TWO-DIMENSIONAL CODE GENERATION METHOD AND DEVICE, AND TWO-DIMENSIONAL CODE RECOGNITION METHOD AND DEVICE

CLAIM OF PRIORITY

This application claims priority to Chinese Patent Application No. 201710078901.6 filed on February 14, 2017, the entire contents of which are hereby incorporated by reference.

TECHNICAL FIELD

The present application relates to the field of graphic image technologies, and in particular, to a two-dimensional code generation method and device, and a two-dimensional code recognition method and device.

BACKGROUND ART

A two-dimensional code, also known as a two-dimensional barcode, records data symbol information by using a bar/space alternating graph formed by a particular type of geometric graphs distributed on a plane (in a two-dimension direction) according to a particular rule. The two-dimensional code has features such as a large information capacity, a wide coding range, strong error tolerance, and high decoding reliability, and further has advantages such as low costs and easy manufacturing. Therefore, two-dimensional codes are widely applied in people's life.

With development of the Internet and popularization of mobile terminals, two-dimensional codes can be seen everywhere in daily life. For example, a merchant may post a payment two-dimensional code at a checkout, and a user can scan the payment two-dimensional code by using a two-dimensional code scanning function in an application. Or, in commodity promotion, a merchant may post an app downloading two-dimensional code at relatively crowded places (such as a metro or a shopping mall), to attract people passing by to scan the two-dimensional code and download the application. Therefore, merchants or third parties need to provide authentic and valid two-dimensional codes for users, and the users need to effectively verify these two-dimensional codes.

When studying and practicing the prior art, the inventor of the present invention finds that it is hard for people to judge the authenticity of a two-dimensional code with naked eyes, and during two-dimensional code recognition, a scanning unit in a mobile phone application will directly recognize various two-dimensional codes without any verification. It can be seen that a technical solution for generating a two-dimensional code so as to guarantee the authenticity and validity of the two-dimensional code is currently demanded, and correspondingly, a technical solution for judging the authenticity of a two-dimensional code during recognition of the two-dimensional code is also demanded.

The information above is merely presented as background information to facilitate

understanding of the present disclosure. It is not yet decided or stated whether any of the information above can be applied as the prior art for the present disclosure.

SUMMARY OF THE INVENTION

A main objective of the present invention is to provide a two-dimensional code generation method and device, and a two-dimensional code recognition method and device, so as to solve the foregoing problem.

An aspect of the present invention provides a two-dimensional code generation method, including: sending, by a first server, a two-dimensional code generation request to a second server, the request including information for generating a two-dimensional code; making, by the second server according to the received two-dimensional code generation request, a digital signature on the information for generating the two-dimensional code; sending the two-dimensional code information including the digital signature to the first server; and generating, by the first server, a two-dimensional code by using the two-dimensional code information.

An aspect of the present invention provides a two-dimensional code generation method, where the method is executed by a first server and includes: sending a two-dimensional code generation request to a second server, the request including information for generating a two-dimensional code; receiving, from the second server, two-dimensional code information including a digital signature generated by the second server; and generating a two-dimensional code by using the two-dimensional code information.

An aspect of the present invention provides a two-dimensional code generation device, including: a sending unit configured to send a two-dimensional code generation request to a second server, the request including information for generating a two-dimensional code; a receiving unit configured to receive, from the second server, two-dimensional code information including a digital signature generated by the second server; and a generation unit configured to generate a two-dimensional code by using the two-dimensional code information.

An aspect of the present invention provides a two-dimensional code generation method, where the method is executed by a second server and includes: receiving a two-dimensional code generation request from a first server, the request including information for generating a two-dimensional code; making a digital signature on the information for generating the two-dimensional code; and sending the two-dimensional code information including the digital signature to the first server.

An aspect of the present invention provides a two-dimensional code generation apparatus, including: a receiving unit configured to receive a two-dimensional code generation request from a first server, the request including information for generating a two-dimensional code; a processing unit configured to make a digital signature on the information for generating the two-dimensional code; and a sending unit configured to send the two-dimensional code

information including the digital signature to the first server.

An aspect of the present invention provides a two-dimensional code generation system. The system includes a first server and a second server. The first server includes: a sending unit configured to send a two-dimensional code generation request to the second server, the request including information for generating a two-dimensional code; a receiving unit configured to receive, from the second server, two-dimensional code information including a digital signature generated by the first server; and a generation unit configured to generate a two-dimensional code by using the two-dimensional code information. The second server includes: a receiving unit configured to receive the request from the first server; a processing unit configured to make the digital signature on the information for generating the two-dimensional code; and a sending unit configured to send the two-dimensional code information including the digital signature to the first server.

Another aspect of the present invention provides a two-dimensional code recognition method, including: starting an application corresponding to a server that makes a digital signature on information for generating a two-dimensional code; scanning the two-dimensional code by using a scanning unit in the application; verifying the scanned two-dimensional code by using a public key; and recognizing information corresponding to the scanned two-dimensional code if the verification succeeds.

Another aspect of the present invention provides a two-dimensional code recognition device, including: a starting unit configured to start an application corresponding to a server that makes a digital signature on information for generating a two-dimensional code; a scanning unit configured to scan the two-dimensional code by using a scanning unit in the application; a verification unit configured to verify the scanned two-dimensional code by using a public key; and a recognition unit configured to recognize information corresponding to the scanned two-dimensional code if the verification performed by the verification unit succeeds.

Compared with the prior art, in embodiments according to the present invention, a two-dimensional code is generated by means of a synergistic effect between a server and another server, so that the security and effectiveness of the two-dimensional code are enhanced, and in a two-dimensional code recognition process, it is convenient to recognize whether the two-dimensional code is tampered. In addition, according to another embodiment of the present invention, information corresponding to a two-dimensional code is recognized through a specific application by using a public key, so that authenticity judgment is added in the two-dimensional code recognition process, preventing the two-dimensional code from being tampered.

