(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 108322461 A
(43)申请公布日 2018.07.24

(21)申请号 201810094842.6

(22)申请日 2018.01.31

(71)申请人 百度在线网络技术(北京)有限公司
地址 100085 北京市海淀区上地十街10号百度大厦三层

(72)发明人 赵越

(74)专利代理机构 北京品源专利代理有限公司 11332
代理人 孟金喆

(51)Int.Cl.
H04L 29/06(2006.01)
G06F 21/33(2013.01)
G06F 21/60(2013.01)

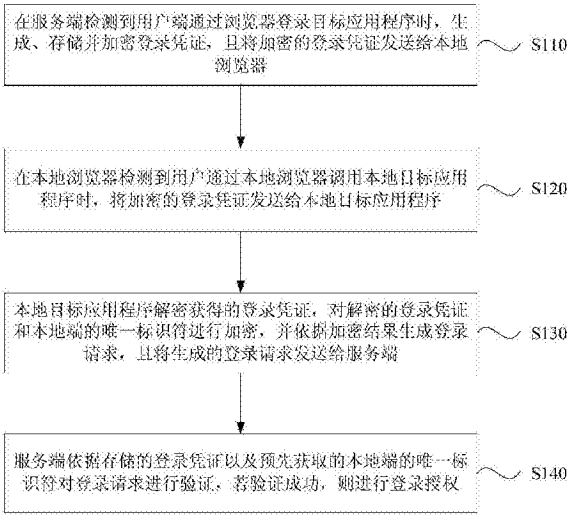
权利要求书2页 说明书11页 附图6页

(54)发明名称

应用程序自动登录的方法、系统、装置、设备和介质

(57)摘要

本发明实施例公开了一种应用程序自动登录的方法、系统、装置、设备和介质。该方法包括：在服务端检测到用户端通过浏览器登录目标应用程序时，生成、存储并加密登录凭证，且将加密的登录凭证发送给本地浏览器；在本地浏览器检测到用户通过本地浏览器调用本地目标应用程序时，将加密的登录凭证发送给本地目标应用程序；本地目标应用程序解密获得的登录凭证，对解密的登录凭证和本地端的唯一标识符进行加密，并依据加密结果生成登录请求，且将生成的登录请求发送给服务端；服务端依据存储的登录凭证以及预先获取的本地端的唯一标识符对登录请求进行验证，验证成功登录授权。该方案可以安全的将登录状态从浏览器同步到客户端程序上。



CN 108322461 A

1.一种应用程序自动登录的方法，其特征在于，包括：

在服务端检测到用户端通过浏览器登录目标应用程序时，生成、存储并加密登录凭证，且将加密的登录凭证发送给本地浏览器；

在所述本地浏览器检测到用户通过所述本地浏览器调用本地目标应用程序时，将所述加密的登录凭证发送给所述本地目标应用程序；

所述本地目标应用程序解密获得的登录凭证，对解密的登录凭证和本地端的唯一标识符进行加密，并依据加密结果生成登录请求，且将生成的登录请求发送给所述服务端；

所述服务端依据存储的登录凭证以及预先获取的本地端的唯一标识符对所述登录请求进行验证，若验证成功，则进行登录授权。

2.根据权利要求1所述的方法，其特征在于，在服务端检测到用户端通过浏览器登录目标应用程序时，生成、存储并加密登录凭证之前，还包括：

若所述服务端检测到本地目标应用程序上报的绑定请求，则获取并存储所述本地目标应用程序上报的本地端的唯一标识符。

3.根据权利要求1所述的方法，其特征在于，所述服务端依据所述登录凭证以及预先获取的本地端的唯一标识符对所述登录请求进行验证，包括：

若所述登录请求中的登录凭证与所述服务端中预先存储的登录凭证相同，且所述登录请求中的唯一标识符与所述服务端预先获取的本地端的唯一标识符相同，则确定验证成功；否则验证失败。

