

KEY TRANSFER METHOD AND SYSTEM BASED ON SHARED SECURITY APPLICATION, STORAGE MEDIUM, AND DEVICE THEREOF

5

TECHNICAL FIELD

[0001] One or a plurality of embodiments of the present specification relate to the field of key communication of a mobile terminal, and in particular, to a key transfer method and system based on a shared security application, a storage
10 medium, and a device thereof.

BACKGROUND

[0002] As mobile intelligent terminals become more and more powerful and have increasingly improved processing performance, secure elements have
15 been more and more integrated in mobile intelligent terminals and applied to more and more service scenarios having high security requirements. In conventional mobile applications using a secure element for secure computing, for each mobile application, a corresponding independent security application needs to be installed in the secure element, and the independent security application is responsible for
20 accomplishing secure computing requirements of the mobile application.

[0003] For some mobile applications, especially for those having unified requirements for secure computing, for example, when corresponding service keys are mainly used to perform encryption and decryption operations or a signing operation, if the conventional approach of separately installing
25 corresponding secure applications is used, high development costs and promotion costs will be caused. A faster or more reliable solution needs to be provided to implement key transfer based on a shared security application.

SUMMARY

[0004] In view of at least one of the aforementioned technical problems, one or a plurality of embodiments of the present specification provide a key transfer method and system based on a shared security application, an electronic device, and a storage medium, wherein a lightweight secure computing solution based on a secure element is employed, so that background docking costs can be avoided when service keys of different applications are securely downloaded to a shared security application; meanwhile, communication security protection keys are combined with service keys, so that flow processing on the service level can be implemented through the service keys while secure communication between mobile applications and the shared security application can be implemented through the communication security protection keys.

[0005] In order to achieve the aforementioned objective, one or a plurality of embodiments of the present specification provide a key transfer method based on a shared security application, comprising the following steps:

[0006] S1, generating, by a management server of a mobile application, an application key applied to the mobile application in an execution environment in a mobile terminal, wherein the application key comprises a service key, and delivering, by the management server of the mobile application, the application key to the mobile terminal; and

[0007] receiving, by a management server of a shared security application, the application key and delivering the application key to the shared security application in a secure element in the mobile terminal, wherein

[0008] the shared security application stores application keys of several mobile applications for responding to invocation of service operations of the respective mobile applications.

[0009] Preferably, the application key comprises a communication security protection key, the management server of the mobile application delivers the communication security protection key to the mobile terminal, and the mobile application in the execution environment in the mobile terminal stores the

communication security protection key.

5 **[0010]** Preferably, the execution environment in the mobile terminal is configured with a shared security application proxy, the shared security application proxy provides an interface between the mobile application in the mobile terminal and the corresponding application key in the shared security application, and the service key is delivered to the shared security application in the secure element in the mobile terminal by the management server of the shared security application and the shared security application proxy in the execution environment in the mobile terminal.

10 **[0011]** Further, the generation of the application key specifically comprises the following substeps:

[0012] S11, distributing, by a management server of a shared security application, a public key thereof to several management servers of mobile applications;

15 **[0013]** S12, generating, by the several management servers of the mobile applications, respective application keys for several corresponding mobile applications, wherein the application keys comprise communication security protection keys and service keys; and

[0014] S13, encrypting, by the several management servers of the mobile applications, application keys to be written to the shared security application using the public key of the management server of the shared security application, and delivering the encrypted application keys to corresponding mobile applications in the mobile terminal.

20 **[0015]** Further, delivering the application key to a shared security application in a secure element in the mobile terminal specifically comprises the following steps:

[0016] S15, invoking, by several mobile applications, corresponding interfaces provided by the shared security application proxy in the mobile terminal, to import respective application keys encrypted and protected using the public key of the management server of the shared security application;

[0017] S16, sending, by the shared security application proxy, the imported encrypted application keys of the several mobile applications to the management server of the shared security application;

5 [0018] S17, performing, by the management server of the shared security application, decryption using a private key thereof to obtain application keys of the several mobile applications to be written to the shared security application; and

[0019] S18, delivering, by the management server of the shared security application, the application keys of the several mobile applications to the shared security application in the secure element of the mobile terminal.

[0020] Further, the shared security application securely stores the application keys of the several mobile applications, and application keys of different mobile applications are isolated from each other.

15 [0021] Preferably, the mobile applications are placed in a rich execution environment and/or a trusted execution environment of the mobile terminal.

[0022] Preferably, the shared security application proxy is placed in a rich execution environment and/or a trusted execution environment of the mobile terminal.

20 [0023] Preferably, the public key and the private key of the management server of the shared security application are an asymmetric key pair.

[0024] A key transfer method based on a shared security application, comprises the following steps:

25 [0025] S2: invoking, by a mobile application in an execution environment in a mobile terminal, a corresponding communication security protection key of the mobile application in a shared security application through an interface provided by a shared security application proxy in the execution environment in the mobile terminal to first decrypt service data, then performing service processing using a service key, and returning a service processing result to the corresponding mobile application.

30 [0026] Further, step S2 comprises the following substeps:

[0027] S21, initiating, by the mobile application, a service operation request, and encrypting, by the mobile application using the communication protection key, service data to be processed using the service key;

5 [0028] S22, sending, by the mobile application, the encrypted service data to the shared security application by invoking the interface provided by the shared security application proxy; and

[0029] S23, decrypting, by the shared security application, the service data using the communication security protection key of the corresponding mobile application, then performing processing using the service key, and returning a
10 processing result to the mobile application.

[0030] Preferably, the mobile applications are placed in a rich execution environment and/or a trusted execution environment of the mobile terminal.

[0031] Preferably, the shared security application proxy is placed in a rich execution environment and/or a trusted execution environment of the mobile
15 terminal.

[0032] A key transfer system based on a shared security application comprises a shared security application built in a secure element of a mobile terminal, several mobile applications and a shared security application proxy built in an execution environment of the mobile terminal, and management servers of
20 the mobile applications, wherein the management servers of the mobile applications are configured to generate respective application keys for several corresponding mobile applications, wherein the application keys comprise service keys;

[0033] the shared security application securely stores the several
25 application keys, and the shared security application proxy provides interfaces between the several mobile applications and the corresponding application keys in the shared security application; and

[0034] the mobile applications invoke the corresponding service keys in the shared security application through the interfaces provided by the shared
30 security application proxy to perform service processing, and return service

processing results to the corresponding mobile applications. The key transfer system based on a shared security application improves the security level of a service product of a mobile terminal, and can be applied to, for example, scenarios such as intelligent door lock services and two-dimensional code anti-counterfeiting.

5 **[0035]** Preferably, the application keys further comprise communication security protection keys, the management servers of the mobile applications deliver the communication security protection keys to the mobile terminal, and the mobile applications in the execution environment in the mobile terminal store
10 the communication security protection keys.

[0036] Preferably, the key transfer system based on a shared security application further comprises a management server of the shared security application, wherein the management server of the shared security application is configured to distribute a public key thereof to the several management servers of
15 the mobile applications; the several management servers of the mobile applications encrypt the application keys to be written to the shared security application using the public key of the management server of the shared security application and deliver the application keys to the corresponding mobile applications of the mobile terminal; the several mobile applications invoke the
20 corresponding interfaces provided by the shared security application proxy to import the respective application keys encrypted and protected using the management server of the shared security application; the shared security application proxy sends the several imported encrypted application keys to the management server of the shared security application; the management server of
25 the shared security application performs decryption using a private key of the management server of the shared security application to obtain the several application keys to be written to the shared security application; the management server of the shared security application delivers the several application keys to the shared security application; and the shared security application securely stores
30 the several application keys, and the application keys of different mobile

applications are isolated from each other.