BRIEF DESCRIPTION OF THE DRAWINGS

The accompanying drawings described herein are used to provide further comprehension of the present application, and constitute a part of the present application.

Schematic embodiments of the present application and the description thereof are used for illustrating the present application, but do not constitute any improper limitation on the present application. In the accompanying drawings:

FIG. 1 is a diagram of a scenario where a two-dimensional code generation method according to an embodiment of the present invention is applied;

FIG. 2 is a schematic diagram of interaction processing between a server 110 and a server 120 shown in FIG. 1;

FIG. 3 is a schematic diagram of recognition of a two-dimensional code according to an embodiment of the present invention;

FIG. 4 is a flowchart of a two-dimensional code generation method executed by a first server according to an embodiment of the present invention;

FIG. 5 is a block diagram of a two-dimensional code generation device that executes the two-dimensional code generation method shown in FIG. 4;

FIG. 6 is a flowchart of a two-dimensional code generation method executed by a second server according to an embodiment of the present invention;

FIG. 7 is a block diagram of a two-dimensional code generation device that executes the two-dimensional code generation method shown in FIG. 6;

FIG. 8 is a flowchart of a two-dimensional code recognition method according to an embodiment of the present invention;

FIG. 9 is a block diagram of a two-dimensional code recognition device that executes the two-dimensional code recognition method shown in FIG. 8 according to an embodiment of the present invention; and

FIG. 10 is a block diagram of an electronic device according to an embodiment of the present invention.

FIG. 11 is a flowchart illustrating an example of a computer-implemented method for generating a 2D code, according to an implementation of the present disclosure.

DETAILED DESCRIPTION

To make the objectives, technical solutions and advantages of the present application clearer, the technical solutions of the present application are described in further detail with reference to the specific embodiments of the present application and the corresponding drawings. Apparently, the described embodiments are merely some rather than all of the embodiments of the present application. All other embodiments derived by those of ordinary skill in the art without creative efforts belong to the protection scope of the present application.

In the following, the embodiments will be described in further detail with reference to the accompanying drawings. Identical reference numerals always represent identical elements. For clearer understanding of the present invention, terms related to the present invention will be

illustrated hereinafter.

A terminal/an electronic device generally refers to an apparatus which is used by a user in a network system and is configured to communicate with a server. A two-dimensional code generation method provided in an embodiment of the present invention may be executed by a two-dimensional code generation device. A two-dimensional code recognition method provided in an embodiment of the present invention may be executed by a two-dimensional code recognition device. The two-dimensional code generation device and the two-dimensional code recognition device each may be a terminal/an electronic device. The terminal/electronic device according to the present invention may include, but is not limited to, any of the following devices having a display unit: a personal computer (PC), a mobile apparatus (such as a cellular phone, a personal digital assistant (PDA), a digital camera, a portable game console, an MP3 player, a portable/personal multimedia player (PMP), a handheld electronic book, a tablet PC, a portable laptop PC, and a global positioning system (GPS) navigator), a smart TV, and the like.

An application (app) is software that can be directly executed by a user on a computer operating system (OS) or a mobile OS. The application may include an embedded application or a third-party application stored in a storage unit of a terminal. The embedded application refers to an application pre-installed in the terminal. For example, the embedded application may be a browser, an e-mail, or an instant messenger. The third-party applications are diversified, and refer to the following applications that are downloaded from an online market and installed on a terminal, for example, a payment application, a shopping application, an entertainment application, and so on.

FIG. 1 is a diagram of a scenario where a two-dimensional code generation method according to an embodiment of the present invention is applied.

As shown in FIG. 1, a server 110 and a server 120 are connected in a wired or wireless manner. It should be understood that, in the embodiments of the present invention, the wireless connection may be a one-to-one wireless connection, such as Bluetooth and Near Field Communication (NFC), and a wireless local area network (WiFi) may also be used in a security guaranteed scenario.

The server 110 is a server for generating a two-dimensional code. For example, the server 110 may be an application server, a website server, or the like. For example, if the server 110 is a payment application server, the application server 110 may generate a two-dimensional code about a commodity in response to a user input. Alternatively, if the server 110 is a website server, various two-dimensional codes generated by the server 110 according to requirements may be displayed on a display. The server 120 is a third-party server with an authorized certification (such as Alipay™ server).

It should be noted that both the server 110 and the server 120 are electronic devices

providing computing services, and can respond to server requests and carry out processing. Moreover, architectures of the two servers are similar to an architecture of a general-purpose computer. In the present invention, the server 110 and the server 120 provide different services.

Referring to FIG. 1, the server 110 and the server 120 carry out interaction processing, and then the server 110 generates a two-dimensional code according to information provided by the server 120. This will be described in detail below with reference to FIG. 2, and therefore is not described herein.

Next, the server 110 may display the two-dimensional code on a display, or directly outputs the two-dimensional code by using an output device (such as a printer). Then, a user may scan the two-dimensional code by using a scanning unit in a mobile terminal.

The interaction processing between the server 110 and the server 120 will be described in detail below with reference to FIG. 2.

As shown in FIG. 2, in step 210, the server 110 sends a two-dimensional code generation request to the server 120. The request may include information for generating a two-dimensional code (which is referred to as two-dimensional code information for short). For example, the two-dimensional code information may include payee account information, or the two-dimensional code information may include website information.

Optionally, before performing step 210, the server 110 sends a registration request to the server 120 in advance. The registration request may include all information related to current registration. For example, the server 110 may provide various qualification proofs thereof to the server 120, for example, a business certificate, commodity details, and so on. Next, after successfully verifying the information provided by the server 110, the server 120 completes registration of the server 110. In addition, the qualification proofs provided by the server 110 are stored in a memory of the server 120. Optionally, the qualification proofs may be stored in a remote memory. In this way, when needing to be invoked, the qualification proofs are invoked by using an address of the remote memory.