4.根据权利要求1所述的方法，其特征在于，在服务端检测到用户端通过浏览器登录目标应用程序时，还包括：

所述服务端获取并存储当前的登录时间戳；

所述服务端依据所述登录凭证以及预先获取的本地端的唯一标识符对所述登录请求进行验证，包括：

若所述登录请求中的登录凭证与所述服务端中预先存储的登录凭证相同，所述登录请求中的唯一标识符与所述服务端预先获取的本地端的唯一标识符相同，以及所述登录请求的发送时间与所述服务端中预先存储的登录时间戳之间的差值小于时间阈值，则确定验证成功；否则，验证失败。

5.一种应用程序自动登录的方法，其特征在于，包括：

在检测到本地浏览器的调用请求时，从本地浏览器获取加密的登录凭证，其中所述加密的登录凭证是在服务端检测到用户端通过浏览器登录目标应用程序时，生成、存储、加密并下发到所述本地浏览器中的；

解密获取的登录凭证，并对所述登录凭证和本地端的唯一标识符进行加密，依据加密结果生成登录请求，且将登录请求发送给服务端，其中所述登录请求用于指示所述服务端依据存储的登录凭证以及预先获取的本地端的唯一标识符对所述登录请求进行验证，若验证成功，则进行登录授权。

6.一种应用程序自动登录的系统，其特征在于，包括：服务端、本地浏览器和本地目标应用程序；其中，

所述服务端用于在检测到用户端通过浏览器登录目标应用程序时，生成、存储并加密登录凭证，且将加密的登录凭证发送给所述本地浏览器；

所述本地浏览器用于在检测到用户通过所述本地浏览器调用所述本地目标应用程序时，将加密的登录凭证发送给所述本地目标应用程序；

所述本地目标应用程序用于解密获得的登录凭证，对解密的登录凭证和本地端的唯一标识符进行加密，并依据加密结果生成登录请求，且将生成的登录请求发送给所述服务端；

所述服务端还用于依据存储的登录凭证以及预先获取的本地端的唯一标识符对所述登录请求进行验证，若验证成功，则进行登录授权。

7.根据权利要求6所述的系统，其特征在于，所述服务器还用于在检测到用户端通过浏览器登录目标应用程序时，生成、存储并加密登录凭证之前，若检测到本地目标应用程序上报的绑定请求，则获取并存储所述本地目标应用程序上报的本地端的唯一标识符。

8.根据权利要求6所述的系统，其特征在于，所述服务端还用于通过如下方式对所述登录请求进行验证：

若所述登录请求中的登录凭证与所述服务端中预先存储的登录凭证相同，且所述登录请求中的唯一标识符与所述服务端预先获取的本地端的唯一标识符相同，则确定验证成功；否则验证失败。

9.根据权利要求6所述的系统，其特征在于，

所述服务端还用于获取并存储当前的登录时间戳；

所述服务端还用于通过如下方式对所述登录请求进行验证：

若所述登录请求中的登录凭证与所述服务端中预先存储的登录凭证相同，所述登录请求中的唯一标识符与所述服务端预先获取的本地端的唯一标识符相同，以及所述登录请求的发送时间与所述服务端中预先存储的登录时间戳之间的差值小于时间阈值，则确定验证成功；否则，验证失败。

10.一种应用程序自动登录的装置，其特征在于，包括：

登录凭证获取模块，用于在检测到本地浏览器的调用请求时，从本地浏览器获取加密的登录凭证，其中所述加密的登录凭证是在服务端检测到用户端通过浏览器登录目标应用程序时，生成、存储、加密并下发到所述本地浏览器中的；

登录请求发送模块，用于解密获取的登录凭证，并对所述登录凭证和本地端的唯一标识符进行加密，依据加密结果生成登录请求，且将登录请求发送给服务端，其中所述登录请求用于指示所述服务端依据存储的登录凭证以及预先获取的本地端的唯一标识符对所述登录请求进行验证，若验证成功，则进行登录授权。

