METHOD FOR APPLYING FOR PAYMENT TOKEN, APPARATUS, SYSTEM, AND SERVER

TECHNICAL FIELD

[0001] The present disclosure relates to the technical field of payment security, more particularly to a method of applying for a payment token, a device, a system, and a server.

BACKGROUND

[0002] With the rapid development of Internet, payment methods for consumers have also undergone tremendous changes, and have gradually evolved from bank card payments on POS machines to payments with QR codes, mobile applications (APPs), IoT devices or in other channels. The payment-sensitive information has also evolved from simple card numbers, etc. to become complex and diverse, thus facing more serious security risks than before in terms of the transmission method, storage location, encryption means, and the like. Without a secure payment security measures, the consumers' payment behavior would face unpredictable risks. In addition, users have tended to use their cell phones for purchases and payments more and more frequently. Currently, major banks, third-party payment providers and others have launched their own wallet applications (e.g., Cloud QuickPass®) that require the users to bind their bank cards to respective wallet applications for payments with the wallet applications. In some scenarios, the users also bind their bank cards to some shopping applications. However, the process of binding bank cards in various applications increases the risk of the bank card information leakage.

SUMMARY

[0003] An object of the present disclosure is to provide a method of applying for a payment token, a device, a system, and a server regarding the deficiencies of the prior art.

[0004] In order to solve the aforesaid technical problem, the present disclosure adopts the following technical solution.

[0005] Embodiments of the present disclosure provide a method of applying for a payment token applicable to a second application module at a terminal device. The method includes: displaying bound payment accounts upon receiving an invocation

instruction from a first application module, for a user to select one from the bound payment accounts, where the invocation instruction carries merchant information of the first application module; and sending to a server an authorization request for token binding, for the first application module to apply to the server for a payment token based on a returned authorization code for token binding, where the authorization request for token binding carries the merchant information of the first application module and the payment account selected by the user.

[0006] Optionally, the authorization request for token binding further carries transaction control parameters.

[0007] Optionally, the transaction control parameters includes at least one selected from a validity period of the payment token, a transaction volume limit per day of the payment token and the maximum number of transactions per day of the payment token.

[0008] Optionally, the method further includes: providing a setting interface for the user to set the transaction control parameters.

[0009] Optionally, the method further includes: receiving the authorization code for token binding from the server and forwarding the received authorization code for token binding to the first application module.

[0010] Optionally, information for confirmation prompt on the payment account and/or on the user's identification is further sent to the first application module while forwarding the received authorization code for token binding to the first application module.

[0011] Embodiments of the present disclosure provide a method of applying for a payment token applicable to a first application module at a terminal device. The method includes: invoking a second application module, and transmitting merchant information of the first application module to the second application module, for the second application module to send to a server an authorization request for token binding, where the authorization request for token binding carries the merchant information of the first application module and a payment account selected by a user from the second application module, the first application module is a software for invoking the second application module, and the payment accounts selectable by the user are bound beforehand by the user to the second application module; receiving an authorization code for token binding generated by the server in response to the authorization request for token binding; sending to the server a payment token request carrying the authorization code for token binding; and receiving a payment token from the server.

[0012] Optionally, the authorization code for token binding is forwarded by the second

application module from the server to the first application module; information for confirmation prompt on the payment account and/or on the user's identification is further received from the second application module while receiving the authorization code for token binding from the second application module; and the method further includes: displaying the information for confirmation prompt on the payment account and/or on the user's identification to prompt the user to confirm whether the information for confirmation prompt is correct, and sending the payment token request to the server if the user confirms that the information for confirmation prompt is correct.

[0013] Optionally, the method further includes: selecting the second application module, which includes sending a query request to the server to query application modules supported by the server, and displaying a list of the queried application modules to allow the user to select an application module therefrom as the second application module.

[0014] Optionally, an encrypted authentication code and the merchant information of the first application module are further sent to the server if the second application module is invoked, where the encrypted authentication code is a ciphertext of the merchant information of the first application module as acquired by the first application module upon registering with the server.

[0015] Optionally, the method further includes: receiving from the server first encrypted data, a second secret key and clock information, where the first encrypted data is a ciphertext acquired by encrypting the authorization code for token binding with the first secret key; synchronizing a local clock according to the received clock information, and acquiring second encrypted data by encrypting the merchant information with the second secret key and the synchronized clock; and further sending the second encrypted data and the first encrypted data to the server if the second application module is invoked.