In this way, after the server 110 sends the two-dimensional code generation request to the server 120, the server 120 performs verification on the server 110 to determine whether the server 110 has been registered in the server 120. If no, the server 120 does not provide subsequent services and reminds the server 110 to carry out registration. Specifically, the server 120 may reject the two-dimensional code generation request of the server 110 and sends a registration invitation to the server 110. If the server 110 is already registered in the server 120, subsequent steps are performed.

Through the foregoing registration step, the server 120 may provide services for all apparatuses that pass verification, thereby guaranteeing the validity of the two-dimensional code synergistically generated by the server 120, and preserving evidence for use in a possible dispute

with a merchant of the server 110.

In step 220, the server 120 may make a digital signature on the two-dimensional code information on the basis of an asymmetric encryption algorithm. Specifically, the server 120 may make a digital signature on the two-dimensional code information in the two-dimensional code generation request in step 210 on the basis of the asymmetric encryption algorithm and by using a pre-generated private key (which is stored in the server 120). In this way, in a two-dimensional code recognition process, if a two-dimensional code is successfully scanned by using a public key, it indicates that the two-dimensional code is a two-dimensional code generated by the server 110 by using the server 120. If the scanning fails, it indicates that the two-dimensional code is probably a counterfeit two-dimensional code or is a two-dimensional code that is only generated by the server 110 but is not processed by using the server 120.

It should be noted that the server 120 generates different digital signatures according to different two-dimensional code information. In other words, the server 120 may provide different private keys according to different two-dimensional code information, and then makes digital signatures on the different two-dimensional code information by using respective private keys. Therefore, even for different link information about different commodities sent by a same website, digital signatures are made on the different link information by using different private keys.

The foregoing asymmetric encryption algorithm may be referred to as a public key encryption algorithm. In this algorithm, data is encrypted by using a public key, and can be decrypted only by using a corresponding private key. Alternatively, data is encrypted by using a private key, and is decrypted by using a corresponding public key. According to an exemplary embodiment, the asymmetric encryption algorithm may include an RSA public key encryption algorithm, an Elgamal encryption algorithm, a knapsack algorithm, and an elliptic curve cryptography (ECC) encryption algorithm. As the objective of the present invention is to guarantee the authenticity and effectiveness of the two-dimensional code, preferably, the RSA public key encryption algorithm may be selected to encrypt generated transaction data.

Because making a digital signature on data by using an asymmetric encryption algorithm is a common technical means in the field, detailed descriptions thereof are omitted herein. It should be noted that, although examples of the asymmetric encryption algorithm are provided above, those skilled in the art should understand that all asymmetric encryption algorithms capable of generating a digital signature can be applied here.

Next, in step 130, the server 120 sends the two-dimensional code information including the digital signature to the server 110. After receiving the two-dimensional code information including the digital signature, in step 140, the server 110 generates the two-dimensional code by using the two-dimensional code information including the digital signature. Specifically, the

two-dimensional code may be generated on the basis of a two-dimensional code generation algorithm and by using the two-dimensional code information as well as the digital signature. Preferably, the two-dimensional code generation algorithm includes a Quickresponse code (QR code) algorithm.

Because generating a two-dimensional code by using a two-dimensional code generation algorithm is a common technical means in the field, detailed descriptions thereof are omitted herein. Those skilled in the art should understand that all two-dimensional code generation algorithms capable of generating a two-dimensional code can be applied here.

Next, a schematic diagram of verification on a two-dimensional code will be described in detail below with reference to FIG. 3.

As shown in FIG. 3, a user may verify, by using a mobile terminal, an electronic two-dimensional code displayed on a display or a two-dimensional code output by an output device.

The user may start an application corresponding to the server 120. For example, the user may start, in the mobile terminal, the Alipay™ application corresponding to the Alipay™ server. Subsequently, the user scans the two-dimensional code displayed on the display or the printed two-dimensional code by using a scanning unit in the application.

It should be noted that, in the two-dimensional code recognition process, the user should use the scanning unit in the application corresponding to the server 120, because the application corresponding to the server 120 includes or can acquire a public key corresponding to the two-dimensional code.

Optionally, after the two-dimensional code is scanned by using the scanning unit in the application, the public key may be acquired from the server 120 or a remote storage device that stores the public key. Subsequently, verification is performed on the scanned two-dimensional code by using the public key. If the verification succeeds, the two-dimensional code can be successfully recognized so as to perform the next operation. In other words, an operation interface corresponding to the two-dimensional code may be displayed on a display interface of the mobile terminal according to a recognition result of the two-dimensional code. For example, if the recognition result of the two-dimensional code is a link about Alipay™, the mobile terminal invokes the Alipay™ application, thereby displaying an operation interface about Alipay™ on a display unit. If the recognition result of the two-dimensional code is a link about WeChat, the mobile terminal invokes the WeChat application, thereby displaying an operation interface about WeChat on the display unit. If the recognition result of the two-dimensional code is a link about a website, the mobile terminal invokes a browser application, thereby displaying an operation interface about the browser on the display unit.

If the verification is unsuccessful, it indicates that the two-dimensional code is tampered. In addition, such a case is also possible: an operator who tampers the two-dimensional code is

also registered with the server 120, and a two-dimensional code for tampering the original two-dimensional code is also synergistically generated by the server 120. For example, a merchant B tears down a two-dimensional code posted outside by a merchant A, and posts his own two-dimensional code. In this case, the two-dimensional code of the merchant B can be successfully recognized by using the scanning unit in the application corresponding to the server 120. In order to solve this problem, the foregoing scanning process may be recorded. In this way, an operator who tampers the two-dimensional code can be determined according to the scanning record, and the identity of the operator can be determined according to information provided by the operator during registration.

