS client wherein the ePOS client communicates the encrypted payment data to the network element.

[0147] Example A02 includes the method of the preceding example, and/or according to any example disclosed herein, wherein the payment credential storage unit stores a plurality of payment credentials and wherein obtaining the selected payment credential comprises obtaining one of the plurality of payment credentials.

[0148] Example A03 includes the method of any of the preceding examples, and/or according to any example disclosed herein, wherein the selected payment credential defines authentication parameters required to validate the merchant identity and process the POS transaction using the payment credential, and the method further comprises providing the authentication parameters to the ePOS client, wherein the ePOS client is to provide the authentication parameters to the network element.

[0149] Example A04 includes the method of any of the preceding examples, and/or according to any example disclosed herein, further comprising receiving, via the ePOS client, a cryptographic merchant certificate wherein the merchant certificate is based on the authentication parameters, wherein the validating comprises decrypting the cryptographic merchant certificate; generating transaction data upon validation of the merchant domain, wherein the transaction data includes a cryptographic client certificate, payment credential transaction terms defined by the authentication parameters, and a merchant authentication challenge; and encrypting the transaction data.

[0150] Example A05 includes the method of any of the preceding examples, and/or according to any example disclosed herein, further comprising receiving, from the network element via the ePOS client, a PIN block; deciphering the PIN block; and upon properly deciphering the PIN block, generating the payment data wherein the payment data includes a digital signature associated with the payment credential and a payment address associated with the payment credential.

[0151] Example A06 includes the method of any of the preceding examples, and/or according to any example disclosed herein, wherein the payment credential storage unit stores a plurality of passcodes in association with a corresponding one of a plurality of payment credentials, wherein each of the plurality of passcodes is to be entered to authorize use of the corresponding one of the plurality of payment credentials, and the selected payment credential is one of the plurality of payment credentials, and the method further comprises providing an indication to the ePOS client to generate a UI for inputting passcodes; receiving an input passcode from the ePOS client; determining whether the input passcode is equal to a passcode stored in association with the selected payment credential; and authorizing the selected payment credential to be used for processing the POS transaction when the input passcode is determined to be equal to the passcode stored in association with the selected payment credential.

[0152] Example A07 includes the method of any of the preceding examples, and/or according to any example disclosed herein, wherein receiving a transaction initiation comprises receiving a transaction initiation that includes a purchase price of the POS transaction and a currency to be used to process the POS transaction.

[0153] Example A08 includes the method of any of the preceding examples, and/or according to any example disclosed herein, wherein the receiving of a transaction initiation, the receiving of a selection of a payment credential, the obtaining, the validating, the encrypting and the providing are performed by the virtual POS terminal operating on a secure processor of the trusted execution environment, with the ePOS client being an only interface communicatively coupled with the virtual POS terminal.

[0154] Example A09 includes the method of any of the preceding examples, and/or according to any example disclosed herein, further comprising receiving, by the ePOS client, the transaction initiation via a text message or a messaging service message that includes a unique identifier of the computing device provided by a user of the computing device.

[0155] Example B01 includes a computing device first means to receive a transaction initiation, and provide a selection of a payment credential to be used to process a POS transaction. The computing device comprises a second means, communicatively coupled with the first means, to process the POS transaction in response to the selection of the payment credential. The second means comprises a third means to store the selected payment credential; and a fourth means to validate a merchant associated with the transaction initiation, process the POS transaction using the selected payment credential to generate payment data, and encrypt the payment data. The first means is further to receive the encrypted payment data from the fourth means, and communicate the encrypted payment data to a network element.

[0156] Example B02 includes the computing device of the preceding example, and/or according to any example disclosed herein, wherein the first means is to receive a transaction initiation that indicates one or more payment options to be used for the POS transaction, and the selected payment credential is to match one of the one or more payment options.

[0157] Example B03 includes the computing device of any of the preceding examples, and/or according to any example disclosed herein, wherein the first means is to provide a selected payment credential that defines authentication parameters required to validate the merchant identity and the second means is to process the POS transaction using the payment credential, wherein the fourth means is to provide the authentication parameters to the first means, and the first means is to provide the authentication parameters to the network element.

