

(12) 发明专利申请

(10) 申请公布号 CN 102946384 A

(43) 申请公布日 2013. 02. 27

(21) 申请号 201210410762. X

(22) 申请日 2012. 10. 24

(71) 申请人 北京奇虎科技有限公司

地址 100088 北京市西城区新街口外大街
28 号 D 座 112 室(德胜园区)

申请人 奇智软件(北京)有限公司

(72) 发明人 于富龙 黄来安

(74) 专利代理机构 北京智汇东方知识产权代理
事务所(普通合伙) 11391

代理人 康正德 范晓斌

(51) Int. Cl.

H04L 29/06 (2006. 01)

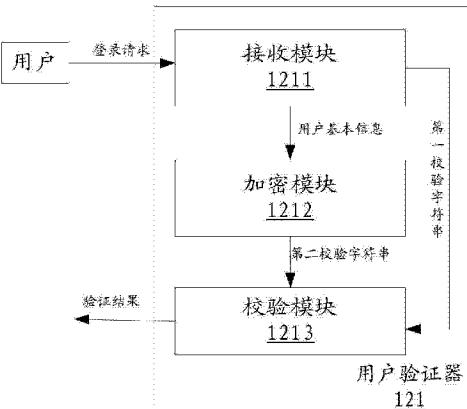
权利要求书 2 页 说明书 11 页 附图 5 页

(54) 发明名称

用户验证方法和设备

(57) 摘要

本发明公开了一种用户验证设备，包括：接收模块，配置为接收来自用户的、包括验证信息的登录请求，其中，该验证信息包括第一信息部分和第二信息部分，第一信息部分包括用户的基本信息，第二信息部分包括利用预定加密规则对用户的基本信息进行加密之后生成的第一校验字符串；加密模块，配置为利用预定加密规则对用户的基本信息进行加密，得到第二校验字符串；验证模块，配置为确定第二校验字符串与第一校验字符串是否匹配，如果匹配则验证成功。采用本发明能够解决恶意用户会通过伪造 cookie 从而登录到系统导致的安全存在隐患的技术问题。本发明还公开了一种相应的用户验证方法和系统。



1. 一种用户验证设备,包括 :

接收模块,配置为接收来自用户的、包括验证信息的登录请求,其中,该验证信息包括第一信息部分和第二信息部分,所述第一信息部分包括所述用户的基本信息,所述第二信息部分包括利用预定加密规则对所述用户的基本信息进行加密之后生成的第一校验字符串;

加密模块,配置为利用所述预定加密规则对所述用户的基本信息进行加密,得到第二校验字符串;

验证模块,配置为确定所述第二校验字符串与所述第一校验字符串是否匹配,如果匹配则验证成功。

2. 根据权利要求 1 的用户验证设备,其特征在于,所述第一校验字符串为利用所述预定加密规则根据密钥对所述用户的基本信息进行加密而生成;

所述第二信息部分还包括与所述密钥相对应的版本信息,以及

所述加密模块还配置为基于所述第二信息部分中的版本信息获取密钥,并且利用所述预定加密规则根据所获取的密钥来对所述用户的基本信息进行加密,生成所述第二校验字符串。

3. 根据权利要求 1 或 2 所述的用户验证设备,其特征在于,所述第一信息部分中的所述用户的基本信息以预定移位规则进行移位;以及

所述加密模块还配置为进行加密操作前,对所接收的第一信息部分中的所述用户的基本信息按照与所述预定移位规则相反的移位规则进行移位处理。

4. 根据权利要求 1 至 3 任一项所述的用户验证设备,其特征在于,所述用户的基本信息包括下列至少一项:

登录用户名、真实姓名、用户唯一标识 id、用户头像 id、注册模板、注册来源、用户类型、登录邮箱。

5. 根据权利要求 1 至 4 任一项所述的用户验证设备,其特征在于,所述第一信息部分和所述第二信息部分存储在 cookie 中,而且所述登录请求通过超文本传送协议 HTTP 发送。

6. 一种用户验证方法,包括 :

接收来自用户的、包括验证信息的登录请求,其中,该验证信息包括第一信息部分和第二信息部分,所述第一信息部分包括所述用户的基本信息,所述第二信息部分包括利用预定加密规则对所述用户的基本信息进行加密之后生成的第一校验字符串;

利用所述预定加密规则对所述用户的基本信息进行加密,生成第二校验字符串;

确定所述第二校验字符串与所述第一校验字符串是否匹配,如果匹配则验证成功。

7. 根据权利要求 6 所述的方法,其特征在于,所述第一校验字符串为利用所述预定加密规则根据密钥对所述用户的基本信息进行加密而生成;

所述第二信息部分还包括与所述密钥相对应的版本信息,以及

所述生成第二校验字符串的步骤包括:基于所述第二信息部分中的版本信息获取密钥,并且利用所述预定加密规则根据所获取的密钥来对所述用户的基本信息进行加密,生成所述第二校验字符串。

8. 根据权利要求 6 或 7 所述的方法,其特征在于,所述第一信息部分中的所述用户的基本信息以预定移位规则进行移位;以及该方法还包括步骤:

进行加密操作前，对所接收的第一信息部分中的所述用户的基本信息按照与所述预定移位规则相反的移位规则进行移位处理。

9. 根据权利要求 6 至 8 任一项所述的方法，其特征在于，所述用户的基本信息包括下列至少一项：

登录用户名、真实姓名、用户唯一标识 id、用户头像 id、注册模板、注册来源、用户类型、登录邮箱。

10. 根据权利要求 6 至 9 任一项所述的方法，其特征在于，所述第一信息部分和所述第二信息部分存储在 cookie 中，而且所述登录请求通过超文本传送协议 HTTP 发送。

用户验证方法和设备

技术领域

[0001] 本发明涉及网络安全领域,具体涉及用户验证方法、用户验证设备、验证服务器以及用户验证系统。

背景技术

[0002] 随着互联网的快速发展,出现了越来越多的网络应用,这些网络应用的用户也越来越多。网络应用为了应对大量用户的登录请求,保证用户登录的安全性,通常需要构建后台数据库来对用户进行验证。目前在网络应用系统中,在用户登录网络应用系统时,网络应用会获取用户名和密码,将其与数据库中存储的用户名和密码进行比较,从而确定用户是否可以正确地登录到网络应用系统中。即,网络应用每次接收到用户的登录请求时,会从数据库中获取相应数据,与登录请求进行匹配;若匹配成功,则允许用户登录,若不成功,则拒绝用户登录。

[0003] 但是,随着网络(Web)应用的快速增多,网络应用对用户进行验证的系统资源消耗也越来越大,这部分消耗在系统的总资源中所占的比重也在逐渐加大,增加了系统的负担。若在短时间内出现大量用户同时登录,超出系统承载能力,甚至可能造成系统崩溃的恶果。

[0004] 为了减少大量用户登录请求对系统的影响,部分网络应用系统采用了 cookie (HTTP (Hypertext transfer protocol, 超文本传送协议) 标准中的一种缓存机制) 登录机制。具体的,对一个采用了 cookie 机制的网络应用,当用户首次成功登录之后,生成对该用户的 cookie 信息,将相关信息存储在 cookie 中。随后,在预定时间内,cookie 维持有效状态,该用户根据 cookie 中的标识直接进入系统而无需再次对用户的登录进行验证。

[0005] 然而,由于 cookie 的安全性等原因,如果不对 cookie 的信息进行验证,有些恶意用户会通过伪造 cookie 从而登录到系统中,这会导致系统的安全存在隐患。