[0016] Embodiments of the present disclosure provide a method of applying for a payment token applicable to a server. The method includes: receiving from a second application module an authorization request for token binding, where the authorization request for token binding carries merchant information of a first application module and a payment account selected by a user from bound payment accounts of the second application module; generating an authorization code for token binding if validation of the authorization request for token binding passes, and sending the authorization code for token binding to the first application module; receiving from the first application module a payment token request carrying the authorization code for token binding; and generating a payment token if validation of the payment token request passes, and

sending the payment token to the first application module.

[0017] Optionally, the authorization request for token binding further carries merchant information of the first application module; and validation of the authorization request for token binding passes if the merchant information carried by the authorization request for token binding is information of a merchant registered with the server.

[0018] Optionally, the method further includes: providing a registration service to the first application module; encrypting the merchant information of the first application module to acquire an encrypted authentication code; sending the encrypted authentication code to the first application module; receiving from the first application module the encrypted authentication code and the merchant information; and decrypting the encrypted authentication code, comparing a resulting plaintext to the received merchant information, and determining that validation of the authorization request for token binding passes if the plaintext and the merchant information are consistent.

[0019] Optionally, the payment token request further carries merchant information of the first application module; and validation of the payment token request includes at least one selected from following three validation methods: determining that the validation passes if the authorization code for token binding exists; determining that the validation passes if the merchant information is consistent with that carried by the authorization request for token binding; and determining that the validation passes if the authorization code for token binding is within a validity period.

[0020] Optionally, first encrypted data, a second secret key and clock information are further sent to the first application module while generating the authorization code for token binding, where the first encrypted data is a ciphertext acquired by encrypting the authorization code for token binding with the first secret key; and

the method further includes:

receiving from the first application module the second encrypted data and the first encrypted data; acquiring a third encrypted data by encrypting the merchant information of the first application module with the second secret key and the clock information, determining whether the third encrypted data is consistent with the second encrypted data received from the first application module, determining whether a result acquired by decrypting the received first encrypted data with the first secret key is consistent with the authorization code for token binding, and if both determination results are consistent, determining that validation of the payment token request passes.

[0021] Optionally, sending the authorization code for token binding to the first

application module includes: sending the authorization code for token binding to the first application module by forwarding via the second application module.

[0022] Embodiments of the present disclosure provide a method of applying for a payment token. The method includes: at a terminal device, invoking by a first application module a second application module, and transmitting merchant information of the first application module to the second application module; at the terminal device, displaying by the second application module bound payment accounts in the second application module upon receiving an invocation instruction from the first application module, for a user to select one from the bound payment accounts, and sending an authorization request for token binding to a server, where the authorization request for token binding carries the merchant information of the first application module and the payment account selected by the user; generating by the server an authorization code for token binding if validation of the authorization request for token binding passes, and sending the authorization code for token binding to the first application module; at the terminal device, sending by the first application module a payment token request to the server, where the payment token request carries the authorization code for token binding; and generating by the server a payment token if validation of the payment token request passes, and sending the payment token to the first application module.

[0023] Embodiments of the present disclosure provide a terminal device, including a first memory and a first processor, the first memory having a first instruction and/or a second instruction stored therein, where the first instruction, when executed by the first processor, causes the aforesaid method of applying for a payment token applicable to the first application module to be carried out, and the second instruction, when executed by the first processor, causes the aforesaid method of applying for a payment token applicable to the second application module to be carried out.

[0024] Embodiments of the present disclosure provide a server, including a second memory and a second processor, the second memory having a payment token managing program stored therein, where the payment token managing program, when executed by the second processor, causes the aforesaid method of applying for a payment token applicable to the server to be carried out.

[0025] Embodiments of the present disclosure provide a system for applying for a payment token, which includes the aforesaid terminal device and the aforesaid server.

[0026] Compared with the prior art, the present disclosure has following beneficial effects. The user only needs to bind more than one payment account (e.g., bank card)

to the second application module, and if he or she needs to bind a payment account to the first application module, the operation may be made via the second application module to which the payment account has been bound, such that the first application module has no access to the bank card number or other payment-sensitive information, thus the security of the payment account binding process can be enhanced.