5 **[0037]** Preferably, the mobile application initiates a service operation request, and the mobile application encrypts, using the communication protection key, service data to be processed using the service key; the mobile application sends the encrypted service data to the shared security application by invoking the interface provided by the shared security application proxy; and the shared security application decrypts the service data using the corresponding communication security protection key, then processes the decrypted service data using the service key, and returns a processing result to the mobile application.

10 **[0038]** Preferably, the mobile application is placed in a rich execution environment and/or a trusted execution environment of the mobile terminal.

[0039] Preferably, the shared security application proxy is placed in a rich execution environment and/or a trusted execution environment of the mobile terminal.

15 **[0040]** Preferably, the public key and the private key of the management server of the shared security application are an asymmetric key pair, and an encryption algorithm of the asymmetric key pair comprises any one of RSA, Elgamal, a knapsack algorithm, Rabin, D-H, and ECC.

20 **[0041]** An electronic device comprises a memory, a processor, and a computer program stored on the memory and executable by the processor, wherein the processor, when executing the program, implements the key transfer method based on a shared security application described above.

25 **[0042]** A computer-readable storage medium comprises a computer program stored thereon, wherein the computer program, when executed by a processor, implements the key transfer method based on a shared security application described above.

[0043] As compared with the current technologies, one or a plurality of embodiments of the present specification have the following advantages:

30 **[0044]** one or a plurality of embodiments of the present specification provide a key transfer method based on a shared security application, wherein a

shared security application placed in a secure element stores application keys invoked by a plurality of mobile applications, and meanwhile, the shared security application provides security isolation protection for an independent and dedicated application key of each mobile application. The one or plurality of
5 embodiments of the present specification further relate to a key transfer system based on a shared security application, an electronic device, and a storage medium. The one or plurality of embodiments of the present specification provide a novel security application solution architecture, especially for mobile applications having common secure computing requirements, by employing a
10 lightweight secure computing solution based on a secure element, so that background docking costs can be avoided when service keys of different applications are securely downloaded to a shared security application; meanwhile, communication security protection keys are combined with service keys, so that flow processing on the service level can be implemented through the service keys
15 while secure communication between mobile applications and the shared security application can be implemented through the communication security protection keys. Especially for mobile applications that need to realize simple and universal secure computing requirements through a secure element, it is unnecessary to independently develop an application software development kit and download,
20 install, and personalize the kit to the secure element, thereby greatly reducing the costs of secure computing performed by the mobile applications using the secure element.

[0045] The above description is merely an overview of technical solutions in the one or plurality of embodiments of the present specification. In order to
25 understand the technical means in the one or plurality of embodiments of the present specification more clearly and implement the technical means according to the content of the specification, preferred embodiments of the one or plurality of embodiments of the present specification are described in detail below with reference to the accompanying drawings. The specific implementation of the one
30 or plurality of embodiments of the present specification is provided in detail by

the following embodiments and accompanying drawings thereof.

BRIEF DESCRIPTION OF THE DRAWINGS

5 [0046] Further detailed description is provided below with reference to the accompanying drawings and the implementation of one or a plurality of embodiments of the present specification.

 [0047] FIG. 1 is a flowchart of a key transfer method based on a shared security application in one or a plurality of embodiments of the present specification;

10 [0048] FIG. 2 is a flowchart illustrating substeps of step S1 in the key transfer method based on a shared security application in one or a plurality of embodiments of the present specification;

 [0049] FIG. 3 is a flowchart illustrating substeps of step S2 in the key transfer method based on a shared security application in one or a plurality of
15 embodiments of the present specification;

 [0050] FIG. 4 is a schematic diagram of a key transfer system based on a shared security application in one or a plurality of embodiments of the present specification;

20 [0051] FIG. 5 is a schematic diagram of the key transfer system based on a shared security application in Embodiment 1 of the present specification;

 [0052] FIG. 6 is a schematic diagram of the key transfer system based on a shared security application in Embodiment 2 of the present specification;

 [0053] FIG. 7 is a schematic diagram of the key transfer system based on a shared security application in Embodiment 3 of the present specification;

25 [0054] FIG. 8 is a schematic diagram of the key transfer system based on a shared security application in Embodiment 4 of the present specification;

 [0055] FIG. 9 is a schematic diagram of the key transfer system based on a shared security application in Embodiment 5 of the present specification;

30 [0056] FIG. 10 is a schematic diagram of the key transfer system based on a shared security application in Embodiment 6 of the present specification; and

[0057] FIG. 11 is a schematic diagram of the key transfer system based on a shared security application in Embodiment 7 of the present specification.

DETAILED DESCRIPTION

5 [0058] In order to make the objectives, technical solutions and advantages of one or a plurality of embodiments of the present specification clearer, the one or plurality of embodiments of the present specification are further described in detail below with reference to the accompanying drawings and embodiments. It should be understood that the specific embodiments described here are merely
10 used for explaining one or a plurality of embodiments of the present specification, not for limiting the one or plurality of embodiments of the present specification.

 [0059] As shown in FIG. 1, a key transfer method based on a shared security application includes the following steps:

 [0060] S1, a management server of a mobile application generates an
15 application key applied to a mobile application in an execution environment in a mobile terminal, wherein the application key includes a service key, and the management server of the mobile application delivers the application key to the mobile terminal; and the management server of the shared security application receives the application key and delivers the application key to a shared security
20 application in a secure element in the mobile terminal; wherein the shared security application stores application keys of several mobile applications for responding to invocation of service operations of the respective mobile applications.

 [0061] In one embodiment or a plurality of embodiments (not shown in the drawing), a service key may be generated by a management server of a mobile
25 application and then directly delivered back to a management server of a shared security application, and delivered to the shared security application in a secure element in a mobile terminal through the management server of the shared security application, and application keys of several mobile applications are stored in the shared security application for invocation by service operations.
30 Compared with the conventional security application authentication, background

(server side) docking costs can be avoided when service keys of different third-party applications are securely downloaded to a shared security application, and a lightweight secure computing solution based on a secure element is provided for mobile applications having common secure computing requirements.

5 **[0062]** In one embodiment or a plurality of embodiments, the application key further includes a communication security protection key, the management server of the mobile application delivers the communication security protection key to the mobile terminal, and the mobile application in the execution environment in the mobile terminal stores the communication security protection
10 key. The communication security protection key is used for protecting a communication process of the mobile application, preventing it from being stolen; the service key is used for protecting a service delivery and execution process of each mobile application after initiation of a service request, preventing it from being stolen.

15 **[0063]** In one embodiment or a plurality of embodiments, the execution environment in the mobile terminal is configured with a shared security application proxy, the shared security application proxy provides an interface between the mobile application in the mobile terminal and the corresponding application key in the shared security application, and the service key is delivered
20 to the shared security application in the secure element in the mobile terminal through the management server of the shared security application and the shared security application proxy in the execution environment in the mobile terminal. It should be understood that step S1 may exist independently; that is, step S1 may be implemented as a delivery configuration phase of an application key in one
25 embodiment or a plurality of embodiments. An existing management server of a mobile application generates a corresponding application key and all application keys are deployed in a shared security application, wherein application keys of different mobile applications are stored while being isolated from each other in the shared security application to prepare for subsequent response to service
30 requests of mobile applications. It should be further understood that the

application keys include communication security protection keys and/or service keys, the communication security protection keys are used for protecting communication processes of the mobile applications, preventing it from being stolen; the service keys are used for protecting service delivery and execution processes of the mobile applications after initiation of service requests, preventing it from being stolen. In one embodiment or a plurality of embodiments, compared with the conventional security application authentication, background (server side) docking costs can be avoided when service keys of different third-party applications are securely downloaded to a shared security application, and a lightweight secure computing solution based on a secure element is provided for mobile applications having common secure computing requirements.

