

PROVISIONING OF AUTHENTICATION CREDENTIALS ON MOBILE DEVICES

Technical Field

[001] The present invention relates to methods and systems for provisioning and use of authentication credentials on mobile devices.

Background

[002] Smart cards are used in many different contexts, in order to provide access to services. In some cases this may include providing accessibility to secured physical environments. For example, transit systems or other physical environments may have physical barriers that can be opened via the transmission and validation of authentication credentials received from the smart card by a communications interface, such as a near-field communications (NFC) or other wireless interface. Smart cards can also be used to provide authentication credentials to computer systems to enable users to access secured digital environments (e.g. strong security authentication for single sign-on), or to effect payment or non-payment transactions.

[003] More recently it has become possible to provision card-based authentication credentials onto smartphones and other mobile computing devices that operate via contactless technologies such as NFC, magnetic secure transmission (MST) and the like. Accordingly, the mobile device emulates, and may conveniently be used in place of, a physical smart card. This is advantageous in that multiple physical cards can be provisioned onto a single mobile device, such that the user only needs to carry the mobile device, and not necessary all their smart cards, to be able to provide authentication credentials to access multiple physical and/or digital environments. The process of provisioning the authentication credentials may also be called digitization.

[004] Typically, a mobile device may have means for containerizing the multiple authentication credentials such that a single (secure) application has access to, and control of, the multiple authentication credentials to access different services. This secure container application may be referred to in at least some contexts as a digital

wallet. Digital wallets are typically capable of storing or otherwise maintaining availability of both payment and non-payment credentials such as those for accessing physically secured environments as noted above.

[005] To prevent malicious actors from obtaining and misusing the authentication credentials, various digital security measures are implemented in devices that have digital wallet capability. For example, some devices have a dedicated hardware component called a secure element (SE) that is hardened against cryptographic attack. Such devices may store the authentication credentials, the secure application, or both, on the SE in encrypted form. Devices that do not have a SE may instead have a secure portion of the main processor called a trusted execution environment (TEE) that is an isolated environment in which the secure application executes, and in which the credentials are stored.

[006] Other devices may implement digital wallet functionality via a technique called Host Card Emulation (HCE). In HCE implementations, the authentication credentials are typically not stored on the device at all, but rather in a secure remote system, and retrieved by the mobile device on an as-needed basis. For example, the secure remote system may be accessed by the mobile device on a per-transaction basis to obtain a one-time credential, or periodically to obtain a limited-use credential or batch of one-time credentials.

[007] Typically, an HCE implementation will make use of a tokenization process that replaces the actual credential (or part thereof) with a surrogate value called a token. For example, a tokenization service may map a payment account number, or at least some of the digits of the payment account number (e.g., all but the first 6 digits), to another number that has the same format, but from which the real payment account number cannot be obtained without having knowledge of the mapping. The token is usually associated with that specific mobile device. Accordingly, an attacker obtaining the token cannot use it outside the environment for which it was assigned. The secure remote system maintains a data store called a token vault, which stores the mapping between the actual credential and the token, possibly together with other information such as an identifier of the mobile device, and various restrictions on use or validity of the token (e.g., single-use, valid only for a certain time period, or valid for use only in specific environments or for specific transactions).

[008] Tokenization is not exclusive to HCE digital wallet implementations. The same technique can be implemented with SE or TEE implementations in order to further improve security of those implementations. For example, the Apple Pay digital wallet application is implemented using the SE of the iPhone or Apple Watch, and also makes use of tokenization.

[009] One problem that arises in relation to digitization of debit cards in particular is that in most countries, debit cards are issued using domestic debit chip specifications governed by local regulatory standards, and these are different from the specifications used by card networks such as Mastercard (which has a chip specification called M/Chip) and Visa. Accordingly, although the smart card may have on its face a Mastercard (for example) card number, the debit application stored thereon does not operate in accordance with the M/Chip specification, and transactions initiated by the smart card are switched by the domestic debit network (also referred to herein as a domestic switch) and not the Mastercard network. Domestic debit networks do not have the ability to tokenize domestic debit cards to provision them to digital wallets. In the absence of a routing path for an authorization request to reach a tokenization service such as that provided by Mastercard, there is no way for transactions using such digitized smart cards to be processed.

[010] The present invention seeks to address one or more of the above difficulties, or at least to provide a useful alternative.

Summary

[011] In a first aspect, the present disclosure relates to a method for provisioning authentication credentials on a mobile device, the method comprising:

- receiving a provisioning request message comprising: one or more authentication credentials of a smart card, including at least an identifier of the smart card, and a request for card personalization keys;

- determining a type of the smart card based on the identifier of the smart card;

- generating, based on the type of the smart card, a card personalization script comprising computer-readable instructions for configuring the mobile device to

emulate the smart card;

- generating a token based on the authentication credentials or part thereof;

- generating one or more card personalization keys, respective card personalization keys being for different respective authentication types; and

- transmitting, to the mobile device, the card personalization script for execution by the mobile device;

wherein the card personalization script comprises computer-readable instructions to cause the mobile device to retrieve the one or more card personalization keys and the token to thereby enable emulation of the smart card by the mobile device.