FIG. 4 is a flowchart of a two-dimensional code generation method executed by a first server according to an embodiment of the present invention.

In step S410, a two-dimensional code generation request is sent to a second server, the request including information for generating a two-dimensional code.

In step S420, two-dimensional code information including a digital signature generated by the second server is received from the second server.

In step S430, the two-dimensional code is generated by using the two-dimensional code information.

In an alternative embodiment, before the step of sending a two-dimensional code generation request to a second server, the method further includes: sending a registration request to the second server, the registration request including all information related to current registration.

FIG. 5 is a block diagram of a two-dimensional code generation device (i.e., the first server) that executes the two-dimensional code generation method shown in FIG. 4.

Those skilled in the art may understand that the structure of the two-dimensional code generation device shown in FIG. 5 does not limit the electronic device of the present invention. The electronic device may include more or fewer components than those shown in the figure, or some components may be combined, or a different component layout may be employed.

As shown in FIG. 5, the two-dimensional code generation device may include a sending unit 510, a receiving unit 520 and a generation unit 530.

The sending unit 510 may be configured to send a two-dimensional code generation request to a second server, the request including information for generating a two-dimensional code.

The receiving unit 520 may be configured to receive, from the second server, two-dimensional code information including a digital signature generated by the second server.

The generation unit 530 may be configured to generate a two-dimensional code by using the two-dimensional code information.

In an alternative embodiment, the sending unit 510 is configured to send a registration request to the first server before sending the two-dimensional code generation request to the first server, the registration request including all information related to current registration.

FIG. 6 is a flowchart of a two-dimensional code generation method executed by a second server according to an embodiment of the present invention.

In step S610, a two-dimensional code generation request is received from a first server, the request including information for generating a two-dimensional code.

In step S620, a digital signature is made on the information for generating the two-dimensional code.

In step S630, two-dimensional code information including the digital signature is sent to the first server.

In an optional embodiment, before the step of receiving a two-dimensional code generation request from a first server, the method further includes: receiving a registration request from the first server, the registration request including all information related to current registration. Moreover, after the step of receiving a registration request from the first server, the two-dimensional code generation method may further include: completing registration of the first server according to the registration request.

FIG. 7 is a block diagram of a two-dimensional code generation device (i.e., the second server) that executes the two-dimensional code generation method shown in FIG. 6.

Those skilled in the art may understand that the structure of the two-dimensional code generation device shown in FIG. 7 does not limit the electronic device of the present invention. The electronic device may include more or fewer components than those shown in the figure, or some components may be combined, or a different component layout may be employed.

The two-dimensional code generation device may include a receiving unit 710, a processing unit 720 and a sending unit 730.

The receiving unit 710 is configured to receive a two-dimensional code generation request from a first server, the request including information for generating a two-dimensional code. In an alternative embodiment, the receiving unit is further configured to receive a registration request from the first server, the registration request including all information related to current registration.

In an alternative embodiment, the two-dimensional code generation device may include a registration unit, and the registration unit is configured to complete registration of the first server according to the registration request.

The processing unit 720 is configured to make a digital signature on the information for generating the two-dimensional code. Optionally, the processing unit 720 may be configured to make, on the basis of an asymmetric encryption algorithm and by using a private key, a digital

signature on the information for generating the two-dimensional code.

The sending unit 730 is configured to send the two-dimensional code information including the digital signature to the first server.

In an alternative embodiment, the two-dimensional code generation device further includes a determination unit. The determination unit is configured to determine whether the first server is registered in the second server. If yes, a digital signature is made according to the received two-dimensional code generation request on the information for generating the two-dimensional code. If no, no digital signature is made on the information for generating the two-dimensional code, and a registration invitation is sent to the first server.

In an alternative embodiment, according to different information for generating the two-dimensional code, different private keys are used in the step of making a digital signature on the information for generating the two-dimensional code.

In another exemplary embodiment, a two-dimensional code generation system may be provided. The system includes a two-dimensional code generation device as shown in FIG. 5 and a two-dimensional code generation device as shown in FIG. 7.

As described above, according to the two-dimensional code generation method and device provided in the embodiments of the present invention, a two-dimensional code is generated by means of a synergistic effect between a server and another server, so that the security and effectiveness of the two-dimensional code are enhanced, and in a two-dimensional code recognition process, it is convenient to recognize whether the two-dimensional code is tampered.

FIG. 8 is a flowchart of a two-dimensional code recognition method according to an embodiment of the present invention.

In step S810, an application corresponding to a server that makes a digital signature on information for generating a two-dimensional code is started.

In step S820, the two-dimensional code is scanned by using a scanning unit in the application.

In step S830, the scanned two-dimensional code is verified by using a public key.

In step S840, information corresponding to the scanned two-dimensional code is recognized if the verification succeeds.

In an alternative embodiment, the public key is obtained from the server or a remote storage device. In addition, the method may further include storing related information in the verification step.

FIG. 9 is a block diagram of a two-dimensional code recognition device that executes the two-dimensional code recognition method shown in FIG. 8 according to an embodiment of the present invention.

Those skilled in the art may understand that the structure of the two-dimensional code recognition device shown in FIG. 9 does not limit the electronic device of the present invention. The electronic device may include more or fewer components than those shown in the figure, or some components may be combined, or a different component layout may be employed.

The two-dimensional code recognition device may include a starting unit 910, a scanning unit 920, a verification unit 930 and a recognition unit 940.

The starting unit 910 may be configured to start an application corresponding to a server that makes a digital signature on information for generating a two-dimensional code.

The scanning unit 920 may be configured to scan the two-dimensional code by using a scanning unit in the application.