[0064] Specifically, in one embodiment or a plurality of embodiments, when the application key includes only the communication security protection key, the application key implements secure communication between mobile application And the shared security application through only the communication security protection key; in one embodiment or a plurality of embodiments, when the application key includes only the service key, the application key implements protection of flow processing on the service level through only the service key; in one embodiment or a plurality of embodiments, when the application key includes both the communication security protection key and the service key, the application key implements secure communication between mobile application And the shared security application through the communication security protection key, and implements protection of flow processing on the service level through the service key.

[0065] As shown in FIG. 2, step S1 includes the following substeps:

[0066] S11, the management server of a shared security application distributes a public key thereof to several management servers of mobile applications;

[0067] S12, the several management servers of the mobile applications generate respective application keys for several corresponding mobile

applications, wherein the application keys include communication security protection keys and service keys;

5 **[0068]** S13, the several management servers of the mobile applications encrypt application keys to be written to the shared security application using the public key of the management server of the shared security application, and deliver the application keys to corresponding mobile applications in the mobile terminal;

[0069] S14, several mobile applications store respective communication security protection keys;

10 **[0070]** S15, several mobile applications invoke corresponding interfaces provided by the shared security application proxy in the mobile terminal to import respective application keys encrypted and protected using the public key of the management server of the shared security application;

15 **[0071]** S16. the shared security application proxy sends the imported encrypted application keys of the several mobile applications to the management server of the shared security application;

[0072] S17, the management server of the shared security application performs decryption using a private key thereof to obtain application keys of the several mobile applications to be written to the shared security application;

20 **[0073]** S18, the management server of the shared security application delivers the application keys of the several mobile applications to the shared security application in the secure element of the mobile terminal; and

25 **[0074]** S19, the shared security application securely stores the application keys of the several mobile applications, wherein application keys of different mobile applications are isolated from each other.

[0075] As shown in FIG. 1, the key transfer method based on a shared security application further includes step S2: a mobile application in an execution environment in a mobile terminal invokes a corresponding communication security protection key of the mobile application in a shared security application
30 through an interface provided by a shared security application proxy in the

execution environment in the mobile terminal to first decrypt service data, then performs service processing using a service key, and returns a service processing result to the corresponding mobile application. It should be understood that step S2 is implemented in one or a plurality of embodiments as that a shared security application in a mobile terminal responds to a service request of a mobile application and processes the service request by using an application key in the shared security application.

[0076] As shown in FIG. 3, step S2 includes the following substeps:

[0077] S21, the mobile application initiates a service operation request, and the mobile application encrypts, using the communication protection key, service data to be processed using the service key;

[0078] S22, the mobile application sends the encrypted service data to the shared security application by invoking the interface provided by the shared security application proxy; and

[0079] S23, the shared security application decrypts the service data using the communication security protection key of the corresponding mobile application, then performs processing using the service key, and returns a processing result to the mobile application to be used to complete the subsequent service flow.

[0080] In one embodiment or a plurality of embodiments, communication security protection keys may be configured as symmetric keys; for example, an encryption algorithm including, but not limited to, any one of DES, 3DES, IDEA, FEAL, and BLOWFISH is employed to configure the communication security protection keys as symmetric keys. In this application scenario, communication security protection keys of several mobile applications in the shared security application may be configured as the same key or different keys. Similarly, the communication security protection keys may also be configured as asymmetric keys; for example, an encryption algorithm of asymmetric keys including, but not limited to, any one of RSA, Elgamal, a knapsack algorithm, Rabin, D-H, and ECC is employed to configure communication security protection keys in mobile

applications in the mobile terminal as public keys for encryption and encapsulation, and configure communication security protection keys in the shared security application as private keys for decryption.. In this application scenario, communication security protection keys of several mobile applications
5 in the shared security application are configured as different keys.

[0081] In one embodiment or a plurality of embodiments, service keys may be configured as symmetric keys; for example, an encryption algorithm including, but not limited to, any one of DES, 3DES, IDEA, FEAL, and BLOWFISH is employed to configure the service keys as symmetric keys. In this
10 application scenario, service keys of several mobile applications in the shared security application may be configured as the same key or different keys. Similarly, the service keys may also be configured as asymmetric keys; for example, an encryption algorithm of asymmetric keys including, but not limited to any one of RSA, Elgamal, a knapsack algorithm, Rabin, D-H, and ECC is
15 employed to configure the service keys in the shared security application as public keys for processing and encapsulation. In this application scenario, service keys of several mobile applications in the shared security application are configured as different keys.

[0082] As shown in FIG. 4, a key transfer system based on a shared
20 security application includes a shared security application built in a secure element of a mobile terminal, several mobile applications (for example, a mobile application A and a mobile application B) and a shared security application proxy built in an execution environment of the mobile terminal, and management servers of the mobile applications (for example, a management server of mobile
25 application A and a management server of mobile application B). The management servers of the mobile applications are configured to generate respective application keys for several corresponding mobile applications, wherein the application keys include communication security protection keys and service keys. The mobile applications in the execution environment in the mobile
30 terminal store the communication security protection keys. The shared security

application securely stores the several application keys (for example, an application key of mobile application A and an application key of mobile application B), and the shared security application proxy provides interfaces between the several mobile applications and the corresponding application keys in the shared security application (for example, the shared security application proxy provides an interface between mobile application A and the application key of mobile application A in the shared security application, and the shared security application proxy provides an interface between mobile application B and the application key of mobile application B in the shared security application). The mobile applications invoke the corresponding communication security protection keys in the shared security application through the interfaces provided by the shared security application proxy to first decrypt service data, and then perform service processing using the service keys, and return service processing results to the corresponding mobile applications to be used to complete the subsequent service flow. The secure element can provide a more secure data storage and computing environment. The shared security application is located in the secure element, allows a mobile application to download an application key (including a communication security protection key and a service key) thereof, and provides a standard secure computing function for the mobile application. The shared security application proxy is responsible for providing an access interface for the mobile application, and implements life cycle management, for example, operations such as download, installation, use, and deletion, on the service key of the mobile application in the shared application. It should be understood that in this implementation, when the application key includes only the service key, the application key is applied to scenarios having no requirement on communication security or being protected using other communication security protection mechanisms.

[0083] Preferably, as shown in FIG. 4, the key transfer system based on a shared security application further includes a management server of the shared security application, wherein the management server of the shared security

application is configured to distribute a public key thereof to the several management servers of the mobile applications; the several management servers of the mobile applications encrypt the application keys to be written to the shared security application using the public key of the management server of the shared security application and deliver the application keys to the corresponding mobile applications of the mobile terminal; the several mobile applications invoke the corresponding interfaces provided by the shared security application proxy to import the respective application keys encrypted and protected using the management server of the shared security application; the shared security application proxy sends the several imported encrypted application keys to the management server of the shared security application; the management server of the shared security application performs decryption using a private key of the management server of the shared security application to obtain the several application keys to be written to the shared security application; the management server of the shared security application delivers the several application keys to the shared security application; and the shared security application securely stores the several application keys, and the application keys of different mobile applications are isolated from each other.

[0084] Preferably, the mobile application initiates a service operation request, and the mobile application encrypts, using the communication protection key, service data to be processed using the service key; the mobile application sends the encrypted service data to the shared security application by invoking the interface provided by the shared security application proxy; and the shared security application decrypts the service data using the corresponding communication security protection key, then processes the decrypted service data using the service key, and returns a processing result to the mobile application.

[0085] Preferably, the execution environment of the mobile terminal is a set of software and hardware existing in a mobile device, which can provide necessary capability support for execution of applications in the mobile device, and generally includes components such as a hardware processing unit, a volatile

storage unit, a non-volatile storage unit, an operating system, an invocation interface, etc. Generally, the execution environment includes a rich execution environment and a trusted execution environment. The rich execution environment is an open execution environment running in the mobile device, which provides open and rich running capability support for the execution of applications therein, but has relatively weak security protection capability. The trusted execution environment is an isolated execution environment running in the mobile device, which has stronger security capability than the rich execution environment to ensure that applications, sensitive data, and so on running therein are stored, processed, and protected in a relatively trusted environment. The mobile applications are placed in the rich execution environment and/or the trusted execution environment of the mobile terminal. The shared security application proxy is placed in the rich execution environment and/or the trusted execution environment of the mobile terminal. In this embodiment, a key pair (a public key and a private key) of the management server of the shared security application is configured as asymmetric keys. For example, an encryption algorithm of asymmetric keys including, but not limited to, any one of RSA, Elgamal, a knapsack algorithm, Rabin, D-H, and ECC is employed to configure the key pair of the management server of the shared security application as a public key and a private key for encapsulation of encryption or decryption.