发明内容

[0006] 鉴于上述问题,提出了本发明以便提供一种克服上述问题或者至少部分地解决上述问题的用户验证设备、验证服务器、用户验证系统和相应的用户验证方法。

[0007] 依据本发明的一个方面,提供了一种用户验证设备,包括:接收模块,配置为接收来自用户的、包括验证信息的登录请求,其中,该验证信息包括第一信息部分和第二信息部分,第一信息部分包括用户的基本信息,第二信息部分包括利用预定加密规则对用户的基本信息进行加密之后生成的第一校验字符串;加密模块,配置为利用预定加密规则对用户的基本信息进行加密,得到第二校验字符串;验证模块,配置为确定第二校验字符串与第一校验字符串是否匹配,如果匹配则验证成功。

[0008] 可选的,第一校验字符串为利用预定加密规则根据密钥对用户的基本信息进行加密而生成;第二信息部分还包括与密钥相对应的版本信息,以及加密模块还配置为基于第二信息部分中的版本信息获取密钥,并且利用预定加密规则根据所获取的密钥来对用户的基本信息进行加密,生成第二校验字符串。

[0009] 可选的,第一信息部分中的用户的基本信息以预定移位规则进行移位;以及加密模块还配置为进行加密操作前,对所接收的第一信息部分中的用户的基本信息按照与预定移位规则相反的移位规则进行移位处理。

[0010] 可选的,用户的基本信息包括下列至少一项:登录用户名、真实姓名、用户唯一标识 id、用户头像 id、注册模板、注册来源、用户类型、登录邮箱。

[0011] 可选的,第一信息部分和第二信息部分存储在 cookie 中,而且登录请求通过 HTTP 发送。

[0012] 根据本发明的另一方面,提供了一种验证服务器,包括:根据本发明的用户验证设备;信息获取器,耦接到用户验证器,配置为当用户验证器验证成功时,从第一信息部分中解析出用户的基本信息,并发送至用户;以及用户登录接口,耦接到用户验证器,配置为当用户验证器验证失败时,向用户呈现用户登录界面。

[0013] 可选的,用户登录接口还配置为接收用户经由用户登录界面发送的用户名和密码;以及,验证服务器还包括:用户信息存储器,配置为存储各用户的基本信息;系统验证器,耦接到用户信息存储器,配置为接收来自用户登录接口的用户名和密码,并确定用户名和密码是否存储在用户信息存储器中,若是,则验证成功;以及验证信息生成器,耦接到系统验证器,配置为当系统验证器验证成功时,从用户信息存储器中获取该用户的基本信息,并生成包括第一信息部分和第二信息部分的验证信息,其中,第一信息部分包括用户的基本信息,第二信息部分包括利用预定加密规则对用户的基本信息进行加密之后生成的第一校验字符串。

[0014] 根据本发明的另一方面,提供了一种用户验证系统,包括:上述验证服务器;客户端,耦接到验证服务器,配置为向验证服务器发起来自用户的登录请求。

[0015] 可选的,客户端包括:用户名 / 密码输入接口,配置为向验证服务器发送包括用户名和密码的登录请求;cookie,配置为存储验证服务器中的验证信息生成器所生成的验证信息;服务器登录接口,配置为向验证服务器发送包括验证信息的登录请求。

[0016] 根据本发明的另一方面,提供了一种用户验证方法,包括:接收来自用户的、包括验证信息的登录请求,其中,该验证信息包括第一信息部分和第二信息部分,第一信息部分包括用户的基本信息,第二信息部分包括利用预定加密规则对用户的基本信息进行加密之后生成的第一校验字符串;利用预定加密规则对用户的基本信息进行加密,生成第二校验字符串;确定第二校验字符串与第一校验字符串是否匹配,如果匹配则验证成功。

[0017] 可选的,第一校验字符串为利用预定加密规则根据密钥对用户的基本信息进行加密而生成;第二信息部分还包括与密钥相对应的版本信息,以及,生成第二校验字符串的步骤包括:基于第二信息部分中的版本信息获取密钥,并且利用预定加密规则根据所获取的密钥来对用户的基本信息进行加密,生成第二校验字符串。

[0018] 可选的,第一信息部分中的用户的基本信息以预定移位规则进行移位;以及,该方法还包括步骤:进行加密操作前,对所接收的第一信息部分中的用户的基本信息按照与预定移位规则相反的移位规则进行移位处理。

[0019] 可选的,用户的基本信息包括下列至少一项:登录用户名、真实姓名、用户唯一标识 id、用户头像 id、注册模板、注册来源、用户类型、登录邮箱。

[0020] 可选的,第一信息部分和第二信息部分存储在 cookie 中,而且登录请求通过 HTTP

发送。

[0021] 根据本发明的第一信息部分、第二信息部分可以实现用户的登录请求的自校验，由此解决了恶意用户会通过伪造 cookie 从而登录到系统中，这会导致系统的安全存在隐患的技术问题，取得了保障系统的安全性的有益效果。

[0022] 另外，由于本发明提供了一种安全和快捷的用户登录认证方式，因此，可以将用户的较多基本信息均存储在客户端，而无需每次都在服务器上进行数据查询操作来获取用户的信息，从而显著减少了服务器的负载，提高了验证服务器的效率。

[0023] 上述说明仅是本发明技术方案的概述，为了能够更清楚了解本发明的技术手段，而可依照说明书的内容予以实施，并且为了让本发明的上述和其它目的、特征和优点能够更明显易懂，以下特举本发明的具体实施方式。

附图说明

[0024] 通过阅读下文优选实施方式的详细描述，各种其他的优点和益处对于本领域普通技术人员将变得清楚明了。附图仅用于示出优选实施方式的目的，而并不认为是对本发明的限制。而且在整个附图中，用相同的参考符号表示相同的部件。在附图中：

[0025] 图 1 示出了根据本发明一个实施例的用户验证系统的结构示意图；

[0026] 图 2 示出了根据本发明一个实施例的用户验证器的结构示意图；

[0027] 图 3 示出了根据本发明一个实施例的用户信息验证方法的流程图；

[0028] 图 4 示出了根据本发明一个实施例的用户信息验证方法的流程图；

[0029] 图 5 示出了根据本发明一个实施例的用户信息验证方法的流程图；

[0030] 图 6 示出了根据本发明一个实施例的用户登录的验证过程的网络环境示意图。

具体实施方式

[0031] 下面将参照附图更详细地描述本公开的示例性实施例。虽然附图中显示了本公开的示例性实施例，然而应当理解，可以以各种形式实现本公开而不应被这里阐述的实施例所限制。相反，提供这些实施例是为了能够更透彻地理解本公开，并且能够将本公开的范围完整的传达给本领域的技术人员。

[0032] 相关技术中提及，在用户首次登录网络应用后，可以在一定的时间内利用 cookie 直接进入系统，无需再次进行验证。而基于 cookie 本身的属性，可能出现恶意用户伪造 cookie 登录到系统中的情况，对系统的安全造成隐患。

[0033] 为解决上述技术问题，本发明实施例提供了与相关技术不同的登录方式。图 1 示出了根据本发明一个实施例的用户验证系统的结构示意图。如图 1 所示，客户端 110 钩接到验证服务器 120。当客户端 110 进行登录时，该客户端 110 向验证服务器 120 发起用户登录请求，进而经由验证服务器 120 对登录请求进行验证。据此，将本发明实施例提供的系统称之为用户验证系统。由此可见，采用本发明实施例的用户验证系统，在用户登录时，即使在存在 cookie 的情况下，也需要对登录请求进行验证，不能够利用 cookie 直接进入系统，避免了恶意用户伪造 cookie 登录到系统中的情况，提高了系统的安全性。