[012] A second aspect of the present disclosure relates to a method of processing an authentication request, comprising:

- providing, at a network switch, a routing table comprising a mapping of smart card identifier ranges to access management nodes;

- receiving, at the network switch, an authorization request message from a digital wallet application executing on a mobile device, the authorization request message comprising a token that is representative of an authentication credential, and a request to perform an action at one of said access management nodes;

- determining, using the token and the routing table, which access management node to transmit the authentication request to; and

- transmitting the authorization request message to the access management node, without detokenizing the token.

[013] A third aspect of the present disclosure relates to a system for provisioning authentication credentials on a mobile device, comprising at least one processor in communication with computer-readable storage that stores instructions for causing the at least one processor to carry out a method according to the first aspect.

[014] A fourth aspect of the present disclosure relates to a system for processing an authentication request, comprising at least one processor in communication with computer-readable storage that stores instructions for causing the at least one processor to carry out a method according to the second aspect.

[015] In a fifth aspect, the present disclosure relates to a system for provisioning authentication credentials on a mobile device, comprising an access management node and a first network switch, the access management node being configured to:

- receive, from the first network switch, a provisioning request message comprising: one or more authentication credentials of a smart card, including at least an identifier of the smart card, and a request for card personalization keys;
- determine a type of the smart card based on the identifier of the smart card;
- generate, based on the type of the smart card, a card personalization script comprising computer-readable instructions for configuring the mobile device to emulate the smart card;
- generate a token based on the authentication credentials, or part thereof;
- generate one or more card personalization keys, respective card personalization keys being for different respective authentication types; and
- transmit, to the mobile device via the first network switch, the card personalization script for execution by the mobile device;

wherein the card personalization script, when executed, causes the mobile device to retrieve the one or more card personalization keys and the token to thereby enable emulation of the smart card by the mobile device, such that the mobile device is configured to transmit the token to the access management node as part of an access request, via a second network switch, without the second network switch detokenizing the token.

[016] In a sixth aspect, the present disclosure relates to a non-volatile computer-readable storage medium containing instructions which, when executed by at least one processor, cause the at least one processor to carry out a method according to the first or second aspects.

Brief Description of the Drawings

[017] Certain embodiments of the invention are hereafter described, by way of non-limiting example only, with reference to the accompanying drawings, in which:

- Figure 1 is a schematic diagram of an access management system;
- Figure 2 is a flowchart of an authentication credential provisioning process;
- Figure 3 is a flowchart of a service request authorization process;

Figure 4 is a block diagram of an example mobile computing device of the system shown in Figure 1; and

Figure 5 is a schematic diagram showing components of an example authorization system of the system of Figure 1.

Detailed Description

[018] Figure 1 shows an access management system 100. The system 100 allows a user of a mobile device 12 to provision smart card authentication credentials, for example into a digital wallet, via a first network switch 22, such that the authentication credentials are usable in an authorization process to authenticate the user and/or the device 12 over a second, different, network switch 24. For example, the first network switch 22 may be a payment network switch such as Banknet™ operated by Mastercard International Incorporated, and the second network switch 24 may be a domestic switch such as NETS (Singapore), ITMX (Thailand), EFTPOS (Australia/New Zealand), and the like.

[019] Rather than using the actual credentials in the authentication process, which may be sensitive (e.g. a primary account number (PAN) of a debit or credit card, expiry date, card verification value and so forth), a digital wallet may use a tokenized version of those credentials (referred to as a token) for authentication. The system 100 is therefore capable of processing transactions made using a digital wallet, and processed by a domestic switch, without substantial changes being made at the domestic switch.

[020] As referred to herein, authentication credentials for a smart card are data elements that are used by the smart card to identify itself to a remote system as part of an access request and/or to allow the remote system to verify that the user of the smart card is entitled to access the service. Authentication credentials may include one or more of the following: card data such as an identifier, an expiry date, a card security code (also known as a CVV or CVC), or a locale (such as country) of issuance of the smart card; user identification data, such as a billing address, or biometric data (such as fingerprint or facial recognition data); a username and/or password; and cryptographic data, such as one or more encryption keys or public/private key pairs, and/or a digital certificate.

[021] In certain embodiments, the access management system 100 comprises:

- (a) mobile computing device 12;
- (b) digital wallet system 20;
- (c) first switch 22;
- (d) point of interaction (POI) device 14;
- (e) acquirer system 16;
- (f) second switch 24; and
- (g) access management node 30.

[022] The components of system 100 are in communication via a network such as the Internet, a mobile telecommunications network and/or one or more local area networks (not shown).

[023] The system 100 enables tokenized transactions, including digital wallet transactions, to be processed over domestic switch payment systems such as second switch 24. Typically, domestic switch payment systems do not support tokenized transactions and as such, are not able to process digital wallet transactions on the domestic switch payment network. The first switch 22 provides tokenization functionality and operates in conjunction with the second (domestic) switch 24, thereby allowing the tokenized transaction to be processed by the second switch 24.