[0086] The situation that the mobile applications are placed in the rich execution environment and/or the trusted execution environment of the mobile terminal, and the situation that the shared security application proxy is placed in the rich execution environment and/or the trusted execution environment of the mobile terminal are respectively described below through Embodiments 1 to 7 with reference to FIG. 5 to FIG. 11. It should be understood that in Embodiments 1 to 7, the application key is configured as a communication security protection key and a service key, and the application key implements secure communication between mobile application And the shared security application through the communication security protection key and implements protection of flow

processing on the service level through the service key.

[0087] Embodiment 1

[0088] As shown in FIG. 5, the management server of the shared security application distributes a public key thereof to the management server of mobile application A and the management server of mobile application B; the
5 management server of mobile application A generates an application key of mobile application A, the management server of mobile application B generates an application key of mobile application B, and the application keys include communication security protection keys and service keys; the management server
10 of mobile application A and the management server of mobile application B respectively encrypt application keys to be written to the shared security application using a public key of the management server of the shared security application, and deliver the application keys to corresponding mobile applications (for example, mobile application A and mobile application B) in the rich
15 execution environment of the mobile terminal; the mobile applications (for example, mobile application A and mobile application B) store respective communication security protection keys; the mobile applications in the rich execution environment invoke corresponding interfaces provided by a shared security application proxy that is also in the rich execution environment to import
20 respective application keys encrypted and protected using the management server of the shared security application; the shared security application proxy in the rich execution environment sends the imported encrypted application keys of the mobile applications (for example, the application key of mobile application A and the application key of mobile application B) to the management server of the
25 shared security application; the management server of the shared security application performs decryption using a private key thereof to obtain the application keys of the mobile applications (for example, the application key of mobile application A and the application key of mobile application B) to be written to the shared security application; the management server of the shared
30 security application delivers the application keys of the mobile applications (for

example, the application key of mobile application A and the application key of mobile application B) to the shared security application in the secure element of the mobile terminal; the shared security application securely stores the application keys of the mobile applications (for example, the application key of mobile application A and the application key of mobile application B), wherein the application key of mobile application A and the application key of mobile application B are stored while being isolated from each other, waiting for responding to service requests of the mobile applications in the rich execution environment.

10 **[0089]** Mobile application A or mobile application B in the rich execution environment initiates a service operation request, and mobile application A or mobile application B encrypts, using the communication protection key, service data to be processed using the service key; mobile application A or mobile application B sends the encrypted service data to the shared security application
15 by invoking the interface provided by the shared security application proxy in the rich execution environment; and the shared security application decrypts the service data using the communication security protection key of the corresponding mobile application, then performs processing using the service key, and returns a processing result to mobile application A or mobile application B in
20 the rich execution environment, so as to be used to complete the service flow of mobile application A or mobile application B in the rich execution environment. Compared with the conventional security application authentication, background (server side) docking costs can be avoided when service keys of mobile applications in a rich execution environment are securely downloaded to a shared
25 security application, and a lightweight secure computing solution based on a secure element is provided for mobile applications in the rich execution environment having common secure computing requirements.

[0090] Embodiment 2

30 **[0091]** As shown in FIG. 6, the management server of the shared security application distributes a public key thereof to the management server of mobile

application A and the management server of mobile application B; the management server of mobile application A generates an application key of mobile application A, the management server of mobile application B generates an application key of mobile application B, and the application keys include communication security protection keys and service keys; the management server of mobile application A and the management server of mobile application B respectively encrypt application keys to be written to the shared security application using a public key of the management server of the shared security application, and deliver the application keys to the corresponding mobile applications (for example, mobile application A and mobile application B) in the trusted execution environment of the mobile terminal; the mobile applications (for example, mobile application A and mobile application B) in the trusted execution environment store respective communication security protection keys; the mobile applications (for example, mobile application A and mobile application B) in the trusted execution environment invoke corresponding interfaces provided by a shared security application proxy in the rich execution environment to import respective application keys encrypted and protected using the management server of the shared security application; the shared security application proxy in the rich execution environment sends the imported encrypted application keys (for example, the application key of mobile application A and the application key of mobile application B) to the management server of the shared security application; the management server of the shared security application performs decryption using a private key thereof to obtain the application keys of the mobile applications (for example, the application key of mobile application A and the application key of mobile application B) to be written to the shared security application; the management server of the shared security application delivers the application keys of the mobile applications (for example, the application key of mobile application A and the application key of mobile application B) to the shared security application in the secure element of the mobile terminal; the shared security application securely stores the application keys of the mobile

applications (for example, the application key of mobile application A and the application key of mobile application B), wherein the application key of mobile application A and the application key of mobile application B are stored while being isolated from each other, waiting for responding to service requests of the mobile applications in the trusted execution environment.

[0092] Mobile application A or mobile application B in the trusted execution environment initiates a service operation request, and mobile application A or mobile application B encrypts, using the communication protection key, service data to be processed using the service key; mobile application A or mobile application B sends the encrypted service data to the shared security application by invoking the interface provided by the shared security application proxy in the rich execution environment; and the shared security application decrypts the service data using the communication security protection key of the corresponding mobile application, then performs processing using the service key, and returns a processing result to mobile application A or mobile application B in the trusted execution environment, so as to be used to complete the service flow of mobile application A or mobile application B in the trusted execution environment. Compared with the conventional security application authentication, background (server side) docking costs can be avoided when service keys of mobile applications in a trusted execution environment are securely downloaded to a shared security application, and a lightweight secure computing solution based on a secure element is provided for mobile applications in the trusted execution environment having common secure computing requirements.

[0093] Embodiment 3

[0094] As shown in FIG. 7, the management server of the shared security application distributes a public key thereof to the management server of mobile application A and the management server of mobile application B; the management server of mobile application A generates an application key of mobile application A, the management server of mobile application B generates

an application key of mobile application B, and the application keys include communication security protection keys and service keys; the management server of mobile application A and the management server of mobile application B respectively encrypt application keys to be written to the shared security application using a public key of the management server of the shared security application, and deliver the application keys to mobile application A in the trusted execution environment of the mobile terminal and mobile application B in the rich execution environment; mobile application A in the trusted execution environment and mobile application B in the rich execution environment store respective communication security protection keys; mobile application A in the trusted execution environment and mobile application B in the rich execution environment respectively invoke corresponding interfaces provided by a shared security application proxy in the rich execution environment to import respective application keys encrypted and protected using the management server of the shared security application; the shared security application proxy in the rich execution environment sends the imported encrypted application keys (for example, the application key of mobile application A and the application key of mobile application B) to the management server of the shared security application; the management server of the shared security application performs decryption using a private key thereof to obtain the application keys of the mobile applications (for example, the application key of mobile application A and the application key of mobile application B) to be written to the shared security application; the management server of the shared security application delivers the application keys of the mobile applications (for example, the application key of mobile application A and the application key of mobile application B) to the shared security application in the secure element of the mobile terminal; the shared security application securely stores the application keys of the mobile applications (for example, the application key of mobile application A and the application key of mobile application B), wherein the application key of mobile application A and the application key of mobile application B are stored while

being isolated from each other, waiting for responding to a service request of the mobile application in the trusted execution environment or a service request of the mobile application in the rich execution environment. In this implementation, mobile application A represents a type of mobile applications in the trusted
5 execution environment, and mobile application B represents a type of mobile applications in the rich execution environment.