BRIEF DESCRIPTION OF THE DRAWINGS

[0027] FIG. 1 is a flow chart of a method of applying for a payment token according to an embodiment of the present disclosure;

[0028] FIG. 2 is a flow chart of a method of applying for a payment token according to another embodiment of the present disclosure;

[0029] FIG. 3 is a flow chart of a method of applying for a payment token according to still another embodiment of the present disclosure;

[0030] FIG. 4 is a flow chart of a method of applying for a payment token according to yet another embodiment of the present disclosure;

[0031] FIG. 5 is an interaction flow chart of a method of applying for a payment token according to yet another embodiment of the present disclosure;

[0032] FIG. 6 is a block structural diagram of a system for applying for a payment token according to an embodiment of the present disclosure.

DETAILED DESCRIPTION

[0033] In the present disclosure, it should be understood that terms such as "include" or "have", etc. are intended to indicate the existence of the characteristics, figures, steps, actions, components and parts as disclosed by the specification or combinations thereof, without excluding the existence of one or more other characteristics, figures, steps, actions, components, parts or combinations thereof.

[0034] Furthermore, it should be noted that, in the case of no conflict, the embodiments of the present disclosure and features of the embodiments may be combined with each other in any manner. The present disclosure will be described in detail below with reference to the accompanying drawings and embodiments.

[0035] In embodiments of the present disclosure, the terminal device may be a device operatable by a terminal user, such as a smart phone, a tablet computer, or a personal computer. The second application module may be an application module for payment, such as Cloud QuickPass®, mobile banks, and a shopping APP with payment function, and may of course also be respective HTML5 pages of these application modules. The

first application module may refer to a software to which a payment account (for example, a bank card) needs to be bound, and for example may also be a Cloud QuickPass® application, a shopping APP with payment function, or their respective HTML5 pages. The first application module and the second application module are distinguished by sequence numbers merely in that in the application scenario the second application module has a payment account bound thereto, while the first application module needs to apply for a payment token via the assistance of the second application module. The first application module and the second application module may be implemented entirely by software, entirely by hardware, or a combination of software and hardware. The server provides a service for managing the payment token. The payment token, also known as a token, is a piece of code that replaces the bank card number or other payment-sensitive information.

[0036] The present disclosure will be further described in detail below in conjunction with embodiments shown by the accompanying drawings.

[0037] Referring to FIG. 1, the method of applying for a payment token according to embodiments of the present disclosure is applicable to a second application module of a terminal device. From the perspective of software, the execution subject is the software that has a payment account bound thereto, such as a Cloud QuickPass® application, or a HTML5 page of the Cloud QuickPass®. From the perspective of hardware, the execution subject may be a terminal device running such software or a server providing HTML5 page services to the terminal device. The method includes following steps.

[0038] S101: bound payment accounts are displayed upon receiving an invocation instruction from a first application module, for a user to select one from the bound payment accounts. The invocation instruction carries merchant information of the first application module.

[0039] The merchant information may be, for example, a merchant number or a token requester number, and is a unique identifier of the first application module, for the server side.

[0040] S102: an authorization request for token binding is sent to a server, for the first application module to apply to the server for a payment token based on an authorization code for token binding returned by the server. The authorization request for token binding carries the merchant information of the first application module and the payment account selected by the user.

[0041] For example, in terms of displaying the bound payment accounts, the second application module may display the transaction control parameters such as partial card

numbers of the bound bank card, or the transaction volume limit or the number of transactions per day of the bound bank cards. The user may select a payment account via the second application module as the payment account applied by the first application module for binding.

[0042] The authorization code for token binding applied by the second application module is a string of codes and serves as a credential for the communication between the first application module and the server, and the authorization code for token binding per se has no specific meaning.

[0043] The server may send the authorization code for token binding directly to the first application module, or may at first send to the second application module, after which the second application module forwards the received authorization code for token binding to the first application module. Subsequently, the first application module may apply for a final payment token from the server with the authorization code for token binding.

[0044] The user only needs to bind the payment account to the second application module, and the first application module may acquire the authorization code for token binding from the server via the second application module and then relies on the authorization code for token binding to acquire the payment token from the server. If the first application module needs to have a payment account bound thereto, the first application module has no need to access the payment account to be bound during the overall payment account binding process, thereby reducing the chance of exposure of the payment account and enhancing security.