[0034] 现分别对该用户验证系统的系统架构以及各部分的功能进行详细说明。

[0035] 在客户端 110 处，考虑到用户可能是首次登录网络应用，也可能是非首次登录该

网络应用,根据登录的类型,登录请求的类型也不相同。若用户是首次登录或者是验证服务器 120 在对用户的登录请求验证失败之外而要求用户明确输入用户名和密码时,则该用户通过客户端发送的登录请求应该是包括用户名和密码信息的登录请求。在其它情况下,则该用户通过客户端发送的应该是包括在诸如 cookie 之类的缓存中存储的验证信息的登录请求。

[0036] 因此,与各登录请求的类型相对应,客户端中需要不同接口及其他结构。参见图 1,客户端 110 包括用户名 / 密码输入接口 111 和服务器登录接口 113。用户名 / 密码输入接口 111 可以向验证服务器 120 发送包括用户名和密码的登录请求,而服务器登录接口 113 可以向验证服务器 120 发送包括验证信息的登录请求。

[0037] 另外,客户端 110 还可以包括诸如 cookie 之类的缓存器 112,其存储来自验证服务器 120 的验证信息,并且在客户端 110 需要利用服务器登录接口 113 来向验证服务器 120 发送登录请求时,将缓存器 112 中缓存的验证信息提供给服务器登录接口 113。

[0038] 可选地,缓存器 112 可以采用多种方式,例如,在客户端 110 和服务器 120 之间的通信采用 HTTP 协议时,缓存器 112 可以采用 cookie 的方式,当然,根据客户端 110 和服务器 120 之间的传输协议,缓存器 112 可以采用其它方式。

[0039] 该客户端 110 的架构仅仅是根据登录请求的类型进行划分,并不对客户端的实际架构造成限定。根据具体情况,客户端中还可以存在多个其他器件或模块或接口。

[0040] 服务器登录接口 115 发送的登录请求中的验证信息包括第一信息部分和第二信息部分,第一信息部分包括用户的基本信息,而第二信息部分包括利用预定加密规则对用户的基本信息进行加密之后生成的第一校验字符串。为保证验证信息的可靠性,该验证信息由验证服务器 120 中的验证信息生成器生成,具体的生成方式在对验证服务器 120 进行描述时来进行详细描述。

[0041] 在验证服务器 120 处,针对客户端 110 的不同登录请求方式,验证服务器的验证方式也不同。当登录请求来自客户端 110 的服务器登录接口 115 时,验证服务器 120 利用用户验证器 121 对该登录请求进行验证,该登录请求包括的验证信息包括上文提及的第一信息部分和第二信息部分。

[0042] 参见图 1,验证服务器 120 包括用户验证器 121、信息获取器 122 和用户登录接口 123。用户验证器 121 对来自客户端 110 的服务器登录接口 115 的登录请求进行验证。信息获取器 122 耦接到用户验证器 121,在当用户验证器 121 验证成功时,信息获取器 122 从第一信息部分中解析出用户的基本信息,并发送至客户端 110。用户登录接口 123 耦接到用户验证器 121,在当用户验证器 121 验证失败时,向用户呈现用户登录界面。

[0043] 根据本发明的实施例,验证服务器中的用户验证器能够对登录请求进行验证,达到保障系统安全性的目的。而当用户验证器验证失败时,可以利用用户登录接口向用户呈现用户登录界面,从而使得用户可以切换界面,使用其他登录方式进行登录,提高用户的感受体验。

[0044] 另外,现有的一般相关技术中,一般网络应用在 cookie 中存放的用户信息较少,这导致当用户登录以后,在获取其基本信息时,需要访问后台数据库,从而导致数据库的负载变大。而在本发明的实施例中,验证服务器通过信息获取器可以直接将第一信息部分中的用户基本信息发送给用户,不需要到后台数据库中进行用户的基本信息的调用和获取,

从而在有效防止用户篡改、伪造登录请求(例如 cookie)的同时,同时减少了验证服务器和后台数据库的消息传递数量,减少了验证服务器和后台数据库的负载,降低系统的消耗,提高系统的稳定性。

[0045] 图2示出了根据本发明一个实施例的用户验证器121的结构示意图。如图2所示,用户验证器121包括接收模块1211、加密模块1212和验证模块1213。接收模块1211接收来自客户端的登录请求(该登录请求例如为来自客户端110的服务器登录接口115)。加密模块1212耦接到接收模块1211,利用预定加密规则对登录请求包括的验证信息中的第一信息部分进行加密,得到第二校验字符串。验证模块1213分别耦接到接收模块1211以及加密模块1212,确定加密模块1212生成的第二校验字符串与接收模块1211接收的登录请求包括的验证信息的第二信息部分中的第一校验字符串是否匹配,如果匹配则验证成功,否则验证失败,并输出验证结果。

[0046] 在本发明实施例中,用户的登录请求包括验证信息,而该验证信息包括两部分内容,一部分(即第一信息部分)是用户自身的基本信息,另外一部分(即第二信息部分)是第一校验字符串。用户登录后,用户验证器会根据用户的基本信息生成第二校验字符串,若两次生成的校验字符串不匹配,则验证不成功,用户无法登录到该网络应用中。即,在本发明实施例中,增加了对登录请求进行验证的步骤,若恶意用户伪造 cookie 登录到系统中,伪造的 cookie 与用户验证器生成的第二校验字符串必然不匹配,从而能够拒绝恶意用户的登录,保障了系统的安全性。

[0047] 另外,在本发明实施例中,利用用户验证器就能够实现对用户登录请求的验证,不需要到数据库中获取用户名和密码,减少了对数据库的数据调用操作,减少了数据库的负载,从而降低了系统的负载,提高系统的安全性。

[0048] 根据上述分析可以得知,登录请求包括的验证信息中的第一信息部分和第二信息部分对于用户验证器121能否验证成功具有关键的意义,因此,下面以一个具体示例来描述验证信息的具体内容。

[0049] 为方便描述,在本例中,将第一信息部分称为Q串,将第二信息部分称为T串,其中,Q串中存储用户的基本信息,而T串为对Q串及自身进行校验后生成的校验字符串。

[0050] Q串中存储用户的基本信息,其一种可选的定义方式如下:

[0051] u={encryptUserName}&r={encryptRealname}&qid={qid}&im={imageId}&s={theme}&src={src}&t={type}&le={loginEmail}。

[0052] 具体来说,Q串的各属性介绍如下:

[0053] 登录用户名(u={encryptUserName})、真实姓名(r={encryptRealname})、用户唯一id(qid={qid})、用户头像id(im={imageId})、注册模板(s={theme})、注册来源(src={src})、用户类型(t={type})、登录邮箱(le={loginEmail})等信息以关键词/值(key/value)对形式存在,且使用&连接不同的key/value值。

[0054] 由于Q串包含的用户基本信息的种类不同,则Q串的定义规则也不同。本实施例中提供的Q串的定义规则仅仅是一个示例。例如,对于登录用户名和真实姓名,考虑到用户隐私,还可以在存储到Q串中之前进行诸如移位之类的加密处理。

[0055] Q串的一个具体示例如下:

[0056] u=fvznbcvt&r=%25Q3%25QN%2508%2500%25P1%25SN&qid=13792776&im=2d01121qc

4a1&s=&src=i360&t=1&le=yufulong@yahoo.cn

[0057] 从举例的 Q 串可以看出,其用户名和真实姓名并没有呈现,而是以扰码的形式出现,此处是针对用户名、真实姓名利用预定移位规则进行了移位处理。本例中的预定移位规则采用的 ROT13 编码,即,对 rawurlencode 后的 GBK 编码用户名字符串进行移位处理,移位前为 simaopig。当然,根据本发明的实施例,也可以采用其他的预定移位规则,例如间隔性移位或者按照指定顺序进行字符换位等等,所有可以将用户名、真实姓名进行移位操作的方式都在本发明的保护范围之内。