The verification unit 930 may be configured to verify the scanned two-dimensional code by using a public key.

The recognition unit 940 may be configured to recognize information corresponding to the scanned two-dimensional code if the verification performed by the verification succeeds.

In an alternative embodiment, the public key is obtained from the server or a remote storage device.

In an alternative embodiment, the two-dimensional code recognition device further includes a storage unit, and the storage unit is configured to store related information in the verification step.

FIG. 10 is a block diagram of an electronic device that executes the two-dimensional code generation method or two-dimensional code recognition method according to the embodiment of the present invention. Referring to FIG. 10, in a hardware level, the electronic device includes a processor, an internal bus, a network interface, a memory, and a non-volatile storage. Definitely, the electronic device may further include other hardware required by services. The processor reads a corresponding computer program from the non-volatile storage into the memory and runs the computer program, to form a webpage screenshot apparatus in a logic level. Definitely, in addition to the software implementation, the present application does not exclude other implementations, such as logical devices or a software and hardware combined manner. In other words, the following processing procedure is not limited to being executed by various logical units, but may also be executed by hardware or logical devices.

The two-dimensional code recognition method and device provided according to the embodiments of the present invention recognize, through a specific application, information corresponding to a two-dimensional code by using a public key, thus adding authenticity judgment in a two-dimensional code recognition process, and preventing the two-dimensional code from being tampered.

In the 1990s, an improvement on a technology may be obviously distinguished as an

improvement on hardware (for example, an improvement on a circuit structure such as a diode, a transistor, and a switch) or an improvement on software (an improvement on a method procedure). However, with the development of technologies, improvements of many method procedures at present may be considered as direct improvements on hardware circuit structures. Almost all designers program the improved method procedures into hardware circuits to obtain corresponding hardware circuit structures. Therefore, it is improper to assume that the improvement of a method procedure cannot be implemented by using a hardware entity module. For example, a Programmable Logic Device (PLD) (for example, a Field Programmable Gate Array (FPGA)) is such an integrated circuit whose logic functions are determined by devices programmed by a user. Designers program by themselves to "integrate" a digital system into a piece of PLD, without the need to ask a chip manufacturer to design and manufacture a dedicated integrated circuit chip. Moreover, at present, the programming is mostly implemented by using "logic compiler" software, instead of manually manufacturing an integrated circuit chip. The logic compiler software is similar to a software complier used for developing and writing a program, and original code before compiling also needs to be written by using a specific programming language, which is referred to as a Hardware Description Language (HDL). There are many types of HDLs, such as Advanced Boolean Expression Language (ABEL), Altera Hardware Description Language (AHDL), Confluence, Cornell University Programming Language (CUPL), HDCal, Java Hardware Description Language (JHDL), Lava, Lola, MyHDL, PALASM, and Ruby Hardware Description Language (RHDL), among which Very-High-Speed Integrated Circuit Hardware Description Language (VHDL) and Verilog are most commonly used now. Those skilled in the art also should know that a hardware circuit for implementing the logic method procedure may be easily obtained by slightly logically programming the method procedure using the above several hardware description languages and programming it into an integrated circuit.

A controller may be implemented in any suitable manner. For example, the controller may be in the form of, for example, a microprocessor or a processor and a computer readable medium storing computer readable program code (for example, software or firmware) executable by the (micro)processor, a logic gate, a switch, an Application Specific Integrated Circuit (ASIC), a programmable logic controller, and an embedded micro-controller. Examples of the controller include, but are not limited to, the following micro-controllers: ARC 625D, Atmel AT91SAM, Microchip PIC18F26K20, and Silicone Labs C8051F320. A memory controller may also be implemented as a part of control logic of a memory. Those skilled in the art also know that, the controller may be implemented by using pure computer readable program code, and in addition, the method steps may be logically programmed to enable the controller to implement the same function in a form of a logic gate, a switch, an application specific integrated

circuit, a programmable logic controller, an embedded microcontroller, and the like. Therefore, this type of controller may be considered as a hardware component, and apparatuses included therein for implementing various functions may also be considered as structures inside the hardware component. Or, the apparatuses used for implementing various functions may even be considered as both software modules for implementing the method and structures inside the hardware component.

The system, apparatus, module or unit illustrated in the above embodiments may be specifically implemented by using a computer 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 cellular 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 combination of any of these devices.

For ease of description, when the apparatus is described, it is divided into various units in terms of functions for respective descriptions. Definitely, when the present application is implemented, functions of the units may be implemented in the same or multiple pieces of software and/or hardware.

Those skilled in the art should understand that the embodiments of the present invention may be provided as a method, a system, or a computer program product. Therefore, the present invention may be implemented as a complete hardware embodiment, a complete software embodiment, or an embodiment combining software and hardware. Moreover, the present invention may be a computer program product implemented on one or more computer usable storage media (including, but not limited to, a magnetic disk memory, a CD-ROM, an optical memory, and the like) including computer usable program code.

The present application is described with reference to flowcharts and/or block diagrams according to the method, device (system) and computer program product according to the embodiments of the present invention. It should be understood that a computer program instruction may be used to implement each process and/or block in the flowcharts and/or block diagrams and combinations of processes and/or blocks in the flowcharts and/or block diagrams. These computer program instructions may be provided for a general-purpose computer, a special-purpose computer, an embedded processor, or a processor of any other programmable data processing device to generate a machine, so that the instructions executed by a computer or a processor of any other programmable data processing device generate an apparatus for implementing a specified function in one or more processes in the flowcharts and/or in one or more blocks in the block diagrams.

These computer program instructions may also be stored in a computer readable memory that can instruct the computer or any other programmable data processing device to

work in a particular manner, such that the instructions stored in the computer readable memory generate an artifact that includes an instruction apparatus. The instruction apparatus implements a specified function in one or more processes in the flowcharts and/or in one or more blocks in the block diagrams.