[0045] Based on the method of applying for a payment token of FIG. 1, some embodiments of the present disclosure further provide some specific implementation solutions and extension solutions of the method of applying for a payment token, which will be described below.

[0046] Optionally, the authorization request for token binding further carries transaction control parameters. The transaction control parameters may for example comprise at least one selected from a validity period of the payment token, a transaction volume limit per day of the payment token and the maximum number of transactions per day of the payment token.

[0047] The transaction control parameters may be, by default, those of the payment account selected in the second application module. Alternatively, the second application module at the terminal device may further provide a setting interface for the user to set the transaction control parameters.

[0048] Optionally, information for confirmation prompt on the payment account and/or

on the user's identification is further sent to the first application module while forwarding the received authorization code for token binding to the first application module.

[0049] As a result, the first application module at the terminal device can display the information on the payment account and/or on the user's identification to prompt the user to perform confirmation.

[0050] Based on the similar inventive concept, with reference to FIG. 2, embodiments of the present disclosure provide a method of applying for a payment token applicable to the first application module of the terminal device. From the perspective of software, the execution subject is the software to which a payment account is desired to be bound (referred to as a first application module in this specification), such as a Cloud QuickPass® application, or a HTML5 page of the Cloud QuickPass®. From the perspective of hardware, the execution subject may be a terminal device running such software or a server providing HTML5 page services to the terminal device. The method includes following steps.

[0051] S201: the second application module is invoked and merchant information of the first application module is transmitted to the second application module, for the second application module to send an authorization request for token binding to a server. The authorization request for token binding carries the merchant information of the first application module and a payment account selected by a user from the second application module, the first application module is a software for invoking the second application module, and the payment accounts selectable by the user are bound beforehand by the user to the second application module.

[0052] S202: an authorization code for token binding generated by the server in response to the authorization request for token binding is received. The authorization code for token binding may be received from the server or received by being forwarded by the second application module.

[0053] S203: the server a payment token request carrying the authorization code for token binding is sent.

[0054] S204: a payment token is received from the server.

[0055] The first application module applies to the server for an authorization code for token binding via the second application module, and then applies to the server for a payment token by virtue of the authorization code for token binding. Thus, the first application module has no need to access sensitive information such as the payment account during the overall process, thereby improving the security.

[0056] Based on the method of applying for a payment token of FIG. 2, some

embodiments of the present disclosure further provide some specific implementation solutions and extension solutions of the method of applying for a payment token, which will be described below.

[0057] Optionally, the authorization code for token binding is forwarded by the second application module from the server to the first application module; information for confirmation prompt on the payment account and/or on the user's identification is further received from the second application module while receiving the authorization code for token binding from the second application module; and the method further includes: displaying the information on the payment account and/or on the user's identification to prompt the user to confirm whether the information for confirmation prompt is correct, and sending the payment token request to the server if the user confirms that the information for confirmation prompt is correct.

[0058] The information for confirmation prompt on the payment account may for example be last four digits of the bank card number, the type of the bank card, the name of the bank, and the like. The information for confirmation prompt on the user's identity information may for example be the user's ID number with some digits removed, the user's cell phone number registered for the bank card, and the like. More information for confirmation prompt may be provided for the user to confirm.

[0059] Thus, the user can confirm the payment account to be bound and the personal information in the first application module.

[0060] Optionally, the method further includes: selecting the second application module, which includes sending a query request to the server to query application modules supported by the server, and displaying a list of the queried application modules to allow the user to select an application module therefrom as the second application module.

[0061] That is, the user may select the specific software as a relay for the first application module to apply for the authorization code for token binding.

[0062] In addition, the first application module may further provide a display interface to show the relevant information of the payment account that has been bound (e.g. the issuing bank, the account type, the last number of the corresponding payment account, the validity of the payment token, the transaction volume limit, etc.).

[0063] The first application module (which has the card already bound thereto) may further provide the function of unbinding, and may simply send to the server a message informing of a token status change according to the unbinding operation of the user.

[0064] If the second application module performs operation on the payment account bound to the first application module, the first application module may also receive the

corresponding notification from the server, such that the first application module can be informed of the latest status information of the payment token from the notification.

[0065] Optionally, an encrypted authentication code and the merchant information of the first application module are further sent to the server if the second application module is invoked, where the encrypted authentication code is a ciphertext of the merchant information of the first application module as acquired by the first application module upon registering with the server.