[0058] T 串用于对 Q 串及自身进行校验,防止用户篡改、伪造登录请求。本实施例提供的 T 串的一种定义如下:

[0059] s={signature}&t={logintime}&a={is_keep_alive}&v=1.0

[0060] 其中,T 串的各属性介绍如下:

[0061] T 串具体包括:用户签名(s={signature})、用户的登录时间(t={logintime})、登录时是否选择记住登录状态(a={is_keep_alive})、签名所用私钥的版本(v=1.0)等信息。与 Q 串相对应,T 串中各属性也是以 key/value 对形式存在,且使用 & 连接不同的 key/value 值。

[0062] 与 Q 串相类似,T 串的定义规则也取决包含属性的种类、数量以及各属性的具体值。本实施例中提供的 T 串的定义规则仅仅是一种可选的实例。例如,T 串中还可以包括用户签名时间等属性。

[0063] 其中,生成 T 串所使用的用户签名的预定加密规则也是可以有多种,例如,可以采用如下的加密算法来生成用户签名:

[0064] md5(\$gbkUsername.\$qid.\$logintime.\$loginemail. 与私钥版本相对应的私钥)。

[0065] 在生成 T 串时,需要利用私钥。相对应的,在用户验证器 121 进行验证时,同样需要利用相同的私钥和加密算法来对 Q 串进行处理以生成第二校验字符串。

[0066] 为了保证可以更新私钥和相应的加密算法。根据本发明的一个实施例,可以在 T 串中包含私钥版本,这样,用户验证器 121 中的加密模块可以根据 T 串中的私钥版本来选择相应的私钥和加密算法来生成第二校验字符串。

[0067] T 串的一个具体示例如下:

[0068] s=07a47447ddac1331c89901a1accf 32zm&t=1346250824&a=0&v=1.0

[0069] 在 T 串中,记录用户的登录时间,结合登录时登录状态(is_keep_alive)来校验用户 cookie 的生成周期,可以防止用户更改 cookie 的过期时间来伪造请求。

[0070] 返回到图 1,当用户验证器 121 的验证结果为验证成功时,信息获取器 122 会解析出登录请求中的验证信息的用户基本信息(例如解析出 Q 串中的用户基本信息),并且将这些基本信息返回给客户端 110,以便客户端 110 进行后续处理。而当用户验证器 121 的验证结果为验证失败时,用户登录接口 123 会向客户端 110 提供用户登录界面,以便用户通过客户端 110 的用户名 / 密码接入接口 111 向验证服务器 120 发送包括用户名和密码信息的登录请求。为了对来自用户名 / 密码接入接口 111 的登录请求进行处理,参见图 1,验证服务器 120 中还可以包括用户信息存储器 124 和系统验证器 125。用户信息存储器 124 中存储有各用户的基本信息。系统验证器 125 耦接到用户信息存储器 124,在接收到来自用户登录

接口 123 的用户名和密码之后，并确定接收的用户名和密码信息是否存储在用户信息存储器 124 中，若是，则验证成功。

[0071] 在系统验证器 125 验证成功后，用户能够成功登录到系统。此时，验证服务器 120 还包括验证信息生成器 126，其耦接到系统验证器 125，从用户信息存储器 124 中获取该用户的基本信息，并生成包括第一信息部分和第二信息部分的验证信息，其中，第一信息部分包括用户的基本信息，第二信息部分包括利用预定加密规则对用户的基本信息进行加密之后生成的第一校验字符串。

[0072] 前文提及，验证信息生成器 126 能够生成包括第一信息部分和第二信息部分的验证信息。生成的验证信息存储在客户端 110 的 cookie 112 中，当客户端 110 发起登录请求时，从 cookie 112 中获取包括第一信息部分和第二信息部分的验证信息，将其携带在登录请求中，发送至用户验证器 121 进行校验。据此可知，用户验证器 121 是对验证信息生成器 126 生成的验证信息进行验证，两者是相辅相成的。若用户验证器 121 验证成功，则用户能够成功登录业务应用，用户本次登录所使用的验证信息是正确的，不需要验证信息生成器 126 重新生成。而在用户验证器 121 验证失败的时候，用户会触发系统验证器 125 的验证操作，该操作证明此次登录没有正确的验证信息。因此，在系统验证器 125 验证通过后，验证信息生成器 126 会根据相关信息生成新的验证信息，而不是使用一成不变的验证信息。若下一次登录该网络应用的用户不是伪造 cookie 的恶意用户，则该用户使用的验证信息是由验证信息生成器 126 最新生成的。由此可见，本发明实施例提供的验证方法既能保证验证信息的可靠性，又能够提高用户登录的安全性。

[0073] 上述提供的验证服务器的架构仅仅是优选实施例，并不对验证服务器的实际架构造成限定。验证服务器中还可以存在多个其他器件或模块或接口，根据具体情况而定。

[0074] 图 3 示出了根据本发明一个实施例的用户信息验证方法的流程图。该流程起始于步骤 S302，在步骤 S302 中，接收来自客户端的、包括验证信息的登录请求。随后，在步骤 S304 中，提取步骤 S302 中客户端发起的登录请求，验证用户的验证信息是否正确。如上所述，客户端的登录请求包括验证信息，而验证信息包括第一信息部分和第二信息部分，第一信息部分包括用户的基本信息，而第二信息部分包括利用预定加密规则对用户的基本信息进行加密之后生成的第一校验字符串。步骤 S304 中的具体验证处理会在下面参考图 5 给出的方法中进行详细描述。

[0075] 如果在步骤 S304 的验证成功，则可选地，验证方法进入步骤 S306，其中从步骤 S302 所获取的登录请求中的验证信息包括的第一信息部分中解析出用户的基本信息，并传输返回至客户端。

[0076] 如果在步骤 S304 的验证失败，则可选地，该方法可以进入步骤 S308，在该步骤 S308 中，向用户呈现用户登录界面，以便用户输入用户名和密码，随后在验证服务器中利用系统验证器进行进一步验证，此时具体的验证步骤请参见图 4。

[0077] 由图 3 所示流程可知，用户发起登录请求时，利用用户验证系统中的验证服务器对登录请求进行验证，能够提高用户登录的安全性。综上，可以得知，在验证服务器一侧，对于登录请求进行验证，可以有两种验证结果，一种是验证通过，另外一种是对应的验证失败。

[0078] 在验证通过的情况下，验证服务器会利用信息获取器从第一信息部分中解析出用

户的基本信息，并发送至用户。在用户基本信息的获取过程中，验证服务器不需要到后台数据库中进行用户的基本信息的调用和获取，减少了验证服务器和后台数据库的消息传递数量，减少了验证服务器和后台数据库的负载，降低系统的消耗，提高系统的稳定性。

[0079] 在验证不通过，或者称之为验证失败的情况下，验证服务器会将用户登录界面呈现给用户。与此相应的，用户可以通过用户登录接口重新输入用户名和密码，再次进行登录。

[0080] 图 4 示出了根据本发明一个实施例的用户通过输入用户名和密码进行登录的用户信息验证方法的流程图。该流程适用于用户验证器验证失败或用户首次登录的情况，起始于步骤 S402。在步骤 S402 中，通过用户登录接口接收用户经由用户登录界面发送的用户名和密码。随后，在步骤 S404 中，利用系统验证器接收步骤 S402 接收的用户名和密码，对该用户名和密码进行查询，确定用户名和密码是否存储在用户信息存储器中。用户信息存储器中存储各用户的基本信息，该基本信息包括用户名和密码信息，还可以包括其他信息，例如用户头像、用户头像尺寸、用户邮箱地址、真实姓名等等，具体的查询方式可以有多种，例如，索引查询、哈希排序查询等等。