[0095] Mobile application A in the trusted execution environment or mobile application B in the rich execution environment initiates a service operation request, and mobile application A or mobile application B encrypts,
10 using the communication protection key, service data to be processed using the service key; mobile application A or mobile application B sends the encrypted service data to the shared security application by invoking the interface provided by the shared security application proxy in the rich execution environment; and the shared security application decrypts the service data using the communication
15 security protection key of the corresponding mobile application, then performs processing using the service key, and returns a processing result to mobile application A in the trusted execution environment or mobile application B in the rich execution environment, so as to be used to complete the service flow of mobile application A in the trusted execution environment or mobile application
20 B in the rich execution environment. Compared with the conventional security application authentication, background (server side) docking costs can be avoided when service keys of mobile applications in a trusted execution environment or mobile applications in a rich execution environment are securely downloaded to a shared security application, and a lightweight secure computing solution based on
25 a secure element is provided for mobile applications in the trusted execution environment and mobile applications in the rich execution environment having common secure computing requirements.

[0096] Embodiment 4

[0097] As shown in FIG. 8, the management server of the shared security
30 application distributes a public key thereof to the management server of mobile

application A and the management server of mobile application B; the management server of mobile application A generates an application key of mobile application A, the management server of mobile application B generates an application key of mobile application B, and the application keys include communication security protection keys and service keys; the management server of mobile application A and the management server of mobile application B respectively encrypt application keys to be written to the shared security application using a public key of the management server of the shared security application, and deliver the application keys to corresponding mobile applications (for example, mobile application A and mobile application B) in the trusted execution environment of the mobile terminal; the mobile applications (for example, mobile application A and mobile application B) store respective communication security protection keys; the mobile applications in the trusted execution environment invoke corresponding interfaces provided by a shared security application proxy that is also in the trusted execution environment to import respective application keys encrypted and protected using the management server of the shared security application; the shared security application proxy in the trusted execution environment sends the imported encrypted application keys (for example, the application key of mobile application A and the application key of mobile application B) to the management server of the shared security application; the management server of the shared security application performs decryption using a private key thereof to obtain the application keys of the mobile applications (for example, the application key of mobile application A and the application key of mobile application B) to be written to the shared security application; the management server of the shared security application delivers the application keys of the mobile applications (for example, the application key of mobile application A and the application key of mobile application B) to the shared security application in the secure element of the mobile terminal; the shared security application securely stores the application keys of the mobile applications (for example, the application key of mobile application A and the

application key of mobile application B), wherein the application key of mobile application A and the application key of mobile application B are stored while being isolated from each other, waiting for responding to service requests of the mobile applications in the trusted execution environment.

5 **[0098]** Mobile application A or mobile application B in the trusted execution environment initiates a service operation request, and mobile application A or mobile application B encrypts, using the communication protection key, service data to be processed using the service key; mobile application A or mobile application B sends the encrypted service data to the
10 shared security application by invoking the interface provided by the shared security application proxy in the trusted execution environment; and the shared security application decrypts the service data using the communication security protection key of the corresponding mobile application, then performs processing using the service key, and returns a processing result to mobile application A or
15 mobile application B in the trusted execution environment, so as to be used to complete the service flow of mobile application A or mobile application B in the trusted execution environment. Compared with the conventional security application authentication, background (server side) docking costs can be avoided when service keys of mobile applications in a trusted execution environment are
20 securely downloaded to a shared security application, and a lightweight secure computing solution based on a secure element is provided for mobile applications in the trusted execution environment having common secure computing requirements.

[0099] Embodiment 5

25 **[00100]** As shown in FIG. 9, the management server of the shared security application distributes a public key thereof to the management server of mobile application A and the management server of mobile application B; the management server of mobile application A generates an application key of mobile application A, the management server of mobile application B generates
30 an application key of mobile application B, and the application keys include

communication security protection keys and service keys; the management server of mobile application A and the management server of mobile application B respectively encrypt application keys to be written to the shared security application using a public key of the management server of the shared security application, and deliver the application keys to corresponding mobile applications (for example, mobile application A and mobile application B) in the rich execution environment of the mobile terminal; the mobile applications (for example, mobile application A and mobile application B) in the rich execution environment store respective communication security protection keys; the mobile applications (for example, mobile application A and mobile application B) in the rich execution environment invoke corresponding interfaces provided by a shared security application proxy in the trusted execution environment to import respective application keys encrypted and protected using the management server of the shared security application; the shared security application proxy in the trusted execution environment sends the imported encrypted application keys (for example, the application key of mobile application A and the application key of mobile application B) to the management server of the shared security application; the management server of the shared security application performs decryption using a private key thereof to obtain the application keys of the mobile applications (for example, the application key of mobile application A and the application key of mobile application B) to be written to the shared security application; the management server of the shared security application delivers the application keys of the mobile applications (for example, the application key of mobile application A and the application key of mobile application B) to the shared security application in the secure element of the mobile terminal; the shared security application securely stores the application keys of the mobile applications (for example, the application key of mobile application A and the application key of mobile application B), wherein the application key of mobile application A and the application key of mobile application B are stored while being isolated from each other, waiting for responding to service requests of the

mobile applications in the rich execution environment.

[00101] Mobile application A or mobile application B in the rich execution environment initiates a service operation request, and mobile application A or mobile application B encrypts, using the communication protection key, service data to be processed using the service key; mobile application A or mobile application B sends the encrypted service data to the shared security application by invoking the interface provided by the shared security application proxy in the trusted execution environment; and the shared security application decrypts the service data using the communication security protection key of the corresponding mobile application, then performs processing using the service key, and returns a processing result to mobile application A or mobile application B in the rich execution environment, so as to be used to complete the service flow of mobile application A or mobile application B in the rich execution environment. Compared with the conventional security application authentication, background (server side) docking costs can be avoided when service keys of mobile applications in a rich execution environment are securely downloaded to a shared security application, and a lightweight secure computing solution based on a secure element is provided for mobile applications in the rich execution environment having common secure computing requirements.

[00102] Embodiment 6

[00103] As shown in FIG. 10, the management server of the shared security application distributes a public key thereof to the management server of mobile application A and the management server of mobile application B; the management server of mobile application A generates an application key of mobile application A, the management server of mobile application B generates an application key of mobile application B, and the application keys include communication security protection keys and service keys; the management server of mobile application A and the management server of mobile application B respectively encrypt application keys to be written to the shared security application using a public key of the management server of the shared security

application, and deliver the application keys to mobile application A in the rich execution environment of the mobile terminal and mobile application B in the trusted execution environment; mobile application A in the rich execution environment and mobile application B in the trusted execution environment store
5 respective communication security protection keys; mobile application A in the rich execution environment and mobile application B in the trusted execution environment respectively invoke corresponding interfaces provided by a shared security application proxy in the trusted execution environment to import
10 respective application keys encrypted and protected using the management server of the shared security application; the shared security application proxy in the trusted execution environment sends the imported encrypted application keys (for example, the application key of mobile application A and the application key of mobile application B) to the management server of the shared security application; the management server of the shared security application performs
15 decryption using a private key thereof to obtain the application keys of the mobile applications (for example, the application key of mobile application A and the application key of mobile application B) to be written to the shared security application; the management server of the shared security application delivers the application keys of the mobile applications (for example, the application key of
20 mobile application A and the application key of mobile application B) to the shared security application in the secure element of the mobile terminal; the shared security application securely stores the application keys of the mobile applications (for example, the application key of mobile application A and the application key of mobile application B), wherein the application key of mobile
25 application A and the application key of mobile application B are stored while being isolated from each other, respectively waiting for responding to a service request of the mobile application in the rich execution environment or a service request of the mobile application in the trusted execution environment. In this implementation, mobile application A represents a type of mobile applications in
30 the rich execution environment, and mobile application B represents a type of

mobile applications in the trusted execution environment.