[0066] The first application module may acquire an encrypted authentication code while registering with the server, where the encrypted authentication code is an encrypted ciphertext of the merchant information of the first application module. The first application module may further send the encrypted authentication code and the merchant information while sending the authorization request for token binding. Then, the server validates the authorization request for token binding accordingly.

[0067] Thus, certain phishing-type fraudulent APPs can be prevented from counterfeiting the relevant merchant information of the first application module, whereby users are prevented from binding their bank cards to the phishing-type APPs and protected from property loses.

[0068] Optionally, the method further includes: receiving from the server first encrypted data, a second secret key and clock information, the first encrypted data being a ciphertext acquired by encrypting the authorization code for token binding with the first secret key; synchronizing a local clock according to the received clock information, and acquiring second encrypted data by encrypting the merchant information with the second secret key and the synchronized local clock; and sending the second encrypted data and the first encrypted data to the server if the second application module is invoked.

[0069] By getting the clock information involved, the application for the payment token by the first application module may be cancelled if there is no response from the server for a certain period of time, thereby avoiding the user to wait for a long time.

[0070] Based on the similar inventive concept, with reference to FIG. 3, embodiments of the present disclosure further provide a method of applying for a payment token to be applicable to the server. From the perspective of software, the execution subject is the software run by the server, and from the perspective of hardware, the execution subject is the server. The method includes following steps.

[0071] S301: an authorization request for token binding is received from a second application module. The authorization request for token binding carries merchant information of a first application module, and a payment account selected by a user from

bound payment accounts of the second application module.

[0072] S302: an authorization code for token binding is generated if validation of the authorization request for token binding passes, and the authorization code for token binding is sent to the first application module.

[0073] S303: a payment token request carrying the authorization code for token binding is received from the first application module.

[0074] S304: a payment token is generated if validation of the payment token request passes, and the payment token is sent to the first application module.

[0075] The server generates an authorization code for token binding for the first application module based on the interaction with the second application module, and then the first application module applies to the server for a payment token by virtue of the authorization code for token binding. Thus, the first application module has no need to access the sensitive information such as payment accounts during the overall process, which improves the security.

[0076] Based on the method of applying for a payment token of FIG. 3, some embodiments of the present disclosure further provide some specific implementation solutions and extension solutions of the method of applying for a payment token, which will be described below.

[0077] Optionally, the payment token request further carries merchant information of the first application module; and validation of the authorization request for token binding passes if the merchant information carried by the authorization request for token binding is information of a merchant registered with the server.

[0078] Optionally, the method further includes: providing a registration service to the first application module; encrypting the merchant information of the first application module to acquire an encrypted authentication code; sending the encrypted authentication code to the first application module; receiving from the first application module the encrypted authentication code and the merchant information; and decrypting the encrypted authentication code, comparing a resulting plaintext to the received merchant information, and determining that validation of the authorization request for token binding passes if the plaintext and the merchant information are consistent.

[0079] That is, when the first application module registers with the server, the server encrypts the merchant information of the first application module and sends the ciphertext (i.e., the encrypted authentication code) to the first application module. The first application module shall provide the correct encrypted authentication code and the correct merchant information to acquire the authorization code for token binding.

[0080] Thus, some phishing-type fraudulent APPs can be prevented from counterfeiting the relevant merchant information of the merchant application, and the users can thereby be prevented from binding their bank cards to the phishing-type APPs and protected from property loses.

[0081] The validation of the payment token request may be implemented by one of the following manners.

[0082] Optionally, the payment token request further carries merchant information of the first application module; and validation of the payment token request includes at least one selected from following three validation methods: determining that the validation passes if the authorization code for token binding exists; determining that the validation passes if the merchant information is consistent with that carried by the authorization request for token binding; and determining that the validation passes if the authorization code for token binding is within a validity period.

[0083] For example, the server may determine whether the authorization code for token binding carried in the payment token request exists in the server; if so, the server determines whether the merchant information carried in the payment token request is consistent with the merchant information carried in the authorization request for token binding corresponding to the authorization code for token binding; if so, the server determines whether the current time is within the validity period of the authorization code for token binding; and if so, it is determined that validation of the payment token request passes.