[0081] 如果在步骤 S404 中的查询结果为用户名和密码存储在用户信息存储器中，则可选的，验证方法进入步骤 S406，系统验证器验证成功，生成包括第一信息部分和第二信息部分的验证信息。在生成该验证信息的过程中，可以从用户信息存储器中获取该用户的基本信息，且第一信息部分包括用户的基本信息，第二信息部分包括利用预定加密规则对用户的基本信息进行加密之后生成的第一校验字符串。

[0082] 如果在步骤 S404 中的查询结果为用户名和密码并不存储在用户信息存储器中，则可选的，验证方法进行步骤 S408，在该步骤 S408 中，通知用户无法登录该网络应用。因用户信息存储器中并没有该用户名和密码的存储信息，那么后续用户可以该网络应用进行注册，以新用户的身份进行登录。

[0083] 上文提及，步骤 S304 中的具体验证处理会在图 5 给出的方法中进行详细描述。图 5 示出了根据本发明的一个实施例的用户信息验证方法的流程图。该流程起始于步骤 S502，在步骤 S502 中，接收来自用户的登录请求。其中，该登录请求中携带有验证信息，该验证信息包括第一信息部分和第二信息部分，第一信息部分包括用户的基本信息，第二信息部分包括利用预定加密规则对用户的基本信息进行加密之后生成的第一校验字符串。随后，在步骤 S504 中，利用预定加密规则对步骤 S502 中接收的用户的基本信息进行加密，生成第二校验字符串。进而，在步骤 S506 中，从步骤 S502 中接收第一校验字符串，再从步骤 S504 中接收第二校验字符串，对两个校验字符串进行匹配，根据匹配结果确定是否验证成功，如果匹配则验证成功，而如果不匹配则验证失败。

[0084] 采用图 5 所示的方法，根据用户的登录请求自身能够实现对用户的验证，不需要到数据库中获取用户名和密码，减少了对数据库的数据调用操作，减少了数据库的负载，从而降低了系统的负载，提高系统的安全性。

[0085] 并且，在本发明实施例中，增加了对登录请求进行验证的处理，若恶意用户伪造 cookie 登录到系统中，伪造的 cookie 与验证模块生成的第二校验字符串必然不匹配，从而能够拒绝恶意用户的登录，保障了系统的安全性。

[0086] 在一个实施例中，第一校验字符串为利用预定加密规则根据密钥对用户的基本信

息进行加密而生成。相应的，第二信息部分中还包括与密钥相对应的版本信息，以及，基于第二信息部分中的版本信息获取密钥，并且利用预定加密规则根据所获取的密钥来对用户的基本信息进行加密，生成第二校验字符串。此处的密钥为私钥，也可以采用公钥，根据具体情况而定。若采用除密钥外的其他元素也能够达到加密的目的，则也可以采用其他元素。

[0087] 在一个实施例中，考虑到用户的基本信息是可以从数据库中获取的，并不是完全保密的，因此，若用户的基本信息以原方式呈现，则还是有可能被恶意用户截取或盗窃到的。为解决该问题，本实施例提供了一种处理方式，即，将第一信息部分中的用户的基本信息以预定移位规则进行移位。相应的，为保证第二校验字符串与第一校验字符串能够匹配上，需要在进行加密操作前，对所接收的第一信息部分中的用户的基本信息按照与预定移位规则相反的移位规则进行移位处理。按照上述步骤处理后，生成第二校验字符串所使用的用户基本信息和生成第一校验字符串所使用的用户基本信息是相同的，不会出现用户基本信息错位的意外情况，保证了后续校验字符串匹配的成功度。

[0088] 可选地，可以直接将第一信息部分和第二信息部分存储在 cookie 中，并通过 HTTP 发送登录请求。

[0089] 在本实施例中，用户的基本信息可以包括下列至少一项：

[0090] 登录用户名、真实姓名、用户唯一标识(id)、用户头像 id、注册模板、注册来源、用户类型、登录邮箱。网络应用在登录时使用较多的为登录用户名、用户唯一 id 以及登录邮箱中的任意一个或多个的组合，而在该应用的实施过程中，则会较多的使用到用户的真实姓名、头像 id 等相关信息。

[0091] 现提供一个具体实施例，对本发明提供的用户登录的验证过程进行具体说明，该实施例的网络环境示意图请参见图 6。

[0092] 在本实施例中，用户通过客户端发出登录请求，业务方(相当于上文的验证服务器)会根据用户的登录请求中是否存在验证信息(例如 Q/T 串)进行判断，其处理步骤如下。

[0093] 1、用户没有 Q/T 串时，业务方引导用户登录；

[0094] 2、用户有 Q/T 串时，业务方可以利用自身的用户验证器自行根据 Q/T 串算法，校验 Q/T 串是否正确，不需要请求数据库(相当于上文的用户信息存储器)；

[0095] 3、用户 Q/T 串非法，业务方引导用户登录；

[0096] 4、用户 Q/T 串合法，业务方利用自身的信息获取器根据 Q/T 串可以反解出用户信息，不需要查询数据库；

[0097] 5、用户输入用户名密码，在数据库匹配后，用户中心服务器(Server，即验证服务器中的验证信息生成器)为用户设置 Q/T 串 cookie。

[0098] 从系统的角度而言，本次用户登录的步骤如下：

[0099] 步骤 A、用户通过多种终端可以使用统一的用户帐号系统；

[0100] 步骤 B、用户登录时，系统校验用户名密码后，从数据库中取出用户信息，将业务方常用字段存储到 cookie Q 中，设置到用户浏览器；

[0101] 步骤 C、系统使用加密规则，将 Q 串内容按照特定顺序，连接系统私钥使用加密算法生成 T 串，并在 T 串中包含用户登录的系统时间，同样设置到用户浏览器；

[0102] 步骤 D、用户访问公司各业务时，用户中心提供 SDK 方法，可以判断用户是否存在 Q/T，同时判断 Q/T 是否合法——使用同样的签名算法，将 Q 串生成新的 T 串，看是否与用户

cookie 中的 T 串保持一致；

[0103] 步骤 E、如果一致，则将 Q 内包含的用户信息返回给业务方，业务方不需要连接数据库即可获得当前登录用户基本信息，有效减少对 DB 数据库系统的压力。

[0104] 综上，本发明实施例提供的用户验证方法、设备及系统，增加了对登录请求进行验证的步骤，若恶意用户伪造 cookie 登录到系统中，伪造的 cookie 与第二校验字符串必然不匹配，从而能够拒绝恶意用户的登录，保障了系统的安全性。

[0105] 并且，在本发明实施例中，利用用户验证器就能够实现对用户登录请求的验证，不需要到数据库中获取用户名和密码，减少了对数据库的数据调用操作，减少了数据库的负载，从而降低了系统的负载，提高系统的安全性。

[0106] 在此提供的算法和显示不与任何特定计算机、虚拟系统或者其它设备固有相关。各种通用系统也可以与基于在此的示教一起使用。根据上面的描述，构造这类系统所要求的结构是显而易见的。此外，本发明也不针对任何特定编程语言。应当明白，可以利用各种编程语言实现在此描述的本发明的内容，并且上面对特定语言所做的描述是为了披露本发明的最佳实施方式。

[0107] 在此处所提供的说明书中，说明了大量具体细节。然而，能够理解，本发明的实施例可以在没有这些具体细节的情况下实践。在一些实例中，并未详细示出公知的方法、结构和技术，以便不模糊对本说明书的理解。