11.一种设备，其特征在于，所述设备包括：

一个或多个处理器；

存储装置，用于存储一个或多个程序，

当所述一个或多个程序被所述一个或多个处理器执行，使得所述一个或多个处理器实现如权利要求5中所述的应用程序自动登录的方法。

12.一种计算机可读存储介质，其上存储有计算机程序，其特征在于，该程序被处理器执行时实现如权利要求5中所述的应用程序自动登录的方法。

应用程序自动登录的方法、系统、装置、设备和介质

技术领域

[0001] 本发明实施例涉及互联网技术领域，尤其涉及一种应用程序自动登录的方法、系统、装置、设备和介质。

背景技术

[0002] 近年来，随着互联网技术及应用程序的广泛普及，通过在网页浏览器上点击链接的方式调起应用程序，并展示相关的内容的切换方法，可以让用户从浏览网页平滑地切换到使用本地应用程序，提升了用户的体验。

[0003] 同时，如果本地记录了应用程序的密码，则在唤起本地应用程序的同时，可以通过本地应用程序登录。然而，密码需要以一种安全的方式保存，并且一旦保存的文件被擦除或者密码改变仍然无法自动登录。另外，由于web服务是将信息参数传递给本地浏览器，并最终传递给唤起的应用程序。若注册表中本地应用程序的URL(Uniform Resource Locator，统一资源定位符)协议头对应的唤起程序被恶意篡改，还会导致web服务传递的参数泄露。因此，目前若想在通过浏览器唤起本地应用程序的同时实现登录，会造成敏感信息泄露，存在安全隐患。

发明内容

[0004] 本发明实施例提供一种应用程序自动登录的方法、系统、装置、设备和介质，可以安全的将登录状态从浏览器同步到客户端程序上。

[0005] 第一方面，本发明实施例提供了一种应用程序自动登录的方法，该方法包括：

[0006] 在服务端检测到用户端通过浏览器登录目标应用程序时，生成、存储并加密登录凭证，且将加密的登录凭证发送给本地浏览器；

[0007] 在所述本地浏览器检测到用户通过所述本地浏览器调用本地目标应用程序时，将加密的登录凭证发送给本地目标应用程序；

[0008] 所述本地目标应用程序解密获得的登录凭证，对解密的登录凭证和本地端的唯一标识符进行加密，并依据加密结果生成登录请求，且将生成的登录请求发送给服务端；

[0009] 所述服务端依据存储的登录凭证以及预先获取的本地端的唯一标识符对所述登录请求进行验证，若验证成功，则进行登录授权。

[0010] 第二方面，本发明实施例还提供了一种应用程序自动登录的方法，该方法包括：

[0011] 在检测到本地浏览器的调用请求时，从本地浏览器获取加密的登录凭证，其中所述加密的登录凭证是在服务端检测到用户端通过浏览器登录目标应用程序时，生成、存储、加密并下发到所述本地浏览器中的；

[0012] 解密获取的登录凭证，并对所述登录凭证和本地端的唯一标识符进行加密，依据加密结果生成登录请求，且将登录请求发送给服务端，其中所述登录请求用于指示所述服务端依据存储的登录凭证以及预先获取的本地端的唯一标识符对所述登录请求进行验证，若验证成功，则进行登录授权。

[0013] 第三方面，本发明实施例还提供了一种应用程序自动登录的系统，该系统包括：服务端、本地浏览器和本地目标应用程序；其中，

[0014] 所述服务端用于在检测到用户端通过浏览器登录目标应用程序时，生成、存储并加密登录凭证，且将加密的登录凭证发送给所述本地浏览器；

[0015] 所述本地浏览器用于在检测到用户通过所述本地浏览器调用所述本地目标应用程序时，将加密的登录凭证发送给所述本地目标应用程序；

[0016] 所述本地目标应用程序用于解密获得的登录凭证，对解密的登录凭证和本地端的唯一标识符进行加密，并依据加密结果生成登录请求，且将生成的登录请求发送给所述服务端；