[024] Only one access management node 30 is shown in Figure 1. However, in practice, multiple access management nodes 30 may form part of the system 100, and respective access management nodes 30 may control access to different services. For example, one node or set of nodes may control access to a financial service such as an electronic payment service, while another node or set of nodes may control access to a transit service. In some contexts a node or set of nodes may be referred to as an "issuer" or "issuer host system" as it, or they, are controlled by an entity that is responsible for issuing (manufacturing, or otherwise generating, and personalizing) smart cards.

[025] In some embodiments, each node 30 may maintain a database of card identifier ranges for cards that are authorized to access services managed by the node 30. For example, in the context of financial services, the card identifiers may be PANs

of payment cards, and the card identifier ranges may be a BIN (bank identification number) range or set of BINs that is managed by the node 30. Nodes 30 may broadcast card identifier tables to other entities in the network, such as acquirer systems 16 and/or second switch 24, such that those other entities may update their routing tables (such as BIN tables) for the purposes of routing request messages in the system 100. A routing table may contain a mapping between access management node identifiers and card identifier ranges, for example.

Provisioning Process 200

[026] Figure 2 shows an exemplary provisioning process 200 for provisioning authentication credentials into a mobile device 12, such that the mobile device 12 can be used to access a service provided via access management node 30, for example a transit service or a financial service. The authentication credentials may be used by a digital wallet application executing on the mobile device 12, for example.

[027] For example, at step 202, a digital wallet application 418 (Figure 4) executing on a cardholder's mobile device 12 receives a cardholder's request to provision a smart card such as a payment card. The request includes a card identifier, for example, the PAN of a payment card. If the smart card is not a payment card, then instead of a PAN, another type of identifier for the smart card may be entered. The identifier may be transmitted in encrypted form.

[028] The digital wallet application 418 then generates and sends message data including the provisioning request to a digital wallet system 20. The message data may include the PAN (or other card identifier), an expiry date (e.g. in MM/YY format), and a card verification code (e.g., a 3-digit or 4-digit number that appears on the card and acts as an additional authentication code).

[029] At step 204, the digital wallet system 20 receives the message data and sends a digitization request including the message data to a first switch 22. The first switch 22 comprises one or more processors that route messages between the mobile device 12, digital wallet system 20 and access management node 30, and which may also modify message content in order to implement additional functionality at

mobile device 12, for example.

[030] At step 206, the first switch 22 receives the message data including the card identifier.

[031] At step 208, the first switch 22 may extract an access management system identifier from the card identifier. The first switch 22 makes a determination as to whether the access management system is registered in a routing table maintained at the first switch 22, for example in database 516 of the computing device 30 of Figure 5. For example, if the card identifier is a PAN, the first switch 22 may extract a Bank Identification Number (BIN) from the PAN and determine whether the BIN is registered in the routing table (also referred to as a BIN table in this context).

[032] If the BIN is not registered, then a decline message is sent to the digital wallet system at step 209a, which in turn sends a decline message to digital wallet application 418 for display at mobile device 12, at step 209b. Otherwise, at step 210, the first switch 22 may append a unique wallet identifier (which is an identifier for digital wallet providers such as ApplePay™ that subscribe to or operate a tokenization service) to the message data. The wallet identifier may be used to customize the provisioned credentials.

[033] At step 212, the first switch 22 forms a tokenization request message from the message data. The tokenization request message includes the card identifier and may also include other authentication credentials and the appended wallet ID from step 210. Typically, the tokenization request message also includes a device identifier for the mobile device 12. The tokenization request message, once formed, is transmitted by the first switch 22 to access management node 30. The tokenization request may be in the form of an ISO 8583 authorization request message (0100 message), for example. The request also includes a request for card personalization keys. The card personalization keys are to be used later for activating the service (e.g., a payment application) on the mobile device 12, and for dynamically generating encrypted messages, such as EMV cryptograms, that are used by mobile device 12 to access services via access management node 30.

[034] At step 214, the access management node 30 receives the tokenization request

message and checks if the request includes a wallet ID. If not, a decline message is sent to first switch 22, which routes it (step 217a) to digital wallet system 20, which in turn routes the decline message (step 217b) to the mobile device 12 for display to the user (at 209b). Otherwise, responsive to a determination that the request includes a wallet ID, step 216 is executed whereby the access management node 30 makes a determination as to whether the tokenization is to be approved. The determination as to whether the tokenization is to be approved may include checking the validity of the authentication credentials, such as confirming that the card identifier is a valid identifier (for example, by checking the format of the card identifier and/or that a checksum digit of the card identifier is valid), and/or that the card identifier is registered and is entitled to access the service that is managed by access management node 30.

[035] Access management node 30 may also analyze device data, such as a device identifier for mobile device 12, risk parameters of the device 12, and/or geolocation of device 12, in determining whether to approve tokenization. For example, access management node 30 may confirm that any device identifier transmitted as part of the tokenization request is a valid device identifier, and/or may confirm that the device 12 geolocation corresponds to a locale in which the smart card that is requested to be tokenized was issued.