[0158] Example B04 includes the computing device of any of the preceding examples, and/or according to any example disclosed herein, wherein the first means is to receive a cryptographic merchant certificate wherein the merchant certificate is based on the authentication parameters, and the fourth means is to decrypt the cryptographic merchant certificate to validate the merchant, and upon validation of the merchant, the fourth means is to generate and encrypt transaction data, wherein the transaction data includes a cryptographic client certificate, payment credential transaction terms defined by the authentication parameters, and a merchant authentication challenge.

[0159] Example B05 includes the computing device of any of the preceding examples, and/or according to any example disclosed herein, wherein the first means is to receive a personal identification number, PIN, solicitation upon proper decryption of the cryptographic client certificate by the network element, and upon proper decryption of the merchant authentication challenge, and in response to PIN solicitation, the first means is to provide a UI to input a PIN, and communicate the input PIN to the network element, wherein the first means is to receive a PIN block and updated transaction terms upon validation of the input PIN.

[0160] Example B06 includes the computing device of any of the preceding examples, and/or according to any example disclosed herein, wherein the first means is to receive

updated transaction terms that are based on a combination of the payment credential transaction terms and merchant required transaction terms, and provide the updated transaction terms to the fourth means, and the fourth means is to accept or deny the updated transaction terms, process the POS transaction according to the payment credential transaction terms when the fourth means denies the updated transaction terms, and process the POS transaction according to the updated transaction terms when the fourth means accepts the updated transaction terms.

[0161] Example B07 includes the computing device of any of the preceding examples, and/or according to any example disclosed herein, wherein the first means is to receive updated transaction terms that include transaction terms which are common to both the payment credential transaction terms and merchant required transaction terms, and any transaction terms that are required by one of the payment credential transaction terms or the merchant required transaction terms but not including the other one of the payment credential transaction terms or the merchant required transaction terms, and provide the updated transaction terms to the fourth means, and the fourth means is to process the POS transaction according to the updated transaction terms.

[0162] Example B08 includes the computing device of any of the preceding examples, and/or according to any example disclosed herein, wherein the first means is to receive PIN block that is enciphered using one of format 0, format 1, format 2, or format 3 in accordance with International Organization for Standardization (ISO) 9564.

[0163] Example B09 includes the computing device of any of the preceding examples, and/or according to any example disclosed herein, wherein the fourth means is to receive the PIN block from the first means, decipher the PIN block, and upon a proper decipher of the PIN block, the fourth means is to generate the payment data wherein the payment data includes a digital signature associated with the payment credential and a payment address associated with the payment credential.

[0164] Example B10 includes the computing device of any of the preceding examples, and/or according to any example disclosed herein, wherein the first means is to receive a payment confirmation from the merchant when the encrypted payment information is properly decrypted by a payment acquiring service associated with the payment credential.

[0165] Example B11 includes the computing device of any of the preceding examples, and/or according to any example disclosed herein, wherein the third means is to store a plurality of payment credentials, and the first means is to display a set of the plurality of payment credentials based on information contained in the transaction initiation.

[0166] Example B12 includes the computing device of any of the preceding examples, and/or according to any example disclosed herein, wherein the third means is to store, in association with each of the plurality of payment credentials, information indicating one or more jurisdictions in which each of the plurality of payment credentials are accepted, and the first means is to display ones of the plurality of payment credentials that are accepted within a jurisdiction of the merchant based on a geographic location of the merchant, wherein the information contained in the transaction initiation includes information indicating the geographic location of the merchant.

[0167] Example B13 includes the computing device of any of the preceding examples, and/or according to any example disclosed herein, wherein the third means is to store a plurality of passcodes in association with a corresponding one of a plurality of payment credentials wherein the selected payment credential is one of the plurality of payment credentials, and the passcode stored in association with the selected payment credential is to be entered to authorize use of the selected payment credential, and wherein the first means is to provide a UI for input of passcodes.