These computer program instructions may also be loaded onto a computer or another programmable data processing device, such that a series of operation steps are performed on the computer or another programmable device, thereby generating computer-implemented processing. Therefore, the instructions executed on the computer or another programmable device provide steps for implementing a specified function in one or more processes in the flowcharts and/or in one or more blocks in the block diagrams.

In a typical configuration, a computation device includes one or more central processing units (CPUs), an input/output interface, a network interface, and a memory.

The memory may include computer readable media such as a volatile memory, a Random Access Memory (RAM), and/or non-volatile memory, e.g., Read-Only Memory (ROM) or flash RAM. The memory is an example of a computer readable medium.

The computer readable medium includes non-volatile and volatile media as well as movable and non-movable media, and can implement information storage by means of any method or technology. Information may be a computer readable instruction, a data structure, and a module of a program or other data. A storage medium of a computer includes, for example, but is not limited to, a phase change memory (PRAM), a static random access memory (SRAM), a dynamic random access memory (DRAM), other types of RAMs, a 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 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 accessed by the computing device. According to the definition of this text, the computer readable medium does not include transitory media, such as a modulated data signal and a carrier.

It should be further noted that the term "include", "comprise" or other variations thereof are intended to cover non-exclusive inclusion, so that a process, method, commodity or device including a series of elements not only includes the elements, but also includes other elements not clearly listed, or further includes inherent elements of the process, method, commodity or device. In a case without any more limitations, an element defined by "including a/an…" does not exclude that the process, method, commodity or device including the element further has other identical elements.

Those skilled in the art should understand that the embodiments of the present application may be provided as a method, a system, or a computer program product. Therefore,

the present application may be implemented as a complete hardware embodiment, a complete software embodiment, or an embodiment combining software and hardware. Moreover, the present application may be in the form of a computer program product implemented on one or more computer usable storage media (including, but not limited to, a magnetic disk memory, a CD-ROM, an optical memory and the like) including computer usable program code.

The present application may be described in a common context of a computer executable instruction executed by a computer, for example, a program module. Generally, the program module includes a routine, a program, an object, an assembly, a data structure, and the like used for executing a specific task or implementing a specific abstract data type. The present application may also be implemented in distributed computing environments, and in the distributed computer environments, a task is executed by using remote processing devices connected through a communications network. In the distributed computer environment, the program module may be located in local and remote computer storage media including a storage device.

The embodiments in the specification are described progressively, identical or similar parts of the embodiments may be obtained with reference to each other, and each embodiment emphasizes a part different from other embodiments. Especially, the system embodiment is basically similar to the method embodiment, so it is described simply, and for related parts, reference may be made to the descriptions of the parts in the method embodiment.

The above descriptions are merely embodiments of the present application, and are not intended to limit the present application. For those skilled in the art, the present application may have various modifications and variations. Any modification, equivalent replacement, improvement or the like made without departing from the spirit and principle of the present application should all fall within the scope of claims of the present application.

FIG. 11 is a flowchart illustrating an example of a computer-implemented method 1100 for generating a 2D code, according to an implementation of the present disclosure. For clarity of presentation, the description that follows generally describes method 1100 in the context of the other figures in this description. However, it will be understood that method 1100 can be performed, for example, by any system, environment, software, and hardware, or a combination of systems, environments, software, and hardware, as appropriate. In some implementations, various steps of method 1100 can be run in parallel, in combination, in loops, or in any order. Method 1100 can be performed by a server that provides 2D code encryption information including digital signature, such as the second server discussed previously.

At 1102, registration information from a first server for authenticating and registering the first server is received. The registration request can include information related to registering the server. For example, the server can provide various qualification authentications, such as

business certificates or merchandize details. After the qualification authentications are verified, the server can be registered. From 1102, method 1100 proceeds to 1104.

At 1104, the first server is registered if the registration information is authenticated. From 1104, method 1100 proceeds to 1106.

At 1106, the registration information is saved. The registration information including the qualification authentications provided by the server can be stored in a local memory or a remote memory. As such, the qualification authentications can be located for future references. From 1106, method 1100 proceeds to 1108.

At 1108, a request from the first server for generating a 2D code is received, wherein the request includes information to be encoded to the 2D code. The information to be encoded using the 2D code can include account information for receiving payment or website information that can redirect the user to the corresponding website. From 1108, method 1100 proceeds to 1110.

At 1110, the information to be encoded to the 2D code is digitally signed by generating a digital signature using asymmetrical encryption. The digital signature can be generated for the 2D code information included in the 2D code generation request. In some cases, the digital signature on the 2D code information can be generated based on an asymmetric encryption algorithm and a pre-generated private key. For example, the digital signature can be generated based on an RSA encryption algorithm. If a 2D code is successfully scanned by a user computing device using a corresponding public key, it can indicate that the 2D code is generated by the first server based on the digitally signed 2D code information. If the 2D code scanning is unsuccessful, it can indicate that the 2D code is tampered with or is not generated based on the digitally signed 2D code information. In some cases, the server that provides the digital signature can check whether the server that sends the 2D code generation request is a registered server. If not, the server can reject the 2D code generation request. From 1110, method 1100 proceeds to 1112.

At 1112, the digitally signed information is sent to the first server. The first server can use the received digitally signed 2D code information to generate the 2D code. From 1112, method 1100 proceeds to 1114.