[036] In response to a determination that the tokenization is approved, the access management node 30 generates, at step 218, card personalization data including the following:

- Card personalization keys for various types of authentication. For example, the access management node 30 may generate different keys for contactless, in-app, magnetic secure transmission (MST) and QR payments for a particular payment card, and/or for different types of service request. In particular, different card personalization keys may be generated for financial transaction requests and transit access requests. Typically, each different transaction type will have a different public/private key pair associated with it. Keys of each key pair may be used to dynamically generate encrypted messages for service requests, such as EMV cryptograms.
- A token number that is to be mapped to the card identifier and stored in a token vault at the access management node 30. The token number has a

format that identifies it as a token that is issued by that specific access management node 30. For example, in the case of payment cards, the token number has a BIN component (e.g. leading 6 digits) that identifies it as a token issued by the issuer (access management system) of the payment card.

- A card profile. The card profile specifies data, including risk parameters, that dictates the functional behavior of the mobile device 12 when the card is selected within the digital wallet application 418 and the mobile device 12 is used by the cardholder at a supported acceptance device such as a payment terminal, or a transit gate or other access point. The risk parameters may be used by the acceptance device in order to determine whether to transmit a service request from the mobile device 12 to the access management node 30, and/or whether to change the manner in which the service request is processed. For example, in the context of a payment request, one risk parameter may be a threshold transaction value below which a contactless payment request will be forwarded without requiring further authentication by the user of device 12.
- A card personalization script. This is executable code that, when executed at mobile device 12 (e.g. via digital wallet application 418), causes the mobile device 12 to download the card personalization keys, token number and card profile such that the card is provisioned into the mobile device 12 (e.g. into digital wallet application 418) such that the mobile device 12 is able to emulate the card when used at an acceptance device such as a contactless payment terminal or NFC-enabled transit gate. The card personalization script ensures that the card profile, card personalization keys and token are securely stored in the mobile device 12 (e.g., within an SE or TEE of the mobile device 12).

[037] In particular, the card personalization script is generated based on the type of the smart card, and in this regard, the access management node 30 may store configuration parameters for different smart card types or network types in its database 516. For example, if the smart card is a domestic debit card such as a Thai domestic debit card, the access management node 30 may retrieve, from database 516, a profile corresponding to the appropriate chip standard (for example, the Thai Chip Card or TCC Standard for Thai domestic debit cards) and configure the

card personalization script according to the retrieved profile. If the smart card is capable of multiple functionalities, a card personalization script may be generated for each function (e.g., one script for domestic debit payment functionality, one script for credit payment functionality, and one script for transit functionality).

[038] Upon generation of the card personalization data, an authorization approval message that includes the card personalization script as part of its payload is transmitted at step 220 to the first switch 22. The authorization approval message may be an ISO 8583 authorization response (0100) message, for example.

[039] At step 222, the first switch 22 receives the authorization approval message and card personalization script, and at step 224 configures, based on the wallet identifier, the card personalization script such that it is suitable for execution by the digital wallet 418. The card personalization script is then transmitted to digital wallet system 20, which in turn transmits (step 226) the script to mobile device 12. Mobile device 12 then executes the card personalization script (step 228). For example, the script may be executed within digital wallet application 418.

[040] Upon execution, the token number, card personalization keys and card profile are downloaded from access management node 30. For example, mobile device 12 may establish a secure connection directly with access management node 30 in order to download the card personalization data or may connect securely through the digital wallet system 20.

[041] The digital wallet application 418 stores the token and card personalization keys, for example in a secure element or other secure area of the mobile device 12, and uses the card profile data to write configuration parameters that dictate the behavior of the digital wallet application 418 when invoked in connection with the provisioned card and used at a point of interaction such as a payment terminal.

[042] At step 230, the digital wallet app 418 generates, on display 502 of the mobile device 12, "Request approved" or a similar indication that the onboarding process has been successful.

[043] The process 200 provisions a token to the mobile device 12 allowing a cardholder

to use the digital wallet application 418 executing on the mobile device 12 to effect requests for services using his or her smart card associated with the token.

Authorization Process 300

[044] An authorization process 300 enables a service request originating from a mobile computing device 12 to be authorized by the access management system 100 at an appropriate access management node 30. To make a service request using a tokenized card, which has been provisioned according to process 200, a cardholder may use a digital wallet application 418 executing on mobile device 12 to initiate the request. Advantageously, service requests may be transmitted from the mobile device 12 to the access management node 30 without going via the first switch 22, and without detokenizing the token.

[045] The service request may be a payment authorization request, for example. The payment authorization request may be initiated for an online transaction. For example, an in app payment may be initiated from the mobile device 12, wherein card transactions are performed using a merchant mobile app which interfaces with the digital wallet app 418 for payment purposes. In this example, a token associated with digital wallet account credentials is sent from the digital wallet app 418 to the merchant app, or through a web browser operating on the mobile computing device 12, and forwarded to a payment gateway (not shown) together with the merchant ID (MID).

[046] A payment authorization request may also be initiated at a point of interaction (POI) device 14, such as a payment terminal that is located in a retail store and in communicated with a point of sale (POS) system. Initiation of payment may occur, for example, after a basket has been generated at the POS system – e.g. one or more products have been scanned using a barcode reader and the basket thus comprises the one or more products.