[0168] Example B14 includes the computing device of any of the preceding examples, and/or according to any example disclosed herein, wherein the first means is to receive a transaction initiation that includes a purchase price of the POS transaction and a currency to be used to process the POS transaction.

[0169] Example B15 includes the computing device of any of the preceding examples, and/or according to any example disclosed herein, wherein the second means comprises another processor, the fourth means is to operate on the other processor to process the POS transaction, and the first means is an only module communicatively coupled with the fourth means.

[0170] Example B16 includes the computing device of any of the preceding examples, and/or according to any example disclosed herein, wherein the second means is one of Intel® Management Engine, Intel® Software Guard Extensions, or Intel® Converged Security Engine (CSE).

[0171] Example B17 includes the computing device of any of the preceding examples, and/or according to any example disclosed herein, wherein the first means is to receive the transaction initiation via a text message or a messaging service message wherein a unique identifier of the computing device is provided by a user of the computing device to initialize the transaction initiation.

[0172] Example B18 includes the computing device of any of the preceding examples, and/or according to any example disclosed herein, wherein the text message is one of a short message service (SMS) message, a multimedia messaging service (MMS) message, an Over-the-Top (OTT) message, a push notification, or an email message.

[0173] Example C01 includes a computing system comprising: application processor circuitry arranged to operate a point of sale (POS) client to: receive a transaction initiation, and provide a selection of a payment credential to be used to process a POS transaction; a trusted execution environment (TEE), the TEE communicatively coupled with the application processor circuitry, the TEE arranged to process the POS transaction in response to the selection of the payment credential, wherein the TEE comprises: a payment credential storage unit arranged to store one or more payment credentials including the selected payment credential; and a virtual POS terminal arranged to validate a merchant associated with the transaction initiation, obtain the selected payment credential from the payment credential storage unit, process the POS transaction using the selected payment credential to generate payment data, and encrypt the payment data, wherein the POS client is arranged to receive the encrypted payment data from the virtual POS terminal; and network interface circuitry communicatively coupled with the application processor circuitry, the network interface circuitry arranged to communicate the encrypted payment data to a merchant system owned or operated by the merchant.

[0174] Example C02 includes the computing system of example C01 and/or some other examples herein, wherein the transaction initiation indicates one or more payment options to be used for the POS transaction, and the selected payment credential is to match one of the one or more payment options.

[0175] Example C03 includes the computing system of examples C01-C02 and/or some other examples herein, wherein the POS client is arranged to provide a selected payment credential that defines authentication parameters required to validate a merchant identity of the merchant and the trusted execution environment is to process the POS transaction using the payment credential, wherein the virtual POS terminal is arranged to provide the authentication parameters to the network interface circuitry via the POS client for transmission to the merchant system.

[0176] Example C04 includes the computing system of example C03 and/or some other examples herein, wherein: the POS client is arranged to receive, via the network interface circuitry and the POS client, a cryptographic merchant certificate that is based on the authentication parameters, and the virtual POS terminal is arranged to decrypt the cryptographic merchant certificate to validate the merchant identity, and upon validation of the merchant identity, the virtual POS terminal is arranged to generate and encrypt transaction data, wherein the transaction data includes a cryptographic client certificate, payment credential transaction terms defined by the authentication parameters, and a merchant authentication challenge.

[0177] Example C05 includes the computing system of example C04 and/or some other examples herein, wherein the POS client is arranged to: receive, via the network interface circuitry and the POS client, a personal identification number (PIN) solicitation upon proper decryption of the cryptographic client certificate by the merchant system; and upon proper decryption of the merchant authentication challenge and in response to PIN solicitation, the POS client is arranged to: cause a UI to input a PIN to be generated and displayed; and provide the input PIN to the network interface circuitry via the POS client for transmission to the merchant system, wherein the POS client is arranged to receive, from the merchant system via the network interface circuitry, a PIN block and updated transaction terms upon validation of the input PIN.

[0178] Example C06 includes the computing system of example C05 and/or some other examples herein, wherein the updated transaction terms are based on a combination of the payment credential transaction terms and merchant required transaction terms, and wherein: the POS client is arranged to provide the updated transaction terms to the virtual POS terminal, and the virtual POS terminal is arranged to accept or deny the updated transaction terms, process the POS transaction according to the payment credential transaction terms when the virtual POS terminal denies the updated transaction terms, and process the POS transaction according to the updated transaction terms when the virtual POS terminal accepts the updated transaction terms.