[00104] Mobile application A in the rich execution environment or mobile application B in the trusted execution environment initiates a service operation request, and mobile application A or mobile application B encrypts, using the communication protection key, service data to be processed using the service key; 5 mobile application A or mobile application B sends the encrypted service data to the shared security application by invoking the interface provided by the shared security application proxy in the trusted execution environment; and the shared security application decrypts the service data using the communication security protection key of the corresponding mobile application, then performs processing using the service key, and returns a processing result to mobile application A in the rich execution environment or mobile application B in the trusted execution environment, so as to be used to complete the service flow of mobile application A in the rich execution environment or mobile application B in the trusted 10 execution environment. Compared with the conventional security application authentication, background (server side) docking costs can be avoided when service keys of mobile applications in a rich execution environment or mobile applications in a trusted execution environment are securely downloaded to a shared security application, and a lightweight secure computing solution based on a secure element is provided for mobile applications in the rich execution 20 environment or mobile applications in the trusted execution environment having common secure computing requirements.

[00105] Embodiment 7

[00106] As shown in FIG. 11, the management server of the shared security application distributes a public key thereof to the management server of 25 mobile application A and the management server of mobile application B; the management server of mobile application A generates an application key of mobile application A, the management server of mobile application B generates an application key of mobile application B, and the application keys include 30 communication security protection keys and service keys; the management server

of mobile application A and the management server of mobile application B respectively encrypt application keys to be written to the shared security application using a public key of the management server of the shared security application, and deliver the application keys to mobile application A in the rich execution environment of the mobile terminal and mobile application B in the trusted execution environment; mobile application A in the rich execution environment and mobile application B in the trusted execution environment store respective communication security protection keys; mobile application A in the rich execution environment invokes a corresponding interface provided by a shared security application proxy C in the rich execution environment, and mobile application B in the trusted execution environment invokes a corresponding interface provided by a shared security application proxy D in the trusted execution environment, to import respective application keys encrypted and protected using the management server of the shared security application; shared security application proxy C in the rich execution environment and shared security application proxy D in the trusted execution environment respectively send the imported encrypted application keys (for example, the application key of mobile application A and the application key of mobile application B) to the management server of the shared security application; the management server of the shared security application performs decryption using a private key thereof to obtain the application keys of the mobile applications (for example, the application key of mobile application A and the application key of mobile application B) to be written to the shared security application; the management server of the shared security application delivers the application keys of the mobile applications (for example, the application key of mobile application A and the application key of mobile application B) to the shared security application in the secure element of the mobile terminal; the shared security application securely stores the application keys of the mobile applications (for example, the application key of mobile application A and the application key of mobile application B), wherein the application key of mobile application A and the

application key of mobile application B are stored while being isolated from each other, waiting for responding to a service request of mobile application A in the rich execution environment or a service request of mobile application B in the trusted execution environment. In this implementation, mobile application A
5 represents a type of mobile applications in the rich execution environment, and mobile application B represents a type of mobile applications in the trusted execution environment.

[00107] Mobile application A in the rich execution environment or mobile application B in the trusted execution environment initiates a service operation
10 request, and mobile application A or mobile application B encrypts, using the communication protection key, service data to be processed using the service key; mobile application A sends the encrypted service data to the shared security application by invoking the interface provided by shared security application proxy C in the rich execution environment, mobile application B sends the
15 encrypted service data to the shared security application by invoking the interface provided by shared security application proxy D in the trusted execution environment; and the shared security application decrypts the service data using the communication security protection key of the corresponding mobile application, then performs processing using the service key, and returns a
20 processing result to mobile application A in the rich execution environment or mobile application B in the trusted execution environment, so as to be used to complete the service flow of mobile application A in the rich execution environment or mobile application B in the trusted execution environment.

Compared with the conventional security application authentication, background
25 (server side) docking costs can be avoided when service keys of mobile applications in a rich execution environment or mobile applications in a trusted execution environment are securely downloaded to a shared security application through respective shared security application proxies, and a lightweight secure computing solution based on a secure element is provided for mobile applications
30 in the rich execution environment or mobile applications in the trusted execution

environment having common secure computing requirements.

[00108] An electronic device includes a memory, a processor, and a computer program stored on the memory and executable by the processor, wherein the processor, when executing the program, implements the key transfer method based on a shared security application described above. It should be understood that the electronic device may include a separate mobile terminal or a combination of a mobile terminal and a server side.

[00109] A computer-readable storage medium comprises a computer program stored thereon, wherein the computer program, when executed by a processor, implements the key transfer method based on a shared security application described above. It should be understood that the computer program is stored in an individual mobile terminal or a combination of a mobile terminal and a server side.

[00110] One or a plurality of embodiments of the present specification provide a key transfer method based on a shared security application, wherein a shared security application placed in a secure element stores application keys invoked by a plurality of mobile applications, and meanwhile, the shared security application provides security isolation protection for an independent and dedicated application key of each mobile application. The one or plurality of embodiments of the present specification further relate to a key transfer system based on a shared security application, an electronic device, and a storage medium. The one or plurality of embodiments of the present specification provide a novel security application solution architecture, especially for mobile applications having common secure computing requirements, the one or plurality of embodiments of the present specification employ a lightweight secure computing solution based on a secure element, so that background docking costs can be avoided when service keys of different applications are securely downloaded to a shared security application; meanwhile, communication security protection keys are combined with service keys, so that flow processing on the service level can be implemented through the service keys while secure

communication between mobile applications and the shared security application can be implemented through the communication security protection keys.

Especially for mobile applications that need to realize simple and universal secure computing requirements through a secure element, it is unnecessary to

5 independently develop an application software development kit and download, install, and personalize the kit to the secure element, thereby greatly reducing the costs of secure computing performed by the mobile applications using the secure element.

[00111] The specific embodiments of the present specification have been
10 described above. Other embodiments fall within the scope of the appended claims. In some cases, the actions or steps recited in the claims may be performed in a sequence different from the order in the embodiments and may still achieve the desired result. Further, the processes depicted in the drawings do not necessarily require the particular order or sequential order shown to achieve the
15 desired result. In certain implementations, multitasking and parallel processing are also possible or may be advantageous.

[00112] The embodiments in the present specification are described in a progressive manner, and for identical or similar parts between different embodiments, reference may be made to each other so that each of the
20 embodiments focuses on differences from other embodiments. In particular, since the apparatus, electronic device, and non-volatile computer storage medium embodiments are basically similar to the method embodiments, the description thereof is relatively brief. For related parts, reference may be made to part of the description of the method embodiments.

25 [00113] The apparatus, the electronic device, and the non-volatile computer storage medium provided in the embodiments of the present specification correspond to the method. Therefore, the apparatus, the electronic device, and the non-volatile computer storage medium also have beneficial technical effects similar to those of the corresponding method. As the beneficial
30 technical effects of the method have been illustrated in detail previously, the

beneficial technical effects of the corresponding apparatus, electronic device, and non-volatile computer storage medium will not be described herein again.