[0084] In other words, validation of the payment token request passes only if the server determines that the token authorization code and merchant information carried in the received payment token request are correct and the validity period is not exceeded.

[0085] Optionally, first encrypted data, a second secret key and clock information are sent to the first application module while generating the authorization code for token binding, where the first encrypted data is a ciphertext acquired by encrypting the authorization code for token binding with the first secret key; and the method further includes: acquiring a third encrypted data by encrypting the merchant information of the first application module with the second secret key and the clock information, determining whether the third encrypted data is consistent with the second encrypted data received from the first application module, determining whether the authorization code for token binding acquired by decrypting the received first encrypted data with the first secret key is consistent with the authorization code for token binding, and if both determination results are consistent, determining that validation of the payment token request passes.

[0086] The detailed process is as follows.

[0087] Firstly, while generating the authorization code for token binding, the server also generates a first secret key and a second secret key, encrypts the authorization code for token binding with the first secret key to acquire the first encrypted data, and sends the first encrypted data, the second secret key and the clock information to the first application module (either directly to the first application module or indirectly by forwarding by the second application module).

[0088] Subsequently, the first application module performs clock synchronization based on the received clock information and encrypts the merchant information with the second secret key and the synchronized clock to form the second encrypted data. The first application module sends the second encrypted data, and the received first encrypted data to the server.

[0089] Finally, the token service provider (server) encrypts the merchant information stored therein with the second secret key and clock information stored therein to acquire the third encrypted data, determines whether the second encrypted data and the third encrypted data are consistent, decrypts the first encrypted data with the first secret key stored therein, and compares the decryption result with the authorization code for token binding stored therein to determine whether they are consistent. If both determination results are consistent, validation of the payment token request passes.

[0090] Due to the clock information involved, the server is enabled to reject the payment token request if there is no response for a certain period of time, thereby avoiding the user to wait for a long time.

[0091] For example, the first application module may invoke the second application module and send the aforesaid information to the server. After receiving the aforesaid information, the server fails to receive the authorization request for token binding from the second application module in time, and thereby may send a reject message to the first application module.

[0092] Optionally, sending the authorization code for token binding to the first application module includes: sending the authorization code for token binding to the first application module by forwarding via the second application module.

[0093] Upon the server finishes distributing the payment token, the user who adopts the first application module may make a payment based on the payment token. The payment request may be sent to the server. The server replaces the payment token in the payment request with a payment account (e.g., a bank card number), and then sends the payment request to the corresponding card issuer.

[0094] For the payment transaction initiated by the payment token, the server adds a token payment identifier to the payment request upon converting the card number, and the card issuer honors the card number transaction with the token payment identifier but without the validation element.

[0095] Based on the similar inventive concept, with reference to FIG. 4 and in combination with FIG. 5, embodiments of the present disclosure further provide a method of applying for a payment token. The method shows the complete interaction process between the server and the first application module and second application module running on the terminal device. The method of applying for a payment token includes following steps.

[0096] S401: at a terminal device, a first application module invokes a second application module, and transmits merchant information of the first application module to the second application module.

[0097] S402: at the terminal device, the second application modules displays bound payment accounts in the second application module upon receiving an invocation instruction from the first application module, for the user to select one from the bound payment accounts, and sends an authorization request for token binding to a server. The authorization request for token binding carries the merchant information of the first application module and the payment account selected by the user.

[0098] S403: a server generates an authorization code for token binding if validation of the authorization request for token binding passes, and sends the authorization code for token binding to the first application module.

[0099] S404: at the terminal device, the first application module sends a payment token request to the server. The payment token request carries the authorization code for token binding.

[00100] S405: the server generates a payment token if validation of the payment token request passes, and sends the payment token to the first application module.

[00101] The aforesaid embodiments may be referred to for details of the implementation of each step and variants thereof, which will not be repeated.

[00102] Referring to FIG. 5, in a specific example, the first application module in step 1 calls the second application module to select the bank card to be bound in the second application module. In step 2, the second application module sends to the server an authorization request for token binding, the merchant information of the first application module， the transaction control parameters, and the merchant number of the first application module. In step 3, the server validates the authorization request for token

binding and generates the authorization code for token binding. In step 4, the server sends the authorization code for token binding to the second application module. In step 5, the second application module sends the authorization code for token binding and the information for confirmation prompt to the first application module. In step 6, the first application module displays the information for confirmation prompt and prompts the user to confirm whether the information for confirmation prompt is correct. In step 7, the first application module sends to the server a payment token request carrying the merchant information and the authorization code for token binding. In step 8: the server validates the payment token request, generates a payment token if the validation passes, and sends the payment token request to the first application module.