At 1114, the 2D code is decrypted after the 2D code is scanned by a user computing device. A user can use an application, software, or software development kit (SDK) provided by the second server to scan the 2D code. In some cases, after the 2D code is scanned, the second server can verify the digital signature included in the 2D code using a public key. If the digital signature is authenticated, the second server can decrypt the 2D code and send the decrypted information to the user. In some cases, the digital signature verification algorithm can be included in the application, software, or SDK provided by the second server and installed on the

user computing device. In such cases, the user can decrypt the 2D code by using the digital signature verification algorithm available on the user computing device. If the scanning is unsuccessful, the 2D code may have been tampered with. The user can be notified that the 2D code cannot be trusted. From 1114, method 1100 proceeds to 1116.

At 1116, a record that the 2D code is scanned is saved on the second server, wherein the record includes the first server that encodes the 2D code. The second server can save scanning activities of the 2D code. If the scanning is successful, but the 2D code was generated by another server misappropriating the digitally signed information, the misappropriation can be identified by the second server because the scanning activities of the 2D code are saved. After 1116, method 1100 ends.

Implementations of the subject matter and the functional operations described in this specification can be implemented in digital electronic circuitry, in tangibly embodied computer software or firmware, in computer hardware, including the structures disclosed in this specification and their structural equivalents, or in combinations of one or more of them. Software implementations of the described subject matter can be implemented as one or more computer programs, that is, one or more modules of computer program instructions encoded on a tangible, non-transitory, computer-readable medium for execution by, or to control the operation of, a computer or computer-implemented system. Alternatively, or additionally, the program instructions can be encoded in/on an artificially generated propagated signal, for example, a machine-generated electrical, optical, or electromagnetic signal that is generated to encode information for transmission to a receiver apparatus for execution by a computer or computer-implemented system. The computer-storage medium can be a machine-readable storage device, a machine-readable storage substrate, a random or serial access memory device, or a combination of computer-storage mediums. Configuring one or more computers means that the one or more computers have installed hardware, firmware, or software (or combinations of hardware, firmware, and software) so that when the software is executed by the one or more computers, particular computing operations are performed.

The term “real-time,” “real time,” “realtime,” “real (fast) time (RFT),” “near(ly) real-time (NRT),” “quasi real-time,” or similar terms (as understood by one of ordinary skill in the art), means that an action and a response are temporally proximate such that an individual perceives the action and the response occurring substantially simultaneously. For example, the time difference for a response to display (or for an initiation of a display) of data following the individual’s action to access the data can be less than 1 millisecond (ms), less than 1 second (s), or less than 5 s. While the requested data need not be displayed (or initiated for display) instantaneously, it is displayed (or initiated for display) without any intentional delay, taking into account processing limitations of a described computing system and time required to, for

example, gather, accurately measure, analyze, process, store, or transmit the data.

The terms "data processing apparatus," "computer," or "electronic computer device" (or an equivalent term as understood by one of ordinary skill in the art) refer to data processing hardware and encompass all kinds of apparatuses, devices, and machines for processing data, including by way of example, a programmable processor, a computer, or multiple processors or computers. The computer can also be, or further include special purpose logic circuitry, for example, a central processing unit (CPU), an FPGA (field programmable gate array), or an ASIC (application-specific integrated circuit). In some implementations, the computer or computer-implemented system or special purpose logic circuitry (or a combination of the computer or computer-implemented system and special purpose logic circuitry) can be hardware- or software-based (or a combination of both hardware- and software-based). The computer can optionally include code that creates an execution environment for computer programs, for example, code that constitutes processor firmware, a protocol stack, a database management system, an operating system, or a combination of execution environments. The present disclosure contemplates the use of a computer or computer-implemented system with an operating system of some type, for example LINUX, UNIX, WINDOWS, MAC OS, ANDROID, IOS, another operating system, or a combination of operating systems.

A computer program, which can also be referred to or described as a program, software, a software application, a unit, a module, a software module, a script, code, or other component can be written in any form of programming language, including compiled or interpreted languages, or declarative or procedural languages, and it can be deployed in any form, including, for example, as a stand-alone program, module, component, or subroutine, for use in a computing environment. A computer program can, but need not, correspond to a file in a file system. A program can be stored in a portion of a file that holds other programs or data, for example, one or more scripts stored in a markup language document, in a single file dedicated to the program in question, or in multiple coordinated files, for example, files that store one or more modules, sub-programs, or portions of code. A computer program can be deployed to be executed on one computer or on multiple computers that are located at one site or distributed across multiple sites and interconnected by a communication network.

While portions of the programs illustrated in the various figures can be illustrated as individual components, such as units or modules, that implement described features and functionality using various objects, methods, or other processes, the programs can instead include a number of sub-units, sub-modules, third-party services, components, libraries, and other components, as appropriate. Conversely, the features and functionality of various components can be combined into single components, as appropriate. Thresholds used to make computational determinations can be statically, dynamically, or both statically and dynamically

determined.

Described methods, processes, or logic flows represent one or more examples of functionality consistent with the present disclosure and are not intended to limit the disclosure to the described or illustrated implementations, but to be accorded the widest scope consistent with described principles and features. The described methods, processes, or logic flows can be performed by one or more programmable computers executing one or more computer programs to perform functions by operating on input data and generating output data. The methods, processes, or logic flows can also be performed by, and computers can also be implemented as, special purpose logic circuitry, for example, a CPU, an FPGA, or an ASIC.

Computers for the execution of a computer program can be based on general or special purpose microprocessors, both, or another type of CPU. Generally, a CPU will receive instructions and data from and write to a memory. The essential elements of a computer are a CPU, for performing or executing instructions, and one or more memory devices for storing instructions and data. Generally, a computer will also include, or be operatively coupled to, receive data from or transfer data to, or both, one or more mass storage devices for storing data, for example, magnetic, magneto-optical disks, or optical disks. However, a computer need not have such devices. Moreover, a computer can be embedded in another device, for example, a mobile telephone, a personal digital assistant (PDA), a mobile audio or video player, a game console, a global positioning system (GPS) receiver, or a portable memory storage device.