[0108] 类似地，应当理解，为了精简本公开并帮助理解各个发明方面中的一个或多个，在上面对本发明的示例性实施例的描述中，本发明的各个特征有时被一起分组到单个实施例、图、或者对其的描述中。然而，并不应将该公开的方法解释成反映如下意图：即所要求保护的本发明要求比在每个权利要求中所明确记载的特征更多的特征。更确切地说，如下面的权利要求书所反映的那样，发明方面在于少于前面公开的单个实施例的所有特征。因此，遵循具体实施方式的权利要求书由此明确地并入该具体实施方式，其中每个权利要求本身都作为本发明的单独实施例。

[0109] 本领域那些技术人员可以理解，可以对实施例中的设备中的模块进行自适应性地改变并且把它们设置在与该实施例不同的一个或多个设备中。可以把实施例中的模块或单元或组件组合成一个模块或单元或组件，以及此外可以把它们分成多个子模块或子单元或子组件。除了这样的特征和 / 或过程或者单元中的至少一些是相互排斥之外，可以采用任何组合对本说明书(包括伴随的权利要求、摘要和附图)中公开的所有特征以及如此公开的任何方法或者设备的所有过程或单元进行组合。除非另外明确陈述，本说明书(包括伴随的权利要求、摘要和附图)中公开的每个特征可以由提供相同、等同或相似目的的替代特征来代替。

[0110] 此外，本领域的技术人员能够理解，尽管在此所述的一些实施例包括其它实施例中所包括的某些特征而不是其它特征，但是不同实施例的特征的组合意味着处于本发明的范围之内并且形成不同的实施例。例如，在下面的权利要求书中，所要求保护的实施例的任意之一都可以以任意的组合方式来使用。

[0111] 本发明的各个部件实施例可以以硬件实现，或者以在一个或者多个处理器上运行的软件模块实现，或者以它们的组合实现。本领域的技术人员应当理解，可以在实践中使用微处理器或者数字信号处理器(DSP)来实现根据本发明实施例的各设备中的一些或者全部

部件的一些或者全部功能。本发明还可以实现为用于执行这里所描述的方法的一部分或者全部的设备或者装置程序(例如,计算机程序和计算机程序产品)。这样的实现本发明的程序可以存储在计算机可读介质上,或者可以具有一个或者多个信号的形式。这样的信号可以从因特网网站上下载得到,或者在载体信号上提供,或者以任何其他形式提供。

[0112] 应该注意的是上述实施例对本发明进行说明而不是对本发明进行限制,并且本领域技术人员在不脱离所附权利要求的范围的情况下可设计出替换实施例。在权利要求中,不应将位于括号之间的任何参考符号构造成对权利要求的限制。单词“包含”不排除存在未列在权利要求中的元件或步骤。位于元件之前的单词“一”或“一个”不排除存在多个这样的元件。本发明可以借助于包括有若干不同元件的硬件以及借助于适当编程的计算机来实现。在列举了若干装置的单元权利要求中,这些装置中的若干个可以是通过同一个硬件项来具体体现。单词第一、第二、以及第三等的使用不表示任何顺序。可将这些单词解释为名称。