[00114] In the 1990s, whether a technical improvement is a hardware improvement (for example, an improvement on a circuit structure such as a diode, a transistor, or a switch) or a software improvement (an improvement on a method procedure) can be differentiated clearly. However, along with the development of technologies, many improvements on method procedures nowadays can be regarded as direct improvements on hardware circuit structures. Almost all the designers program an improved method procedure into a hardware circuit to obtain a corresponding hardware circuit structure. Therefore, it cannot be said that an improvement on a method procedure cannot be implemented by a hardware entity module. For example, a programmable logic device (PLD) (for example, a field programmable gate array (FPGA)) is such an integrated circuit, and a logical function of the PLD is determined by a user through device programming. The designers may program by themselves to "integrate" a digital system with a PLD, and there is no need to ask a chip manufacturer to design and manufacture an application specific integrated circuit chip. Furthermore, instead of manually manufacturing an integrated circuit chip, such programming is often implemented by "logic compiler" software, which is similar to a software compiler used in program development and writing. Original code before compiling also needs to be written in a particular programming language, which is referred to as a hardware description language (HDL). There is not just one, but many types of HDLs, for example, ABEL (Advanced Boolean Expression Language), AHDL (Altera Hardware Description Language), Confluence, CUPL (Cornell University Programming Language), HDCal, JHDL (Java Hardware Description Language), Lava, Lola, MyHDL, PALASM, and RHDL (Ruby Hardware Description Language), among which VHDL (Very-High-Speed Integrated Circuit Hardware Description Language) and Verilog are most commonly used nowadays. Those skilled in the art should also understand that a hardware circuit for implementing a logic method procedure may be easily

obtained only by slightly logically programming the method procedure using the above several hardware description languages and programming the method procedure into an integrated circuit.

5 **[00115]** A controller may be implemented in any suitable manner. For example, the controller may be in the form of a microprocessor or processor and a computer-readable medium that stores computer-readable program code (for example, software or firmware) executable by the (micro)processor, logic gates, switches, an application specific integrated circuit (ASIC), a programmable logic controller, and an embedded microcontroller. The examples of the controller
10 include, but are not limited to the following microcontrollers: ARC 625D, Atmel AT91SAM, Microchip PIC18F26K20, and Silicone Labs C8051F320. The memory controller may also be implemented as part of the control logic of the memory. Those skilled in the art also know that, in addition to implementing a controller by pure computer-readable program code, with a logic programming of
15 method or steps, the controller may implement the same function in the form of a logic gate, a switch, an application specific integrated circuit, a programmable logic controller, an embedded microcontroller, or the like. Therefore, such a controller may be regarded as a hardware component, and an apparatus included therein for implementing various functions may be regarded as an internal
20 structure of the hardware component. Or, the apparatus for implementing various functions can even be regarded as both a software module for implementing a method and a structure in a hardware component.

[00116] The system, apparatus, module, or unit illustrated in the aforementioned embodiments may be specifically implemented by a computer
25 chip or an entity, or a product having a certain function. A typical implementation device is a computer. Specifically, the computer may be, for example, a personal computer, a laptop computer, a cell phone, a camera phone, a smart phone, a personal digital assistant, a media player, a navigation device, an email device, a game console, a tablet computer, a wearable device, or a
30 combination of any of these devices.

[00117] For the sake of convenient description, the above apparatus is functionally divided into various units which are separately described. Certainly, during implementation of one or a plurality of embodiments of the present specification, the functions of various units may be implemented in one or a plurality of instances of software and/or hardware.

[00118] Those skilled in the art should understand that the embodiments of the present specification may be provided as a method, a system, or a computer program product. Therefore, the embodiments of the present specification may be in the form of a full hardware embodiment, a full software embodiment, or an embodiment combining software and hardware aspects. Moreover, the embodiments of the present specification may be in the form of a computer program product implemented on one or a plurality of computer-usable storage media (including, but not limited to, a magnetic disk memory, a CD-ROM, an optical memory, and so on) containing computer-usable program code therein.

[00119] The present specification is described with reference to the flowcharts and/or block diagrams of the method, the device (system), and the computer program product according to the embodiments of the present specification. It should be understood that each flow and/or block in the flowcharts and/or block diagrams and a combination of the flows and/or blocks in the flowcharts and/or block diagrams may be implemented by computer program instructions. These computer program instructions may be provided to a processor of a general-purpose computer, a special-purpose computer, an embedded processor, or any other programmable data processing device to produce a machine, so that instructions executed by the processor of the computer or other programmable data processing device generate an apparatus for implementing a specified function in one or a plurality of procedures in the flowcharts and/or one or a plurality of blocks in the block diagrams.

[00120] These computer program instructions may also be stored in a computer-readable memory that can direct a computer or any other programmable data processing device to operate in a particular manner, so that the instructions

stored in the computer-readable memory produce an article of manufacture including an instruction apparatus, the instruction apparatus implementing a specified function in one or a plurality of procedures in the flowcharts and/or one or a plurality of blocks in the block diagrams.

5 **[00121]** These computer program instructions may also be loaded onto a computer or any other programmable data processing device so that a series of operational steps are performed on the computer or other programmable device to produce computer-implemented processing, and thus the instructions executed on the computer or other programmable device provide the steps for implementing a
10 specified function in one or a plurality of procedures in the flowcharts and/or one or a plurality of blocks in the block diagrams.

[00122] In a typical configuration, the computing device includes one or a plurality of processors (CPUs), input/output interfaces, network interfaces, and memories.

15 **[00123]** The memory may include a computer-readable medium in the form of a non-permanent memory, a random access memory (RAM) and/or a non-volatile memory or the like, such as a read-only memory (ROM) or a flash memory (flash RAM). The memory is an example of a computer-readable medium.

20 **[00124]** The computer-readable medium includes permanent and non-permanent, movable and non-movable media that can achieve information storage by means of any methods or techniques. The information may be computer-readable instructions, data structures, modules of programs or other data. Examples of a storage medium of a computer include, but are not limited to, a
25 phase change memory (PRAM), a static random access memory (SRAM), a dynamic random access memory (DRAM), other types of random access memories (RAMs), a read-only memory (ROM), an electrically erasable programmable read-only memory (EEPROM), a flash memory or other memory technologies, a compact disk read-only memory (CD-ROM), a digital versatile
30 disc (DVD) or other optical storages, a cassette tape, a magnetic tape/magnetic

disk storage or other magnetic storage devices, or any other non-transmission medium, and can be used to store information accessible by a computing device. According to the definitions herein, the computer-readable medium does not include transitory computer-readable media (transitory media), such as a modulated data signal and a carrier wave.

[00125] It should further be noted that the term "include," "comprise," or any other variation thereof is intended to encompass a non-exclusive inclusion, so that a process, method, commodity, or device that includes a series of elements includes not only those elements but also other elements not explicitly listed, or elements that are inherent to such a process, method, commodity, or device. In the absence of more restrictions, the element defined by the statement "including one ..." does not exclude other identical elements in the process, method, commodity or device that includes the element.

[00126] The present specification may be described in a general context of computer-executable instructions executed by a computer, such as a program module. Typically, the program modules include routines, programs, objects, components, data structures, and the like that execute specific tasks or implement specific abstract data types. The present specification may also be implemented in distributed computing environments. In the distributed computing environments, tasks are executed by remote processing devices that are connected by a communication network. In a distributed computing environment, the program module may be located in local and remote computer storage media including storage devices.

[00127] The embodiments in the present specification are described in a progressive manner, and for identical or similar parts between different embodiments, reference may be made to each other so that each of the embodiments focuses on differences from other embodiments. Especially, the system embodiment is described relatively briefly because it is substantially similar to the method embodiments, and for related parts, reference may be made to the method embodiments.

[00128] The above descriptions are merely the embodiments of the present specification, and are not intended to limit one or a plurality of embodiments of the present specification. For those skilled in the art, one or a plurality of embodiments of the present specification may have various alterations and changes. Any modifications, equivalent substitutions, improvements and the like made within the spirit and the principle of the one or plurality of embodiments of the present specification shall fall within the scope of the claims of the one or plurality of embodiments of the present specification.

What Is Claimed Is:

1 1. A computer executed method, comprising:
2 receiving, by an application executing on a terminal device, an application
3 key comprising at least a service key from a management server of the
4 application;
5 forwarding the application key to a management server of a shared
6 security application residing in a secure element in the terminal device, thereby
7 facilitating the management server of the shared security application to deliver the
8 application key to the shared security application; and
9 invoking, by the application, the application key stored in the shared
10 security application to perform services associated with the application, wherein
11 the application key is isolated from other application keys associated with other
12 applications stored in the shared security application.