[0017] 所述服务端还用于依据存储的登录凭证以及预先获取的本地端的唯一标识符对所述登录请求进行验证，若验证成功，则进行登录授权。

[0018] 第四方面，本发明实施例还提供了一种应用程序自动登录的装置，该装置包括：

[0019] 登录凭证获取模块，用于在检测到本地浏览器的调用请求时，从本地浏览器获取加密的登录凭证，其中所述加密的登录凭证是在服务端检测到用户端通过浏览器登录目标应用程序时，生成、存储、加密并下发到所述本地浏览器中的；

[0020] 登录请求发送模块，用于解密获取的登录凭证，并对所述登录凭证和本地端的唯一标识符进行加密，依据加密结果生成登录请求，且将登录请求发送给服务端，其中所述登录请求用于指示所述服务端依据存储的登录凭证以及预先获取的本地端的唯一标识符对所述登录请求进行验证，若验证成功，则进行登录授权。

[0021] 第五方面，本发明实施例还提供了一种设备，该设备包括：

[0022] 一个或多个处理器；

[0023] 存储装置，用于存储一个或多个程序，

[0024] 当所述一个或多个程序被所述一个或多个处理器执行，使得所述一个或多个处理器实现第二方面中任一所述的应用程序自动登录的方法。

[0025] 第六方面，本发明实施例还提供了一种计算机可读存储介质，其上存储有计算机程序，该程序被处理器执行时实现第二方面中任一所述的应用程序自动登录的方法。

[0026] 本发明实施例提供一种应用程序自动登录的方法、系统、装置、设备和介质，当用户通过浏览器登录目标应用程序时，服务器检测到此操作后，生成、存储并加密登录凭证，同时将加密的登录凭证发送给本地浏览器；本地浏览器将加密的登录凭证发送给本地目标应用程序，本地目标应用程序对解密的登录凭证和本地端的唯一标识符进行加密，从而生成登录请求，且将登录请求发送给服务端；服务端对登录请求进行验证，若验证成功，则进行登录授权。解决了在客户机器上记住密客户端软件的登录密码这种方式，如果记住密码所在的文件/文件夹被不小心擦除，或者用户在其他地方修改过密码，则需要重新输入密码进行登录的问题。可实现安全的将登录状态从浏览器同步到客户端程序上，减少用户的重复输入。

附图说明

[0027] 通过阅读参照以下附图所作的对非限制性实施例所作的详细描述，本发明的其它特征、目的和优点将会变得更明显：

[0028] 图1是本发明实施例一中提供的一种应用程序自动登录的方法的流程图；

[0029] 图2A是本发明实施例一中提供的一种应用程序自动登录的方法的信令图；

[0030] 图2B是本发明实施例一中提供的一种窃密者如何获取登录状态的示意图；

[0031] 图3是本发明实施例二中提供的一种应用程序自动登录的方法的流程图；

[0032] 图4是本发明实施例三中提供的一种应用程序自动登录的方法的流程图；

[0033] 图5是本发明实施例四中提供的一种应用程序自动登录的方法的示意图；

[0034] 图6是本发明实施例五中提供的一种应用程序自动登录的系统的结构框图；

[0035] 图7是本发明实施例六中提供的一种应用程序自动登录的装置的结构框图；

[0036] 图8是本发明实施例七中提供的一种设备的结构示意图。

具体实施方式

[0037] 下面结合附图和实施例对本发明作进一步的详细说明。可以理解的是，此处所描述的具体实施例仅用于解释本发明，而非对本发明的限定。另外还需要说明的是，为了便于描述，附图中仅示出了与本发明相关的部分而非全部内容。