[0179] Example C07 includes the computing system of examples C05-C06 and/or some other examples herein, wherein the updated transaction terms include transaction terms which are common to both the payment credential transaction terms and merchant required transaction terms, wherein any transaction terms are required by one of the payment credential transaction terms or the merchant required transaction terms but not included in the other one of the payment credential transaction terms or the merchant required transaction terms, and wherein: the POS client is arranged to provide the updated transaction terms to the virtual POS terminal, and the virtual POS terminal is arranged to process the POS transaction according to the updated transaction terms.

[0180] Example C08 includes the computing system of examples C05-C07 and/or some other examples herein, wherein the virtual POS terminal is arranged to receive the PIN block from the POS client, decipher the PIN block, and upon a proper decipher of the PIN block, the virtual POS terminal is arranged to generate the payment data wherein the payment data includes a digital signature associated with the payment credential and a payment address associated with the payment credential.

[0181] Example C09 includes the computing system of example C08 and/or some other examples herein, wherein the POS client is arranged to receive a payment confirmation from the merchant system via the network interface circuitry when the encrypted payment information is properly decrypted by a payment acquiring service associated with the payment credential.

[0182] Example C10 includes the computing system of examples C01-C09 and/or some other examples herein, wherein the payment credential storage unit is arranged to store a plurality of payment credentials, and the POS client is arranged to cause display of a set of the plurality of payment credentials based on information contained in the transaction initiation.

[0183] Example C11 includes the computing system of examples C01-C10 and/or some other examples herein, wherein the payment credential storage unit is arranged to store a plurality of passcodes in association with a corresponding one of a plurality of payment credentials wherein the selected payment credential is one of the plurality of payment credentials, and the passcode stored in association with the selected payment credential is to be entered to authorize use of the selected payment credential, and wherein the POS client is arranged to cause a UI for input of passcodes to be displayed.

[0184] Example C12 includes the computing system of examples C01-C11 and/or some other examples herein, wherein the transaction initiation includes a purchase price of the POS transaction and a currency to be used to process the POS transaction.

[0185] Example C13 includes the computing system of examples C01-C12 and/or some other examples herein, wherein the TEE is a tamper-resistant chipset including a secure processor, and the virtual POS terminal is arranged to operate on the secure processor to process the POS transaction, and the POS client is an only module outside of the TEE communicatively coupled with the virtual POS terminal.

[0186] Example C14 includes the computing system of examples C01-C12 and/or some other examples herein, wherein the TEE is one of Intel® Management Engine, Intel® Software Guard Extensions, or Intel® Converged Security Engine (CSE).

[0187] Example C15 includes the computing system of examples C01-C14 and/or some other examples herein, wherein the network interface circuitry is arranged to receive the transaction initiation via a text message or a

messaging service message, wherein a unique identifier of the computing system is provided by a user of the computing system to initialize the transaction initiation.

[0188] Example C16 includes the computing system of example C15 and/or some other examples herein, wherein the text message is a short message service (SMS) message, a multimedia messaging service (MMS) message, an Over-the-Top (OTT) message, a push notification, or an email message.

[0189] Example D01 includes a virtual point of sale (POS) method for processing a POS transaction, the method comprising: receiving a transaction initiation from a merchant system via network interface circuitry of a computing system that includes the computing device and a POS user interface (UI) module operated by an application processor of the computing system, wherein the transaction initiation indicates the POS transaction and one or more payment options to be used for processing the POS transaction; receiving, from the POS client, a selection of a payment credential matching one of the one or more payment options; obtaining the selected payment credential from within a payment credential storage unit located in a trusted execution environment (TEE); validating a merchant domain associated with the transaction initiation, the merchant domain including the merchant system; encrypting payment data when the merchant domain is properly validated; and providing the encrypted payment data to the network interface circuitry via the POS client for communication of the encrypted payment data to the merchant system.