1 2. The method according to claim 1, wherein the application key
2 further comprises a communication security protection key, and wherein the
3 method further comprises:
4 encrypting, by the application, service data associated with the services
5 using the communication security protection key; and
6 sending the encrypted service data to the shared security application.

1 3. The method according to claim 2, further comprising:
2 decrypting, by the shared security application, the encrypted service data;
3 processing, by the shared security application, the service data using the
4 service key;
5 encrypting, by the shared security application, the processed service data
6 using the communication security protection key; and

7 returning, by the shared security application, the encrypted processed
8 service data to the application.

1 4. The method according to claim 1, wherein receiving the
2 application key from the management server of the application comprises
3 receiving the application key that is encrypted by the management server of the
4 application using a public key of the management server of the shared security
5 application, and wherein the public key of the management server of the shared
6 security application is distributed to the management server of the application by
7 the management server of the shared security application.

1 5. The method according to claim 4, wherein forwarding the
2 application key comprises:
3 invoking, by the application, a corresponding interface provided by a
4 shared security application proxy on the terminal device external of the secure
5 element;
6 sending, by the application via the corresponding interface and the
7 security application proxy, the encrypted application key to the management
8 server of the shared security application;
9 performing, by the management server of the shared security application,
10 decryption using a private key corresponding to the public key to obtain the
11 application key; and
12 delivering, by the management server of the shared security application,
13 the application key to the shared security application in the secure element.

1 6. The method according to claim 5, wherein the shared security
2 application proxy is executed in a rich execution environment or a trusted
3 execution environment of the terminal device.

1 7. The method according to claim 5, wherein the public key and the
2 private key of the management server of the shared security application form an
3 asymmetric key pair, and wherein an encryption or decryption operation based on
4 the asymmetric key pair comprises one of the following operations: Rivest-
5 Shamir-Adleman (RSA), Elgamal, knapsack, Rabin, Diffie-Hellman (D-H), and
6 elliptic-curve cryptography (ECC).

1 8. The method according to claim 1, wherein the application is
2 executed in a rich execution environment or a trusted execution environment of
3 the terminal device.

1 9. A computer system, comprising:
2 a processor;
3 a storage device coupled to the processor and storing instructions, which
4 when executed by the processor cause the processor to perform a method, the
5 method comprising:
6 receiving, by an application executing on a terminal device, an
7 application key comprising at least a service key from a management
8 server of the application;
9 forwarding the application key to a management server of a shared
10 security application residing in a secure element in the terminal device,
11 thereby facilitating the management server of the shared security
12 application to deliver the application key to the shared security
13 application; and
14 invoking, by the application, the application key stored in the
15 shared security application to perform services associated with the
16 application, wherein the application key is isolated from other application

17 keys associated with other applications stored in the shared security
18 application.

1 10. The computer system according to claim 9, wherein the application
2 key further comprises a communication security protection key, and wherein the
3 method further comprises:

4 encrypting, by the application, service data associated with the services
5 using the communication security protection key; and
6 sending the encrypted service data to the shared security application.

1 11. The computer system according to claim 10, wherein the method
2 further comprises:

3 decrypting, by the shared security application, the encrypted service data;
4 processing, by the shared security application, the service data using the
5 service key;

6 encrypting, by the shared security application, the processed service data
7 using the communication security protection key; and

8 returning, by the shared security application, the encrypted processed
9 service data to the application.

1 12. The computer system according to claim 9, wherein receiving the
2 application key from the management server of the application comprises
3 receiving the application key that is encrypted by the management server of the
4 application using a public key of the management server of the shared security
5 application, and wherein the public key of the management server of the shared
6 security application is distributed to the management server of the application by
7 the management server of the shared security application.

1 13. The computer system according to claim 12, wherein forwarding
2 the application key comprises:
3 invoking, by the application, a corresponding interface provided by a
4 shared security application proxy on the terminal device external of the secure
5 element;
6 sending, by the application via the corresponding interface and the
7 security application proxy, the encrypted application key to the management
8 server of the shared security application;
9 performing, by the management server of the shared security application,
10 decryption using a private key corresponding to the public key to obtain the
11 application key; and
12 delivering, by the management server of the shared security application,
13 the application key to the shared security application in the secure element.

1 14. The computer system according to claim 13, wherein the shared
2 security application proxy is executed in a rich execution environment or a trusted
3 execution environment of the terminal device.

1 15. The computer system according to claim 13, wherein the public
2 key and the private key of the management server of the shared security
3 application form an asymmetric key pair, and wherein an encryption or decryption
4 operation based on the asymmetric key pair comprises one of the following
5 operations: Rivest-Shamir-Adleman (RSA), Elgamal, knapsack, Rabin, Diffie-
6 Hellman (D-H), and elliptic-curve cryptography (ECC).

1 16. The computer system according to claim 9, wherein the application
2 is executed in a rich execution environment or a trusted execution environment of
3 the terminal device.

1 17. A non-transitory computer-readable storage medium storing
2 instructions that when executed by a computer cause the computer to perform a
3 method, the method comprising:
4 receiving, by an application executing on a terminal device, an application
5 key comprising at least a service key from a management server of the
6 application;
7 forwarding the application key to a management server of a shared
8 security application residing in a secure element in the terminal device, thereby
9 facilitating the management server of the shared security application to deliver the
10 application key to the shared security application; and
11 invoking, by the application, the application key stored in the shared
12 security application to perform services associated with the application, wherein
13 the application key is isolated from other application keys associated with other
14 applications stored in the shared security application.

1 18. The non-transitory computer-readable storage medium according
2 to claim 17, wherein the application key further comprises a communication
3 security protection key, and wherein the method further comprises:
4 encrypting, by the application, service data associated with the services
5 using the communication security protection key; and
6 sending the encrypted service data to the shared security application.

1 19. The non-transitory computer-readable storage medium according
2 to claim 18, wherein the method further comprises:
3 decrypting, by the shared security application, the encrypted service data;
4 processing, by the shared security application, the service data using the
5 service key;
6 encrypting, by the shared security application, the processed service data

7 using the communication security protection key; and
8 returning, by the shared security application, the encrypted processed
9 service data to the application.

1 20. The non-transitory computer-readable storage medium according
2 to claim 17, wherein receiving the application key from the management server of
3 the application comprises receiving the application key that is encrypted by the
4 management server of the application using a public key of the management
5 server of the shared security application, and wherein the public key of the
6 management server of the shared security application is distributed to the
7 management server of the application by the management server of the shared
8 security application.

1 21. The non-transitory computer-readable storage medium according
2 to claim 20, wherein forwarding the application key comprises:
3 invoking, by the application, a corresponding interface provided by a
4 shared security application proxy on the terminal device external of the secure
5 element;
6 sending, by the application via the corresponding interface and the
7 security application proxy, the encrypted application key to the management
8 server of the shared security application;
9 performing, by the management server of the shared security application,
10 decryption using a private key corresponding to the public key to obtain the
11 application key; and
12 delivering, by the management server of the shared security application,
13 the application key to the shared security application in the secure element.

1 22. The non-transitory computer-readable storage medium according
2 to claim 21, wherein the shared security application proxy is executed in a rich
3 execution environment or a trusted execution environment of the terminal device.

1 23. The non-transitory computer-readable storage medium according
2 to claim 21, wherein the public key and the private key of the management server
3 of the shared security application form an asymmetric key pair, and wherein an
4 encryption or decryption operation based on the asymmetric key pair comprises
5 one of the following operations: Rivest-Shamir-Adleman (RSA), Elgamal,
6 knapsack, Rabin, Diffie-Hellman (D-H), and elliptic-curve cryptography (ECC).

1 24. The non-transitory computer-readable storage medium according
2 to claim 17, wherein the application is executed in a rich execution environment
3 or a trusted execution environment of the terminal device.