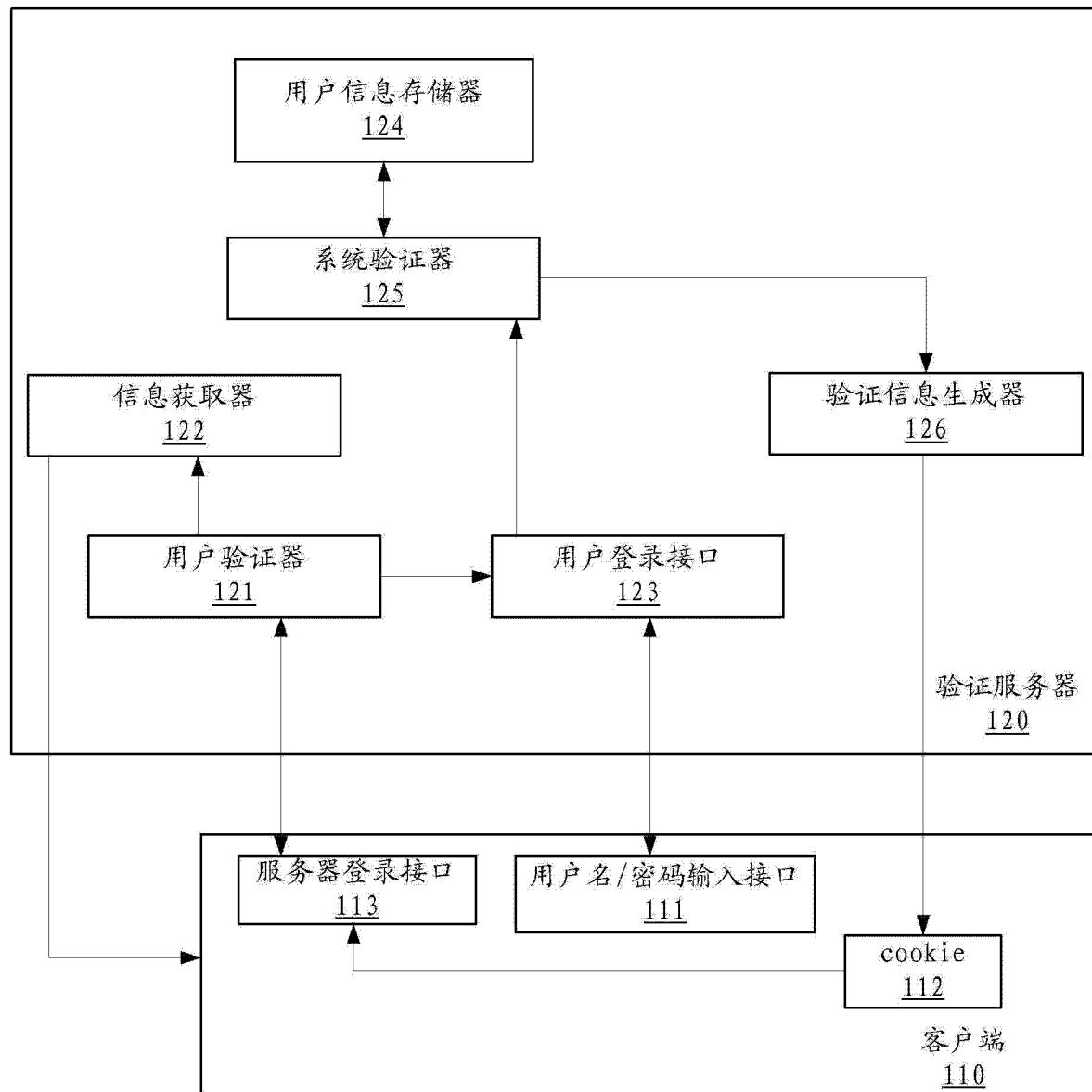


图 1

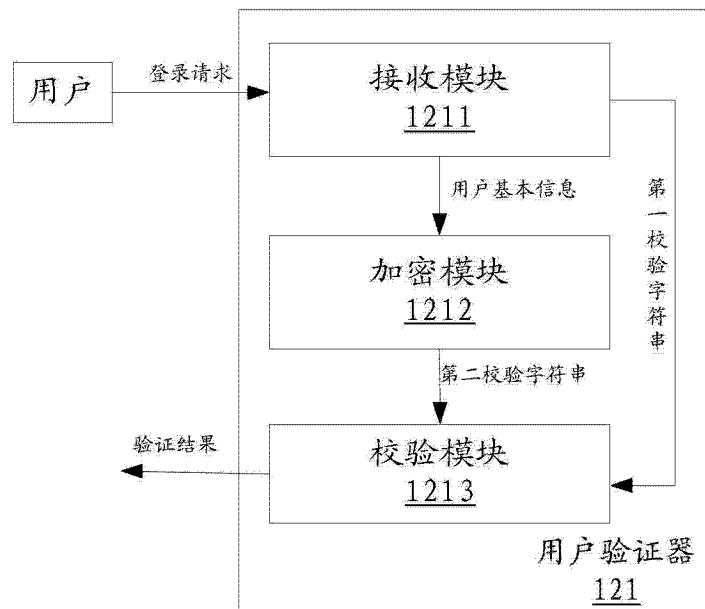


图 2

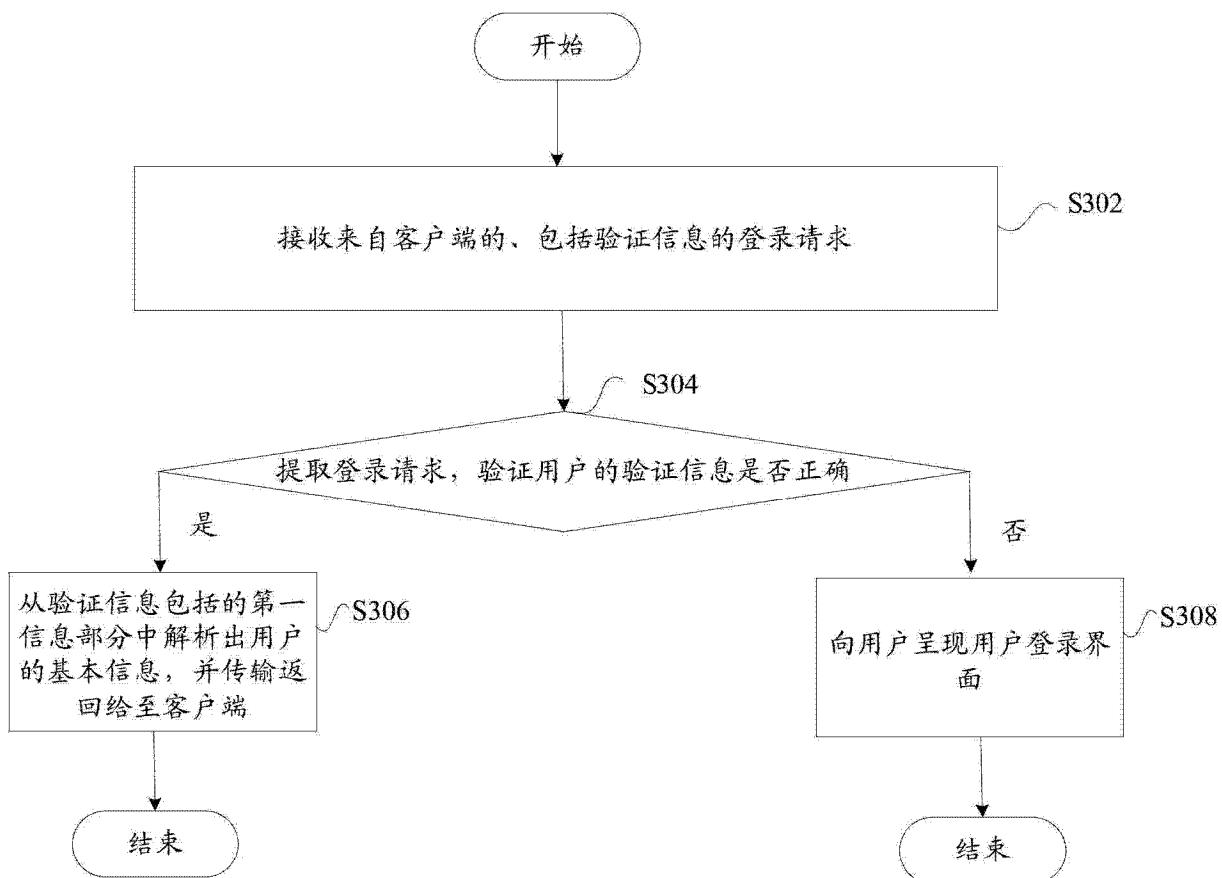


图 3