[0190] Example D02 includes the method of example D01 and/or some other examples herein, wherein the payment credential storage unit stores a plurality of payment credentials, and wherein obtaining the selected payment credential comprises: obtaining one of the plurality of payment credentials from the payment credential storage unit.

[0191] Example D03 includes the method of examples D01-D02 and/or some other examples herein, wherein the selected payment credential defines authentication parameters required to validate a merchant identity of the merchant domain and process the POS transaction using the payment credential, and the method comprises: providing the authentication parameters to the network interface circuitry via the POS client for transmission to the merchant system.

[0192] Example D04 includes the method of example D03 and/or some other examples herein, further comprising: receiving, from the merchant system via the network interface circuitry and the POS client, a cryptographic merchant certificate wherein the merchant certificate is based on the authentication parameters, wherein validating the merchant domain comprises: decrypting the cryptographic merchant certificate, generating transaction data upon validation of the merchant domain, wherein the transaction data includes a cryptographic client certificate, payment credential transaction terms defined by the authentication parameters, and a merchant authentication challenge, and encrypting the transaction data.

[0193] Examples D05 includes the method of example D04 and/or some other examples herein, further comprising: receiving, from the merchant system via the network interface circuitry and the POS client, a PIN block; deciphering the PIN block; and generating, upon proper decipher of the PIN block, the payment data wherein the payment data includes a digital signature associated with the payment credential and a payment address associated with the payment credential.

[0194] Examples D06 includes the method of examples D01-D05 and/or some other examples herein, wherein the payment credential storage unit stores a plurality of passcodes in association with a corresponding one of a plurality of payment credentials, wherein each of the plurality of passcodes is to be entered to authorize use of the corresponding one of the plurality of payment credentials, and the selected payment credential is one of the plurality of payment credentials, and the method comprises: providing an indication to the POS client to generate and display a UI for inputting passcodes; receiving an input passcode from the POS client; determining whether the input passcode is equal to a passcode stored in association with the selected payment credential; and authorizing the selected payment credential to be used for processing the POS transaction when the input passcode is determined to be equal to the passcode stored in association with the selected payment credential.

[0195] Example D07 includes the method of examples D01-D06 and/or some other examples herein, wherein the transaction initiation includes a purchase price of the POS transaction and a currency to be used to process the POS transaction.

[0196] Example D08 includes the method of examples A17-D07 and/or some other examples herein, wherein the computing device is a secure processor of the TEE, and wherein the POS client is to be operated by a host platform of a computing system including the secure processor, and the POS client is an only interface outside of the TEE that is communicatively coupled with the virtual POS terminal.

[0197] Example D09 includes the method of examples D01-D08 and/or some other examples herein, wherein the transaction initiation is a text message or a messaging service message that includes a unique identifier of the computing device provided by a user of the computing device.

[0198] Example Z01 may include an apparatus comprising means to perform one or more elements of a method described in or related to any of examples A01-A09, B01-B18, C01-C16, D01-D09, or any other method or process described herein. Example Z02 may include one or more non-transitory computer-readable media comprising instructions to cause an electronic device, upon execution of the instructions by one or more processors of the electronic device, to perform one or more elements of a method described in or related to any of examples A01-A09, B01-B18, C01-C16, D01-D09, or any other method or process described herein. Example Z03 may include an apparatus comprising logic, modules, or circuitry to perform one or more elements of a method described in or related to any of examples A01-A09, B01-B18, C01-C16, D01-D09, or any other method or process described herein. Example Z04 may include a method, technique, or process as described in or related to any of examples A01-A09, B01-B18, C01-C16, D01-D09, or portions or parts thereof. Example Z05 may include an apparatus comprising: one or more processors and one or more computer-readable media comprising instructions that, when executed by the one or more processors, cause the one or more processors to perform the method, techniques, or process as described in or related to any of examples A01-A09, B01-B18, C01-C16, D01-D09, or portions thereof. Example Z06 may include a signal as