[047] To initiate the payment authorization request, a contactless reader (typically, an NFC reader) of the POI device 14 communicates with the mobile device 12 via a contactless interface 420 of the mobile device 12 (Figure 4). At step 302, the mobile computing device 12 retrieves (from SE 416, for example) a token associated with

a user-selected smart card, such as a payment card, and transmits the token and transaction-specific data, such as an EMV cryptogram, to the POI device via the contactless interface 420.

[048] At step 304, the POI device 14 receives the token from the mobile computing device 12 and generates a transaction request message. The transaction request message includes merchant identifier (ID) and transaction-specific information, for example a transaction amount, and a transaction identifier such as a receipt number. The transaction-specific information may include crypto-specific information, such as an unpredictable number (in the sense envisaged by Book 2 of the EMV specification, the entire contents of which are incorporated herein by reference, for example). The unpredictable number may be, or be based on, the transaction identifier or a portion thereof. In a step not illustrated in Figure 3, and according to a known EMV cryptography process, the POI device 14 may transmit the unpredictable number, together with the other transaction-specific information, to the mobile device 12 such that the mobile device 12 can generate a transaction cryptogram, called an authorization request cryptogram (ARQC).

[049] The transaction request message may be sent directly to second switch 24, or may instead be sent to an intermediary such as an acquirer host system 16. The acquirer host system 16 may maintain an account on behalf of the merchant, and also on behalf of other merchants, thereby acting as an aggregator.

[050] At step 306, the acquirer host system 16 receives the transaction request message including the token, the transaction-specific information, and the transaction cryptogram. The acquirer host system 16 determines, based on the token, that the transaction should be switched via the second switch 24, and then sends the request message to the second switch 24, which receives the request message at step 308. For example, acquirer host system 16 may consult a routing table (BIN table) which maps token number ranges and/or BIN ranges to switches 24.

[051] At step 310, the second switch 24 (which may be a domestic switch, for example) identifies the access management node 30 that is associated with the transaction information, for example by comparing the BIN of the token number (e.g. the first

6 digits of the token number) against a BIN range to identify if the BIN corresponds with a token that is issued by one of the access management nodes 30. To this end, the second switch 24 may perform a lookup of the routing table (e.g., a BIN table) that has been periodically updated by the access management nodes 30 as noted above.

[052] Advantageously, once the appropriate access management node 30 has been identified, the second switch 24 may simply transmit the token and transaction information (including the cryptogram) to the access management node 30 for processing of the authorization request, without needing to detokenize the token. This is because, by virtue of the dissemination of token BIN ranges to acquirers 16 and/or the second switch 24, there is no need for the second switch 24 to map the token back to the actual authentication credential (such as a PAN in the case of a payment card) for routing purposes, and thus no need to transmit the token to the first switch 22. Additionally, as noted above, respective access management nodes 30 are able to validate transaction information (such as the authorization request cryptogram) for cards which they have onboarded via provisioning process 200.

[053] At step 312, the second switch 24 sends the authorization request comprising the token to the access management node 30. At step 314, the access management node 30 determines whether the authorization request is to be approved based on the transaction information. The access management node 30 may conduct a plurality of validation steps on the transaction information, including validating the authorization request cryptogram (for example, in accordance with known EMV cryptovalidation processes). In the context of a payment request, the access management node 30 may also confirm that the user's account is in good standing and has sufficient account balance, and/or analyze the merchant information in the authorization request to confirm that the merchant is allowed to accept transaction requests and that the token, or a payment account to which it is mapped, is allowed to be used for payments at that merchant (or merchant type).

[054] Responsive to a determination that the authorization request is approved, the access management node 30 sends a status message to the second switch 24 indicating status of the authorization request. If the transaction is declined, a decline message is transmitted to the second switch 24, and in turn, this is transmitted

(step 316) to the acquirer host system 16, which transmits it to the POI device 14 (step 318). The POI device 14 displays the decline message (step 320), as does the mobile device 12 (step 322). If the transaction is approved, the second switch 24 receives transaction approval from access management node 30 at step 324. The second switch 324 sends the transaction approval message to the acquirer 16, at step 330. The acquirer transmits the transaction approval to the POI device 14 at step 332.

[055] At step 326, the mobile device 12 receives message data indicating transaction approval. At step 328, the mobile device 12 generates a transaction approval message on display 402 to evidence approval of the transaction – e.g. "Transaction approved".

[056] At step 332, the POI device 14 receives message data indicating transaction approval. At step 334, the merchant terminal 14 displays a similar transaction approval message on display 402.

Mobile Computing Device 12

[057] Figure 4 is a block diagram showing an exemplary mobile computing device 12 in which embodiments of the invention may be practiced. The mobile computer device 12 may be a mobile computer device such as a smart phone, a personal data assistant (**PDA**), a palm-top computer, and multimedia Internet enabled cellular telephones. For ease of description, the mobile computer device 12 is described below, by way of non-limiting example, with reference to a mobile device in the form of an iPhone™ manufactured by Apple™, Inc., or one manufactured by LG™, HTC™ and Samsung™, for example.