[0038] 实施例一

[0039] 图1为本发明实施例一提供的一种应用程序自动登录的方法的流程图，图2A为本发明实施例一中的应用程序自动登录的方法的信令图。本实施例可适用于用户在较为常用的客户机器环境下，用浏览器浏览web网页，在需要时点击web链接，打开本地目标应用程序进行专业性较强的操作的情况。客户机器可以为PC(Personal Computer，个人计算机)、手机或平板电脑等。整套应用程序自动登录的方法通常由服务端以及客户机器上的本地浏览器和本地目标应用程序配合执行，本实施例所提供的方法可以由应用程序自动登录系统来执行。如图1和2A所示，该方法具体包括：

[0040] S110，在服务端检测到用户端通过浏览器登录目标应用程序时，生成、存储并加密登录凭证，且将加密的登录凭证发送给本地浏览器。

[0041] 本实施方式中的服务端可以为处理业务的服务器，例如可以为生成加密登录凭证提供决策，为登录本地目标应用程序提供决策的服务终端，也可以为处理综合业务的服务器。

[0042] 浏览器是用于显示网站服务器或文件系统内的文件，并让用户与这些文件交互的一种应用软件，可用来显示在万维网或局域网等内的文字、图像及其他信息，这些文字或图像可以是连接其他网址的超链接，用户可迅速及轻易地浏览各种信息。

[0043] 登录凭证是指服务器依据应用程序提供给用户注册的规则随机生成的唯一的字符串；不同的应用程序对应的登录凭证不同。

[0044] 例如，当用户通过浏览器输入用户名和密码登录阿里巴巴时，管理阿里巴巴的后端服务器检测到该操作，将对用户的用户名和密码进行验证，若通过，则阿里巴巴的后端服务器就会随机生成一串字符串作为登录凭证，并对该登录凭证进行加密后发送给用户持有的客户机器上的网页版的阿里巴巴，同时还可记录该用户登录的IP(Internet Protocol，互联网协议)地址。其中，目标应用程序程序为阿里巴巴，本地浏览器为用户持有的客户机器上的网页版的阿里巴巴。

[0045] S120，在本地浏览器检测到用户通过本地浏览器调用本地目标应用程序时，将加

密的登录凭证发送给本地目标应用程序。

[0046] 其中，本地目标应用程序为用户持有的客户机器上所安装的客户端程序。

[0047] 具体的操作过程为：当用户为了进一步了解更多的内容，在本地浏览器上通过超链接控件的点击事件或按钮控件的点击事件来调用客户机器上已经安装的客户端程序即本地目标应用程序时，本地浏览器检测到该事件，将把从服务端接收到的加密的登录凭证发送给本地目标应用程序。

[0048] S130，本地目标应用程序解密获得的登录凭证，对解密的登录凭证和本地端的唯一标识符进行加密，并依据加密结果生成登录请求，且将生成的登录请求发送给服务端。

[0049] 其中，本地端的唯一标识符即为客户机器的唯一硬件标识符，是根据硬件序列号生成的注册号，如主板BIOS(Basic Input/Output System，基本输入输出系统)序列号；登录请求是指用于登录本地目标应用程序的请求。

[0050] 具体的，本地目标应用程序接收到本地浏览器发送的加密的登录凭证后，首先对其进行解密操作，取得用户名及登录凭证；然后从客户机器上相应的存储位置上获取本地端的唯一标识符；最后对解密的登录凭证和本地端的唯一标识符重新进行加密，生成登录请求，并发送给服务器进行验证。

[0051] S140，服务端依据存储的登录凭证以及预先获取的本地端的唯一标识符对登录请求进行验证，若验证成功，则进行登录授权。

[0052] 服务端接收到本地目标应用程序发送的登录请求后，先将当前的IP地址与先前记录的IP地址进行比对即进行初步验证，若相同，则对登录请求进行解密，获得用户名、登录凭证和本地端的唯一标识符。将解密得到的登录凭证和本地端的唯一标识符分别与存储的登录凭证以及预先获取的本地端的唯一标识符进行比对，若都相同，则验证成功，对本地目标应该程序进行登录授权；若验证失败，服务器将发出异常登录报警信息。