described in or related to any of examples A01-A09, B01-B18, C01-C16, D01-D09, or portions or parts thereof. Example Z07 may include a datagram, packet, frame, segment, protocol data unit (PDU), or message as described in or related to any of examples A01-A09, B01-B18, C01-C16, D01-D09, or portions or parts thereof, or otherwise described in the present disclosure. Example Z08 may include a signal encoded with data as described in or related to any of examples A01-A09, B01-B18, C01-C16, D01-D09, or portions or parts thereof, or otherwise described in the present disclosure. Example Z09 may include a signal encoded with a datagram, packet, frame, segment, protocol data unit (PDU), or message as described in or related to any of examples A01-A09, B01-B18, C01-C16, D01-D09, or portions or parts thereof, or otherwise described in the present disclosure. Example Z10 may include an electromagnetic signal carrying computer-readable instructions, wherein execution of the computer-readable instructions by one or more processors is to cause the one or more processors to perform the method, techniques, or process as described in or related to any of examples A01-A09, B01-B18, C01-C16, D01-D09, or portions thereof. Example Z11 may include a computer program comprising instructions, wherein execution of the program by a processing element is to cause the processing element to carry out the method, techniques, or process as described in or related to any of examples A01-A09, B01-B18, C01-C16, D01-D09, or portions thereof.

[0199] Although certain embodiments have been illustrated and described herein for purposes of description, a wide variety of alternate and/or equivalent embodiments or implementations calculated to achieve the same purposes may be substituted for the embodiments shown and described without departing from the scope of the present disclosure. This application is intended to cover any adaptations or variations of the embodiments discussed herein, limited only by the claims.

1-25. (canceled)

26. A mobile computing system comprising:

a rich execution environment (REE) to communicatively couple with a trusted execution environment (TEE) during operation of the mobile computing system, the TEE and the REE are to operate in isolation from one another;

the REE is arranged to operate a point of sale (POS) client, the POS client is to provide a user input to a virtual POS terminal (vPOS) operating within the TEE; and

the TEE is arranged to operate the vPOS to: authenticate a user of the mobile computing system based on the user input, process a transaction while a trusted state with a cloud POS service is maintained, and prevent access to the vPOS when the trusted state with the cloud POS service is not maintained.

27. The computing system of claim **26**, wherein maintenance of the trusted state with the cloud POS service takes place over a network connection between the vPOS and the cloud POS service.

28. The computing system of claim **26**, wherein the TEE further includes a cloud POS service security domain (SD), and maintenance of the trusted state with the cloud POS service takes place between the vPOS and the cloud POS service SD.

29. The computing system of claim **26**, wherein, to prevent access to the vPOS, the vPOS is to transition from an active state to an inactive state.

30. The computing system of claim **26**, wherein the POS client is to:

receive a transaction initiation, the transaction initiation indicates one or more payment options to be used for the transaction;

receive another user input indicating a selected payment credential from among the one of the one or more payment options;

provide an indication of the selected payment credential to the vPOS.

31. The computing system of claim **30**, wherein the selected payment credential defines authentication parameters required to validate an identity of a party to the transaction, and the vPOS is to:

provide the authentication parameters for transmission to another computing system operated by the party to the transaction;

receive a first cryptographic certificate based on the authentication parameters from the other computing system;

decrypt the first cryptographic certificate to validate the identity of the party to the transaction, and

upon validation of the identity, generate and encrypt transaction data, the transaction data including a second cryptographic certificate, payment credential transaction terms defined by the authentication parameters, and an authentication challenge.

32. The computing system of claim **31**, wherein the POS client is to:

receive an authentication solicitation upon proper decryption of the second cryptographic certificate by the other computing system; and

after proper decryption of the authentication challenge and in response to authentication solicitation, generate and render a user interface to obtain the user input to authenticate the user.

33. The computing system of claim **32**, wherein the vPOS is to:

receive, from the other computing system, updated transaction terms upon validation of the user, the updated transaction terms being based on a combination of the payment credential transaction terms and transaction terms of the other computing system;

process the transaction according to the payment credential transaction terms when the vPOS denies the updated transaction terms; and

process the transaction according to the updated transaction terms when the vPOS accepts the updated transaction terms.