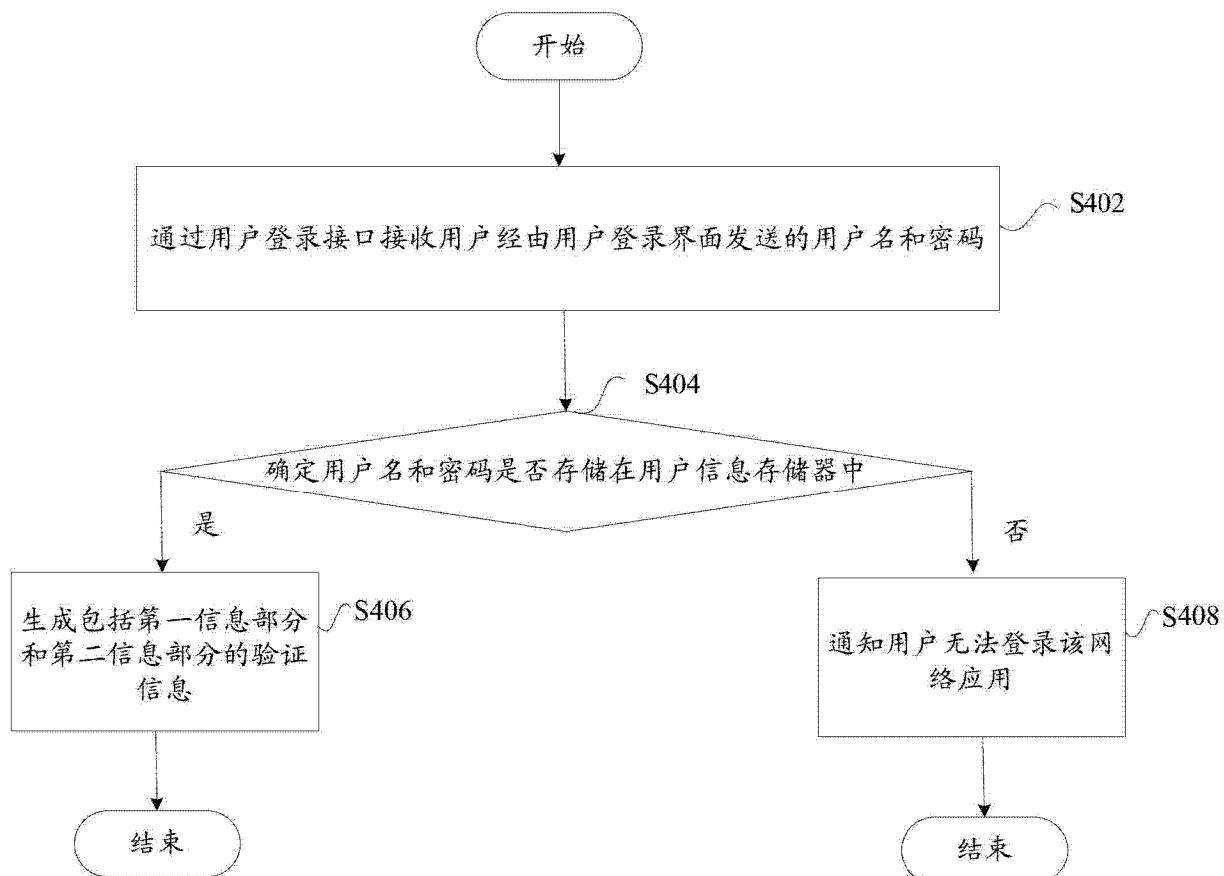


图 4

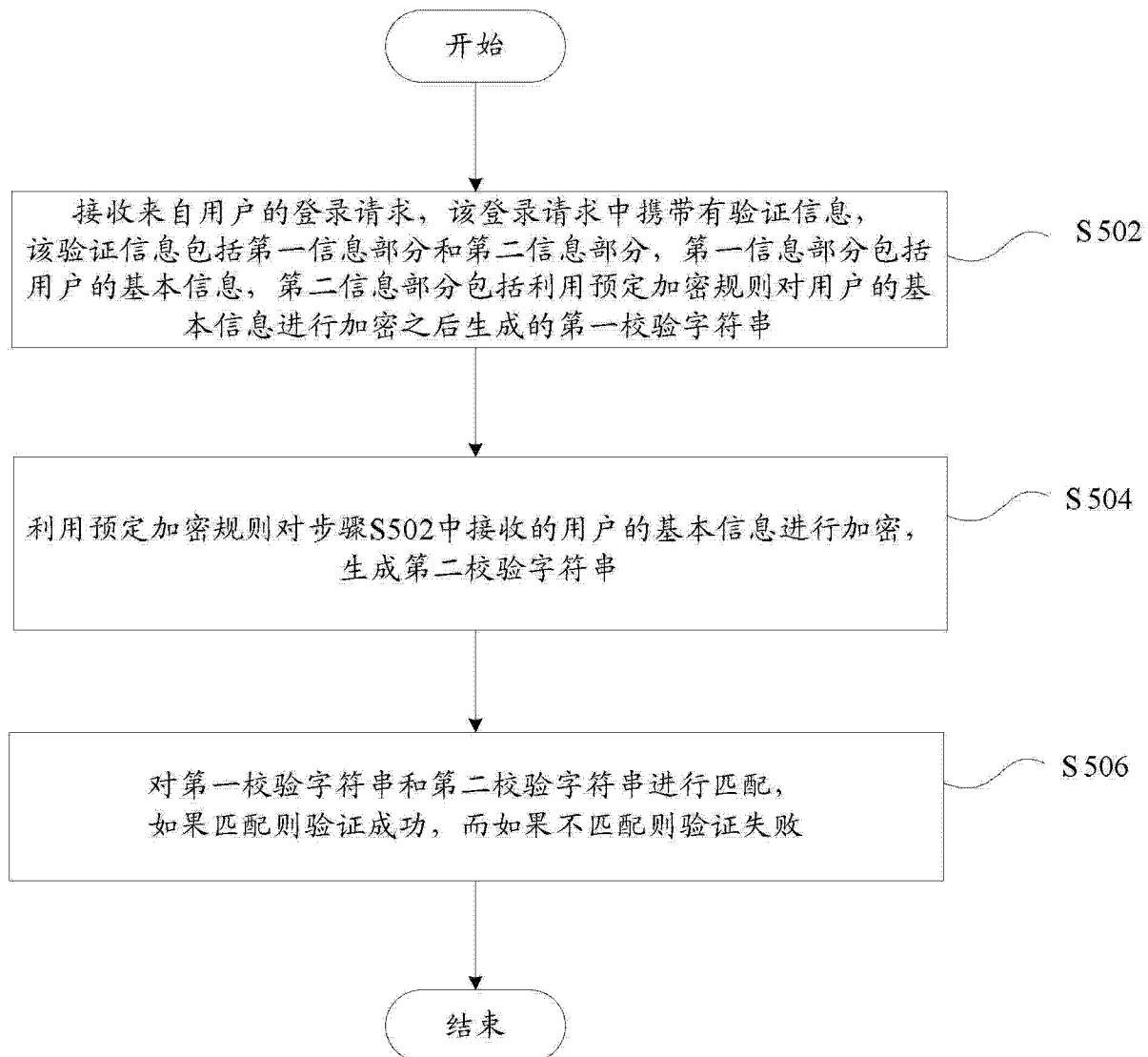


图 5

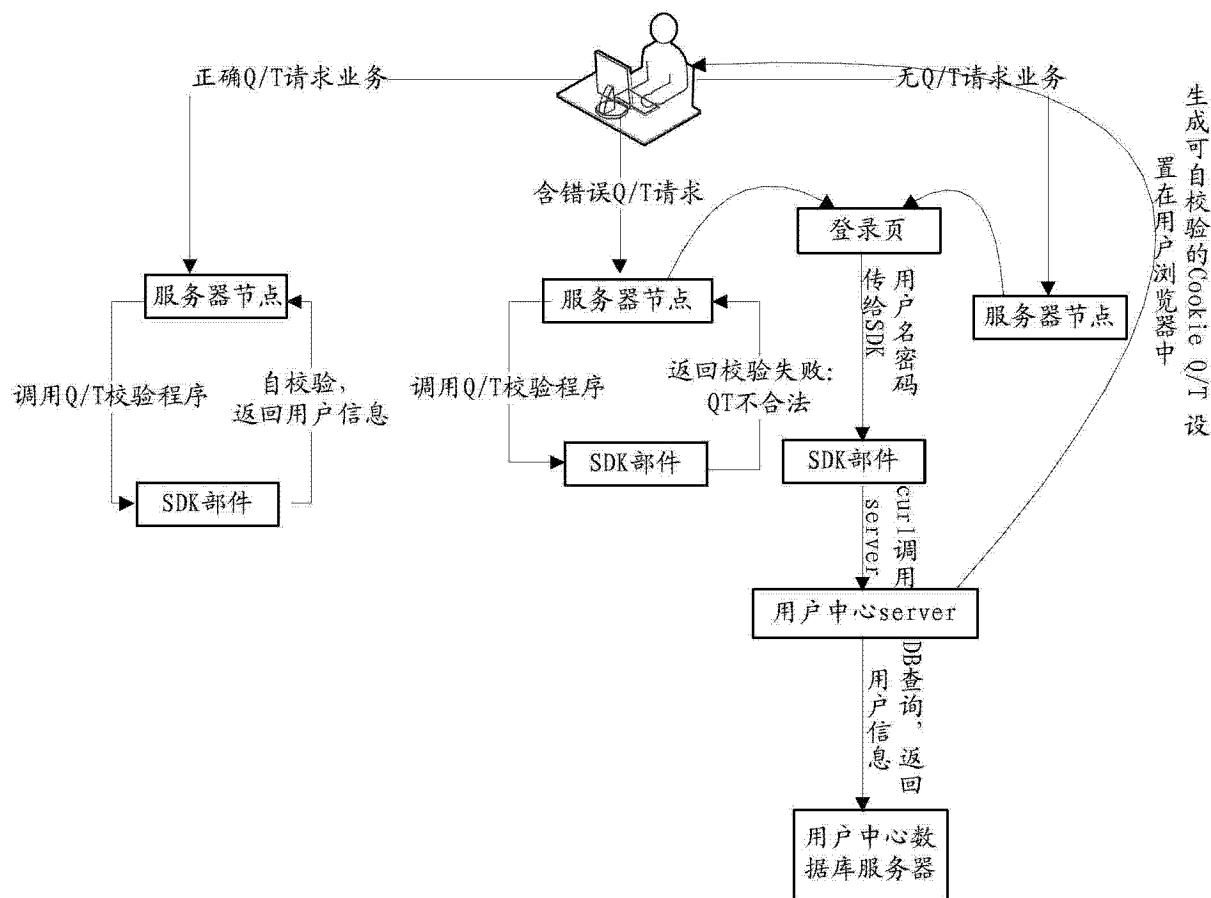


图 6