[058] As shown, the mobile computer device 12 includes the following components in electronic communication via a bus 406:

- (h) a display 402;
- (i) non-volatile (non-transitory) memory 404;
- (j) random access memory ("RAM") 408;
- (k) N processing components 410;

- (l) a transceiver component 412 that includes N transceivers;
- (m) user controls 414;
- (n) a secure element (SE) 416; and
- (o) a NFC controller 420.

[059] Although the components depicted in Figure 4 represent physical components, Figure 4 is not intended to be a hardware diagram. Thus, many of the components depicted in Figure 4 may be realized by common constructs or distributed among additional physical components. Moreover, it is certainly contemplated that other existing and yet-to-be developed physical components and architectures may be utilized to implement the functional components described with reference to Figure 4.

[060] The display 402 generally operates to provide a presentation of content to a user, and may be realized by any of a variety of displays (e.g., CRT, LCD, HDMI, micro-projector and OLED displays).

[061] In general, the non-volatile data storage 404 (also referred to as non-volatile memory) functions to store (e.g., persistently store) data and executable code.

[062] In some embodiments for example, the non-volatile memory 404 includes bootloader code, modem software, operating system code, file system code, and code to facilitate the implementation components, well known to those of ordinary skill in the art, which are not depicted nor described for simplicity.

[063] In many implementations, the non-volatile memory 404 is realized by flash memory (e.g., NAND or ONENAND memory), but it is certainly contemplated that other memory types may be utilized as well. Although it may be possible to execute the code from the non-volatile memory 404, the executable code in the non-volatile memory 404 is typically loaded into RAM 408 and executed by one or more of the N processing components 410.

[064] The N processing components 410 in connection with RAM 408 generally operate to execute the instructions stored in non-volatile memory 404. As one of ordinary skill in the art will appreciate, the N processing components 410 may include a video processor, modem processor, DSP, graphics processing unit (GPU), and other

processing components.

[065] The transceiver component 412 includes N transceiver chains, which may be used for communicating with external devices via wireless networks. Each of the N transceiver chains may represent a transceiver associated with a particular communication scheme. For example, each transceiver may correspond to protocols that are specific to local area networks, cellular networks (e.g., a CDMA network, a GPRS network, a UMTS networks), and other types of communication networks.

[066] The mobile computer device 12 can execute mobile applications. The digital wallet application 418 could be a mobile application, web page application, or computer application. The digital wallet application 418 may be accessed by a computing device such as mobile computer device 12, computing device 30, or a wearable device such as a smartwatch. The digital wallet application 418 can be used with payment cards including credit cards, debit cards, store cards, etc.

[067] The NFC controller 420, in conjunction with the secure element 416 and/or with other hardware components (e.g. one or more of the N processors 410) may be adapted to effect contactless payments. To this end, the SE 416 may emulate the functionality of a standard physical payment or non-payment smart card as described above. Sensitive (e.g. payment-related) information is stored in encrypted form on the SE 416, and it is only addressable by authorized software or hardware components, such as digital wallet application 418, in order to effect EMV transactions. The digital wallet application 418 may itself be stored in the SE 416. Alternatively, in an embodiment which is not illustrated, one or more processors 410 of mobile device 12 may include a trusted execution environment (TEE) in which the digital wallet application 418 executes and/or in which sensitive data such as payment-related data is stored. Some digital wallet applications 418 may alternatively make use of host card emulation (HCE), in which payment functionality is enabled by standard hardware components of the mobile device 12, but sensitive authentication credentials such as the PAN are stored in a remote secure server, i.e. the secure element effectively resides in the cloud. In some cases, HCE can co-exist with TEE or even with SE implementations of digital wallets.

[068] It should be recognized that Figure 4 is merely exemplary and in one or more

exemplary embodiments, the functions described herein may be implemented in hardware, software, firmware, or any combination thereof. If implemented in software, the functions may be transmitted or stored as one or more instructions or code encoded on a non-transitory computer-readable medium 404. Non-transitory computer-readable medium 404 includes both computer storage medium and communication medium including any medium that facilitates transfer of a computer program from one place to another. A storage medium may be any available medium that can be accessed by a computer.

Access management node 30

[069] Figure 5 shows an example computing device 30 that is capable of implementing an access management node of the system 100. In some embodiments, multiple computing devices 30 may be considered to be a single access management node.

[070] The components of the computing device 30 can be configured in a variety of ways. The components can be implemented entirely by software to be executed on standard computer server hardware, which may comprise one hardware unit or different computer hardware units distributed over various locations, which may communicate over a network. A number of the components or parts thereof may also be implemented by application specific integrated circuits (ASICs) or field programmable gate arrays.

[071] In the example shown in Figure 5, the computing device 30 is a commercially available server computer system based on a 32 bit or a 64 bit Intel architecture, and the processes and/or methods executed or performed by the computing device 30 are implemented in the form of programming instructions of one or more software components or modules 522 stored on non-volatile (e.g., hard disk) computer-readable storage 524 associated with the computing device 30. At least parts of the software modules 522 could alternatively be implemented as one or more dedicated hardware components, such as application-specific integrated circuits (ASICs) and/or field programmable gate arrays (FPGAs).