[0053] 示例性的，服务端依据登录凭证以及预先获取的本地端的唯一标识符对登录请求进行验证的具体操作过程可以是：若登录请求中的登录凭证与服务端中预先存储的登录凭证相同，且登录请求中的唯一标识符与服务端预先获取的本地端的唯一标识符相同，则确定验证成功；否则验证失败。

[0054] 示例性的，在服务端检测到用户端通过浏览器登录目标应用程序时，生成、存储并加密登录凭证之前，还可以包括：若服务端检测到本地目标应用程序上报的绑定请求，则获取并存储本地目标应用程序上报的本地端的唯一标识符。

[0055] 其中，绑定请求是指将用户名与客户机器进行绑定的请求，该绑定请求中包含有本地端的唯一标识符和用户名等。具体的，本地目标应用程序安装在客户机器上后，在操作系统公开的相关位置上注册URL协议头对应的唤起程序。当用户第一次使用自己账户登录成功之后，本地目标应用程序弹框提示用户，是否绑定该客户机器，以放入信任列表中。如果用户选择是，则将绑定请求发送给服务端。服务端接收到该绑定请求后，对该请求进行解密，获取并存储本地目标应用程序对应的本地端的唯一标识符，同时，向本地目标应用程序发送绑定成功确认信息。

[0056] 参见图2B，现有的技术中，当客户机器遭到攻击被注入爬虫或木马等后，窃密者可以通过获得并更改用户PC注册表的权限，将URL协议头指向窃密者种下的木马程序；木马程序截获浏览器得到URL，并发送给远程窃密用的PC；通过将该URL发送给远程PC上的本地目

标应用程序，尝试进行获取登录权限；窃密者能够进行IP欺骗，伪装成正常用户IP，将本地目标应用程序获取的登录请求发送给服务端，获取新的登录态，成功后就可以获取相应权限操作账户。而本实施例中存在绑定校验，所以本地目标应用程序将不会收到启动命令或者发送登录请求即登录状态不存在被窃取的风险。

[0057] 本发明实施例提供的应用程序自动登录的方法，当用户通过浏览器登录目标应用程序时，服务器检测到此操作后，生成、存储并加密登录凭证，同时将加密的登录凭证发送给本地浏览器；本地浏览器将加密的登录凭证发送给本地目标应用程序，本地目标应用程序对解密的登录凭证和本地端的唯一标识符进行加密，从而生成登录请求，且将登录请求发送给服务端；服务端对登录请求进行验证，若验证成功，则进行登录授权。解决了在客户机器上记住密客户端软件的登录密码这种方式，如果记住密码所在的文件/文件夹被不小心擦除，或者用户在其他地方修改过密码，则需要重新输入密码进行登录的问题。可实现安全的将登录状态从浏览器同步到客户端程序上，减少用户的重复输入。

[0058] 实施例二

[0059] 图3为本发明实施例二提供的一种应用程序自动登录的方法的流程图。本实施例在上述实施例的基础上进一步地优化。如图3所示，该方法具体包括：

[0060] S310，在服务端检测到用户端通过浏览器登录目标应用程序时，获取并存储当前的登录时间戳，同时生成、存储并加密登录凭证，且将加密的登录凭证发送给本地浏览器。

[0061] 其中，时间戳是用来表示时间的标识，即用户通过浏览器登录目标应用程序对应的时间。

[0062] S320，在本地浏览器检测到用户通过本地浏览器调用本地目标应用程序时，将加密的登录凭证发送给本地目标应用程序。

[0063] S330，本地目标应用程序解密获得的登录凭证，对解密的登录凭证和本地端的唯一标识符进行加密，并依据加密结果生成登录请求，且将生成的登录请求发送给服务端。