[00103] Referring to FIG. 6, based on the similar inventive concept, embodiments of the present disclosure further provides a terminal device 1 including a first memory 11 and a first processor 12. The first memory 11 has a first instruction and/or a second instruction. The first instruction, when executed by the first processor 12, causes the aforesaid method of applying for a payment token applicable to the first application module to be carried out, and the second instruction, when executed by the first processor 12, causes the aforesaid method of applying for a payment token applicable to the second application module to be carried out.

[00104] Accordingly, embodiments of the present disclosure further provide a server 2. The server 2 includes a second memory 21 and a second processor 22. The second memory 21 has a payment token managing program stored therein, and the payment token managing program, when executed by the second processor 22, causes the method of applying for a payment token applicable to the server as previously described to be carried out.

[00105] Correspondingly, embodiments of the present disclosure provide a system for applying for a payment token, which includes the terminal device and the server.

[00106] The respective embodiments of the present disclosure are described in a progressive manner. The reference may be made to each other for the same or similar parts of the respective embodiments, and each embodiment focuses on the differences from other embodiments.

[00107] The protection scope of the present disclosure is not limited to the aforesaid embodiments, and obviously, one skilled in the art may make various modifications and variations to the present disclosure without departing from the scope of the present disclosure. If the modifications and variations of the present disclosure fall within the scope of the claims and their equivalents, the present disclosure is also intended to

encompass such modifications and variations.

WHAT IS CLAIMED IS:

1. A method of applying for a payment token, applicable to a second application module at a terminal device, comprising:

displaying bound payment accounts upon receiving an invocation instruction from a first application module, for a user to select one from the bound payment accounts, wherein the invocation instruction carries merchant information of the first application module; and

sending to a server an authorization request for token binding, for the first application module to apply to the server for a payment token based on an authorization code for token binding returned by the server, wherein the authorization request for token binding carries the merchant information of the first application module and the payment account selected by the user.

2. The method according to claim 1, wherein the authorization request for token binding further carries transaction control parameters.

3. The method according to claim 2, wherein the transaction control parameters comprise at least one selected from a validity period of the payment token, a transaction volume limit per day of the payment token and the maximum number of transactions per day of the payment token.

4. The method according to claim 2, further comprising providing a setting interface for the user to set the transaction control parameters.

5. The method according to claim 1, further comprising receiving the authorization code for token binding from the server and forwarding the received authorization code for token binding to the first application module.

6. The method according to claim 5, wherein information for confirmation prompt on the payment account and/or on the user’s identification is further sent to the first application module while forwarding the received authorization code for token binding to the first application module.

7. A method of applying for a payment token, applicable to a first application module at a terminal device, comprising:

invoking a second application module, and transmitting merchant information of the

first application module to the second application module, for the second application module to send to a server an authorization request for token binding, wherein the authorization request for token binding carries the merchant information of the first application module and a payment account selected by a user from the second application module, the first application module is a software for invoking the second application module, and the payment accounts selectable by the user are bound beforehand by the user to the second application module;

receiving an authorization code for token binding generated by the server in response to the authorization request for token binding;

sending to the server a payment token request carrying the authorization code for token binding; and

receiving a payment token from the server.

8. The method according to claim 7, wherein the authorization code for token binding is forwarded by the second application module from the server to the first application module;

information for confirmation prompt on the payment account and/or on the user's identification is further received from the second application module while receiving the authorization code for token binding from the second application module; and

the method further comprises: displaying the information for confirmation prompt on the payment account and/or on the user's identification to prompt the user to confirm whether the information for confirmation prompt is correct; and

sending the payment token request to the server if the user confirms that the information for confirmation prompt is correct.

9. The method according to claim 7, further comprising selecting the second application module, which comprises:

sending a query request to the server to query application modules supported by the server; and

displaying a list of the queried application modules for the user to select an application module therefrom as the second application module.

10. The method according to claim 7, wherein an encrypted authentication code and the merchant information of the first application module are further sent to the server if the

second application module is invoked, wherein the encrypted authentication code is a ciphertext of the merchant information of the first application module as acquired by the first application module upon registering with the server.