[072] The computing device 30 includes at least one or more of the following standard, commercially available, computer components, all interconnected by a bus 535:

- (p) random access memory (RAM) 526;
- (q) at least one computer processor 528, and
- (r) external computer interfaces 530:
 - (i) universal serial bus (USB) interfaces 530a (at least one of which is connected to one or more user-interface devices, such as a keyboard, a pointing device (e.g., a mouse 532 or touchpad),
 - (ii) a network interface connector (NIC) 530b which connects the computer device 30 to a data communications network and/or to external devices such as a hardware security module (HSM) 550; and
 - (iii) a display adapter 530c, which is connected to a display device 534 such as a liquid-crystal display (LCD) panel device.

[073] The computing device 30 includes a plurality of standard software modules, including:

- (s) an operating system (OS) 536 (e.g., Linux or Microsoft Windows); and
- (t) structured query language (SQL) modules 542 (e.g., MySQL, available from <http://www.mysql.com>), which allow data to be stored in and retrieved/accessed from an SQL database 516.

[074] Advantageously, the database 516 forms part of the computer readable data storage 524. Alternatively, the database 516 is located remote from the server 30 shown in Figure 5.

[075] The boundaries between the modules and components in the software modules 522 are exemplary, and alternative embodiments may merge modules or impose an alternative decomposition of functionality of modules. For example, the modules discussed herein may be decomposed into submodules to be executed as multiple computer processes, and, optionally, on multiple computers. Moreover, alternative embodiments may combine multiple instances of a particular module or submodule. Furthermore, the operations may be combined or the functionality of the operations may be distributed in additional operations in accordance with the invention. Alternatively, such actions may be embodied in the structure of circuitry that

implements such functionality, such as the micro-code of a complex instruction set computer (CISC), firmware programmed into programmable or erasable/programmable devices, the configuration of a field-programmable gate array (FPGA), the design of a gate array or full-custom application-specific integrated circuit (ASIC), or the like.

[076] Each of the blocks of the flow diagrams of the processes of the computing device 30 may be executed by a module (of software modules 522) or a portion of a module. The processes may be embodied in a non-transient machine-readable and/or computer-readable medium for configuring a computer system to execute the method. The software modules may be stored within and/or transmitted to a computer system memory to configure the computer system to perform the functions of the module.

[077] The computing device 30 normally processes information according to a program (a list of internally stored instructions such as a particular application program and/or an operating system) and produces resultant output information via input/output (I/O) devices 530. A computer process typically includes an executing (running) program or portion of a program, current program values and state information, and the resources used by the operating system to manage the execution of the process. A parent process may spawn other, child processes to help perform the overall functionality of the parent process. Because the parent process specifically spawns the child processes to perform a portion of the overall functionality of the parent process, the functions performed by child processes (and grandchild processes, etc.) may sometimes be described as being performed by the parent process.

[078] Software modules 522 may comprise a key management module and a card personalization module. The card personalization module may generate card personalization scripts in accordance with stored card type or network type profiles that are stored in the database 516, as noted above. The card personalization module may also generate card personalization keys, for example by invoking the key management module. The key management module may communicate with HSM 550, for example.

[079] In a conventional four party payment system, a payment authorization request for a credit card transaction is requested by a POI terminal 14, for example, and sent to the acquirer 16, which is forwarded to a payment network server 22 and finally routed to an issuer 30 for authorization.

[080] For a domestic switch payment system, the transaction authorization request is received by the acquirer system 14, which routes the authorization request to the domestic switch 24 and to the issuer system 30 in the manner described above. The authorization request may be formatted according to the ISO 8583 standard, for example, and may include a token number that replaces a primary account number (PAN) of the payment instrument being used for the transaction (e.g. according to some embodiments above), a merchant identifier (MID), and an amount of the transaction. Other transaction-related information may be included in the ISO 8583 message, as will be appreciated by those skilled in the art. The issuer system 30 receives the request, applies authorization logic to approve or decline the request, for example by comparing the transaction amount with a transaction limit associated with the cardholder, and sends an authorization response (approve or decline, optionally with a code indicating the reason for the decline) back to the domestic switch 24.

[081] In some embodiments, the payment authorization request may be received via the issuer system 30, which approves or declines the request (which again may be in ISO 8583 format, and comprise a token, MID, transaction amount etc.) and sends a response directly back to the requestor, e.g. the POI terminal 14.

[082] The system 100 may also process a pre-authorization (or “pre-auth”) request, in which funds are not transferred on approval of the request, but are instead placed on hold. The pre-auth can later be completed, for example by the merchant server, in order to release the funds. Alternatively, the pre-auth can be cancelled, thus effectively cancelling the transaction.

[083] Embodiments enable tokens to be used in a domestic switch payment system thereby allowing digital wallets to be used to initiate a payment, without requiring the domestic switch to have card provisioning (digitization) capability, nor needing to detokenize a token in order to process a transaction that uses the token.