34. The computing system of claim **33**, wherein the vPOS is to:

receive a personal identification number (PIN) block from the other computing system; and

upon a proper decipher of the PIN block, generate payment data to include a digital signature associated with the payment credential.

35. The computing system of claim **29**, wherein the transaction initiation includes a transaction amount of the POS transaction and a currency value to be used to process the POS transaction.

36. One or more non-transitory computer-readable media (NTCRM) comprising instructions for a virtual point of sale terminal (vPOS), wherein execution of the instructions by a trusted execution environment (TEE) is to cause the TEE to:

obtain a user input from a point of sale (POS) client that is to operate within a rich execution environment (REE), the REE is to operate in isolation from the TEE;

authenticate a user of a mobile computing system based on the user input;

process a transaction while a trusted state with a cloud POS service is maintained; and

prevent access to the vPOS when the trusted state with the cloud POS service is not maintained.

37. The one or more NTCRM of claim **36**, wherein the TEE further includes a cloud POS service security domain (SD), and maintenance of the trusted state with the cloud POS service takes place between the vPOS and the cloud POS service SD via a network connection between the vPOS and the cloud POS service.

38. The one or more NTCRM of claim **36**, wherein the REE and the TEE are implemented in a same mobile computing system.

39. The one or more NTCRM of claim **36**, wherein the REE is implemented in a mobile computing system and the TEE is implemented by one or more compute nodes of a cloud computing service.

40. The one or more NTCRM of claim **36**, wherein execution of the instructions is to cause the TEE to:

receive an indication of a selected payment credential from the POS client, wherein the selected payment credential defines authentication parameters required to validate an identity of a party to the transaction;

send the authentication parameters to a computing system operated by the party to the transaction;

receive a first cryptographic certificate based on the authentication parameters from the computing system operated by the party to the transaction;

decrypt the first cryptographic certificate to validate the identity of the party to the transaction;

upon validation of the identity of the party to the transaction, generate and encrypt transaction data, the transaction data including a second cryptographic certificate, payment credential transaction terms defined by the authentication parameters, and an authentication challenge;

receive, from the computing system operated by the party to the transaction, updated transaction terms upon validation of the user, the updated transaction terms being based on a combination of the payment credential transaction terms and transaction terms of the other computing system;

process the transaction according to the payment credential transaction terms when the vPOS denies the updated transaction terms; and

process the transaction according to the updated transaction terms when the vPOS accepts the updated transaction terms.

41. A method of operating a point of sale terminal (POS) client, the POS client is to operate within a rich execution environment (REE) of a mobile computing system, the REE being isolated from a trusted execution environment (TEE), the method comprising:

receiving a transaction initiation from a remote computing system;

generating and rendering a graphical user interface (GUI) in response to receipt of the transaction initiation, the GUI including one or more graphical control elements (GCEs) for selection of a credential;

obtaining, via the GUI, an indication of a selected credential in response to a selection of a GCE of the one or more GCEs;

providing the indication of the selected credential to a virtual POS terminal (vPOS) operating within the TEE to process the transaction.

42. The method of claim **41**, wherein the selected credential defines authentication parameters required to validate an identity of a party to the transaction.

43. The method of claim **42**, further comprising:

receiving encrypted transaction data from the vPOS, the transaction data being based on the authentication parameters; and

transmitting the encrypted transaction data to the remote computing system.

43. The method of claim **42**, further comprising:

receiving a personal identification number (PIN) solicitation upon proper authentication of the user by the remote computing system;

generating and rendering another GUI including one or more other GCEs for input of a PIN;

obtaining, via the other GUI, an input PIN in response to a selection of some or all of the one or more other GCEs; and

providing the input PIN to the vPOS operating within the TEE to authenticate the user.

44. The method of claim **41**, wherein the transaction initiation includes a transaction amount of the transaction or a currency value to be used to process the transaction.

45. The method of claim **41**, wherein the transaction initiation is included in a short message service (SMS) message, a multimedia messaging service (MMS) message, an Over-the-Top (OTT) message, a push notification, or an email message.

* * * * *