11. The method according to claim 7, further comprising:

receiving from the server first encrypted data, a second secret key and clock information, wherein the first encrypted data is a ciphertext acquired by encrypting the authorization code for token binding with the first secret key;

synchronizing a local clock according to the received clock information, and acquiring second encrypted data by encrypting the merchant information with the second secret key and the synchronized clock; and

further sending the second encrypted data and the first encrypted data to the server if the second application module is invoked.

12. A method of applying for a payment token, applicable to a server, comprising:

receiving from a second application module an authorization request for token binding, wherein the authorization request for token binding carries merchant information of a first application module, and a payment account selected by a user from bound payment accounts of the second application module;

generating an authorization code for token binding if validation of the authorization request for token binding passes, and sending the authorization code for token binding to the first application module;

receiving from the first application module a payment token request carrying the authorization code for token binding; and

generating a payment token if validation of the payment token request passes, and sending the payment token to the first application module.

13. The method according to claim 12, wherein the authorization request for token binding further carries merchant information of the first application module; and

validation of the authorization request for token binding passes if the merchant information carried by the authorization request for token binding is information of a merchant registered with the server.

14. The method according to claim 12, further comprising:

providing a registration service to the first application module;

encrypting the merchant information of the first application module to acquire an encrypted authentication code;

sending the encrypted authentication code to the first application module;

receiving from the first application module the encrypted authentication code and the merchant information; and

decrypting the encrypted authentication code, comparing a resulting plaintext to the received merchant information, and determining that validation of the authorization request for token binding passes if the plaintext and the merchant information are consistent.

15. The method according to claim 12, wherein the payment token request further carries the merchant information of the first application module, and validation of the payment token request comprises at least one selected from following three validation methods:

determining that the validation passes if the authorization code for token binding exists;

determining that the validation passes if the merchant information is consistent with that carried by the authorization request for token binding; and

determining that the validation passes if the authorization code for token binding is within a validity period.

16. The method according to claim 12, wherein first encrypted data, a second secret key and clock information are further sent to the first application module while generating the authorization code for token binding, wherein the first encrypted data is a ciphertext acquired by encrypting the authorization code for token binding with a first secret key; and

the method further comprises:

receiving from the first application module second encrypted data and the first encrypted data; and

acquiring a third encrypted data by encrypting the merchant information of the first application module with the second secret key and the clock information, determining whether the third encrypted data is consistent with the second encrypted data received

from the first application module, determining whether a result acquired by decrypting the received first encrypted data with the first secret key is consistent with the authorization code for token binding, and if both determination results are consistent, determining that validation of the payment token request passes.

17. The method according to claim 12, wherein sending the authorization code for token binding to the first application module comprises sending the authorization code for token binding to the first application module by forwarding via the second application module.

18. A method of applying for a payment token, comprising:

at a terminal device, invoking, by a first application module, a second application module, and transmitting merchant information of the first application module to the second application module;

at the terminal device, displaying, by the second application module, bound payment accounts in the second application module upon receiving an invocation instruction from the first application module, for a user to select one from the bound payment accounts, and sending an authorization request for token binding to a server, wherein the authorization request for token binding carries the merchant information of the first application module and the payment account selected by the user;

generating, by the server, an authorization code for token binding if validation of the authorization request for token binding passes, and sending the authorization code for token binding to the first application module;

at the terminal device, sending, by the first application module, a payment token request to the server, wherein the payment token request carries the authorization code for token binding; and

generating, by the server, a payment token if validation of the payment token request passes, and sending the payment token to the first application module.

19. A terminal device, comprising a first memory and a first processor, the first memory having a first instruction and/or a second instruction stored therein, wherein the first instruction, when executed by the first processor, causes the method of applying for a payment token according to any one of claims 7-11 to be performed, and the second instruction, when executed by the first processor, causes the method of applying for a payment token according to any one of claims 1-6 to be performed.

20. A server, comprising a second memory and a second processor, the second memory having a payment token managing program stored therein, wherein the payment token managing program, when executed by the second processor, causes the method of applying for a payment token according to any one of claims 12-17 to be performed.

21. A system for applying for a payment token, comprising the terminal device according to claim 19 and the server according to claim 20.