POI Device 14

[084] The POI device 14 may be, for example, a payment card terminal for interfacing with payment cards or mobile device 12 to make electronic funds transfers. The POI device 14 interfaces with payment cards or mobile device 12 by means of a reader such as a magnetic swipe reader, EMV chip reader or near field communication (NFC) contactless communication.

[085] The POI device 14 may be embodied by a further mobile computing device such as the one described in Figure 4. NFC controller 420 may, in such embodiments, serve as the means by which mobile device 12 interacts with POI device 14. Alternatively, an external interface reader (e.g., an NFC dongle with EMV capability) may be connected to the further mobile computing device to form a POI device 14.

[086] The POI device 14 may be in communication with a merchant's point-of-sale system. In some embodiments, such as in a transit setting, the POI device 14 may include a contactless interface that is in communication with an actuator of a physical access barrier, such as a transit gate or other transit access point. In such embodiments, a successful authorization response message transmitted by an access management node 30 to the POI device 14 may cause the actuator to open the access barrier.

[087] Throughout this specification, unless the context requires otherwise, the word "comprise", and variations such as "comprises" and "comprising", will be understood to imply the inclusion of a stated integer or step or group of integers or steps but not the exclusion of any other integer or step or group of integers or steps.

[088] The reference to any prior art in this specification is not, and should not be taken as, an acknowledgment or any form of suggestion that the prior art forms part of the common general knowledge.

Claims Defining the Invention

1. A method for provisioning authentication credentials on a mobile device, comprising:
 - receiving a provisioning request message comprising one or more authentication credentials of a smart card, including at least an identifier of the smart card, and a request for card personalization keys;
 - determining a type of the smart card based on the identifier of the smart card;
 - generating, based on the type of the smart card, a card personalization script comprising computer-readable instructions for configuring the mobile device to emulate the smart card;
 - generating a token based on the authentication credentials or part thereof;
 - generating one or more card personalization keys, respective card personalization keys being for different respective authentication types;
 - and transmitting, to the mobile device, the card personalization script for execution by the mobile device;
 - wherein the card personalization script comprises computer-readable instructions to cause the mobile device to retrieve the one or more card personalization keys and the token to thereby enable emulation of the smart card by the mobile device.
2. A method according to claim 1, wherein the provisioning request message comprises a digital wallet identifier, and wherein the method further comprises configuring the card personalization script based on a wallet type derived from the digital wallet identifier.
3. A method according to claim 1 or claim 2, further comprising generating card profile data indicative of one or more restrictions on usage of the authentication credentials, wherein the card personalization script comprises computer-readable instructions to further cause the mobile device to retrieve the card profile data.
4. A method according to any one of the preceding claims, wherein the smart card is a payment card.

5. A method according to claim 4, wherein the provisioning request message is an ISO8583 authorization request message.
6. A method according to any one of claims 1 to 5, wherein the provisioning request message includes device data characterizing the mobile device, and wherein the method further includes validating the device data.
7. A method according to claim 6, wherein the device data includes an identifier of the mobile device.
8. A method according to any one of claims 1 to 7, wherein the card personalization script comprises computer-readable instructions to cause the mobile device to store the one or more card personalization keys and the token in a secure hardware component of the mobile device.
9. A system for provisioning authentication credentials on a mobile device, comprising an access management node and a first network switch, the access management node being configured to:
 - receive, from the first network switch, a provisioning request message comprising: one or more authentication credentials of a smart card, including at least an identifier of the smart card, and a request for card personalization keys;
 - determine a type of the smart card based on the identifier of the smart card;
 - generate, based on the type of the smart card, a card personalization script comprising computer-readable instructions for configuring the mobile device to emulate the smart card;
 - generate a token based on the authentication credentials, or part thereof;
 - generate one or more card personalization keys, respective card personalization keys being for different respective authentication types;
 - and transmit, to the mobile device via the first network switch, the card personalization script for execution by the mobile device;
 - wherein the card personalization script, when executed, causes the mobile device to retrieve the one or more card personalization keys and the token to thereby enable emulation of the smart card by the mobile device, such

that the mobile device is configured to transmit the token to the access management node as part of an access request, via a second network switch, without the second network switch detokenizing the token.

10. A system according to claim 9, wherein the provisioning request message comprises a digital wallet identifier, and wherein the access management node is further configured to configure the card personalization script based on a wallet type derived from the digital wallet identifier.

11. A system according to claim 9 or claim 10, wherein the access management node is further configured to generate card profile data indicative of one or more restrictions on usage of the authentication credentials, and wherein the card personalization script comprises computer-readable instructions to further cause the mobile device to retrieve the card profile data.

12. A system according to any one of claims 9 to 11, wherein the smart card is a payment card.

13. A system according to claim 12, wherein the provisioning request message is an ISO8583 authorization request message.

14. A system according to any one of claims 9 to 11, wherein the provisioning request message includes device data characterizing the mobile device, and wherein the access management node is further configured to validate the device data.

15. A system according to claim 14, wherein the device data includes an identifier of the mobile device.

16. A system according to any one of claims 12 to 15, wherein the card personalization script comprises computer-readable instructions to cause the mobile device to store the one or more card personalization keys and the token in a secure hardware component of the mobile device.