Non-transitory computer-readable media for storing computer program instructions and data can include all forms of permanent/non-permanent or volatile/non-volatile memory, media and memory devices, including by way of example semiconductor memory devices, for example, random access memory (RAM), read-only memory (ROM), phase change memory (PRAM), static random access memory (SRAM), dynamic random access memory (DRAM), erasable programmable read-only memory (EPROM), electrically erasable programmable read-only memory (EEPROM), and flash memory devices; magnetic devices, for example, tape, cartridges, cassettes, internal/removable disks; magneto-optical disks; and optical memory devices, for example, digital versatile/video disc (DVD), compact disc (CD)-ROM, DVD+/-R, DVD-RAM, DVD-ROM, high-definition/density (HD)-DVD, and BLU-RAY/BLU-RAY DISC (BD), and other optical memory technologies. The memory can store various objects or data, including caches, classes, frameworks, applications, modules, backup data, jobs, web pages, web page templates, data structures, database tables, repositories storing dynamic information, or other appropriate information including any parameters, variables, algorithms, instructions, rules, constraints, or references. Additionally, the memory can include other appropriate data, such as logs, policies, security or access data, or reporting files. The processor and the memory can be supplemented by, or incorporated in, special purpose logic circuitry.

To provide for interaction with a user, implementations of the subject matter described in this specification can be implemented on a computer having a display device, for example, a CRT (cathode ray tube), LCD (liquid crystal display), LED (Light Emitting Diode), or plasma monitor, for displaying information to the user and a keyboard and a pointing device, for example, a mouse, trackball, or trackpad by which the user can provide input to the computer. Input can also be provided to the computer using a touchscreen, such as a tablet computer surface with pressure sensitivity, a multi-touch screen using capacitive or electric sensing, or another type of touchscreen. Other types of devices can be used to interact with the user. For example, feedback provided to the user can be any form of sensory feedback (such as, visual, auditory, tactile, or a combination of feedback types). Input from the user can be received in any form, including acoustic, speech, or tactile input. In addition, a computer can interact with the user by sending documents to and receiving documents from a client computing device that is used by the user (for example, by sending web pages to a web browser on a user's mobile computing device in response to requests received from the web browser).

The term "graphical user interface," or "GUI," can be used in the singular or the plural to describe one or more graphical user interfaces and each of the displays of a particular graphical user interface. Therefore, a GUI can represent any graphical user interface, including but not limited to, a web browser, a touch screen, or a command line interface (CLI) that processes information and efficiently presents the information results to the user. In general, a GUI can include a number of user interface (UI) elements, some or all associated with a web browser, such as interactive fields, pull-down lists, and buttons. These and other UI elements can be related to or represent the functions of the web browser.

Implementations of the subject matter described in this specification can be implemented in a computing system that includes a back-end component, for example, as a data server, or that includes a middleware component, for example, an application server, or that includes a front-end component, for example, a client computer having a graphical user interface or a Web browser through which a user can interact with an implementation of the subject matter described in this specification, or any combination of one or more such back-end, middleware, or front-end components. The components of the system can be interconnected by any form or medium of wireline or wireless digital data communication (or a combination of data communication), for example, a communication network. Examples of communication networks include a local area network (LAN), a radio access network (RAN), a metropolitan area network (MAN), a wide area network (WAN), Worldwide Interoperability for Microwave Access (WIMAX), a wireless local area network (WLAN) using, for example, 802.11 a/b/g/n or 802.20 (or a combination of 802.11x and 802.20 or other protocols consistent with the present disclosure), all or a portion of the Internet, another communication network, or a combination

of communication networks. The communication network can communicate with, for example, Internet Protocol (IP) packets, Frame Relay frames, Asynchronous Transfer Mode (ATM) cells, voice, video, data, or other information between network nodes.

The computing system can include clients and servers. A client and server are generally remote from each other and typically interact through a communication network. The relationship of client and server arises by virtue of computer programs running on the respective computers and having a client-server relationship to each other.

While this specification contains many specific implementation details, these should not be construed as limitations on the scope of any inventive concept or on the scope of what can be claimed, but rather as descriptions of features that can be specific to particular implementations of particular inventive concepts. Certain features that are described in this specification in the context of separate implementations can also be implemented, in combination, in a single implementation. Conversely, various features that are described in the context of a single implementation can also be implemented in multiple implementations, separately, or in any sub-combination. Moreover, although previously described features can be described as acting in certain combinations and even initially claimed as such, one or more features from a claimed combination can, in some cases, be excised from the combination, and the claimed combination can be directed to a sub-combination or variation of a sub-combination.

Particular implementations of the subject matter have been described. Other implementations, alterations, and permutations of the described implementations are within the scope of the following claims as will be apparent to those skilled in the art. While operations are depicted in the drawings or claims in a particular order, this should not be understood as requiring that such operations be performed in the particular order shown or in sequential order, or that all illustrated operations be performed (some operations can be considered optional), to achieve desirable results. In certain circumstances, multitasking or parallel processing (or a combination of multitasking and parallel processing) can be advantageous and performed as deemed appropriate.

Moreover, the separation or integration of various system modules and components in the previously described implementations should not be understood as requiring such separation or integration in all implementations, and it should be understood that the described program components and systems can generally be integrated together in a single software product or packaged into multiple software products.

Accordingly, the previously described example implementations do not define or constrain the present disclosure. Other changes, substitutions, and alterations are also possible without departing from the spirit and scope of the present disclosure.

Furthermore, any claimed implementation is considered to be applicable to at least a

computer-implemented method; a non-transitory, computer-readable medium storing computer-readable instructions to perform the computer-implemented method; and a computer system comprising a computer memory interoperably coupled with a hardware processor configured to perform the computer-implemented method or the instructions stored on the non-transitory, computer-readable medium.