[0064] S340，若登录请求中的登录凭证与服务端中预先存储的登录凭证相同，登录请求中的唯一标识符与服务端预先获取的本地端的唯一标识符相同，以及登录请求的发送时间与服务端中预先存储的登录时间戳之间的差值小于时间阈值，则确定验证成功；否则，验证失败。

[0065] 为了进一步地保证安全的将登录状态从浏览器同步到本地目标应用程序上，服务器在对解密后的登录请求中的登录凭证以及唯一标识符进行验证的同时，获取本地目标应用程序发送登录请求的时间和预先存储的登录时间戳，将两者做差，若差值小于预先设定的时间阈值即差值处于符合规定的一个较小的范围内，则验证成功，可进行登录授权；若两者的差值大于时间阈值，则验证失败，服务器将会发出异常登录报警信息。其中，时间阈值是预先设置的，可根据实际的应该环境情况进行修正。

[0066] S350，若验证成功，则进行登录授权。

[0067] 本发明实施例提供的应用程序自动登录的方法，当用户通过浏览器登录目标应用程序时，服务器检测到此操作后，生成、存储并加密登录凭证，同时将加密的登录凭证发送给本地浏览器；本地浏览器将加密的登录凭证发送给本地目标应用程序，本地目标应用程序对解密的登录凭证和本地端的唯一标识符进行加密，从而生成登录请求，且将登录请求发送给服务端；服务端对登录请求进行验证，若验证成功，则进行登录授权。解决了在客户

机器上记住密客户端软件的登录密码这种方式，如果记住密码所在的文件/文件夹被不小心擦除，或者用户在其他地方修改过密码，则需要重新输入密码进行登录的问题。可实现安全的将登录状态从浏览器同步到客户端程序上，减少用户的重复输入。

[0068] 实施例三

[0069] 图4为本发明实施例三提供的一种应用程序自动登录的方法的流程图。本实施例可适用于用户通过点击网页链接调起应用程序并自动登录的情况。整套应用程序自动登录的方法通常由服务端以及客户机器上的本地浏览器和本地目标应用程序配合执行，本实施例所提供的方法可以由客户机器上的本地目标应用程序来执行。如图4所示，该方法具体包括：

[0070] S410，在检测到本地浏览器的调用请求时，从本地浏览器获取加密的登录凭证。

[0071] 其中，加密的登录凭证是在服务端检测到用户端通过浏览器登录目标应用程序时，生成、存储、加密并下发到本地浏览器中的。具体的，登录凭证可以为服务器依据应用程序提供给用户注册的规则随机生成的唯一的字符串；不同的应用程序对应的登录凭证不同。

[0072] 调用请求是指本地浏览器调用本地目标应用程序时所发出的请求。如用户在其持有的客户机器上的网页版的阿里巴巴上点击超链接控件或跳转按钮以跳转到客户端阿里巴巴所发出的请求。

[0073] S420，解密获取的登录凭证，并对登录凭证和本地端的唯一标识符进行加密，依据加密结果生成登录请求，且将登录请求发送给服务端。

[0074] 其中，登录请求用于指示服务端依据存储的登录凭证以及预先获取的本地端的唯一标识符对登录请求进行验证，若验证成功，则进行登录授权。

[0075] 其中，本地端的唯一标识符即为客户机器的唯一硬件标识符，是根据硬件序列号生成的注册号，如主板BIOS序列号；登录请求是指用于登录本地目标应用程序的请求。

[0076] 示例性的，服务端依据登录凭证以及预先获取的本地端的唯一标识符对登录请求进行验证的具体操作过程可以是：若登录请求中的登录凭证与服务端中预先存储的登录凭证相同，且登录请求中的唯一标识符与服务端预先获取的本地端的唯一标识符相同，则确定验证成功；否则验证失败。

[0077] 若服务端上存储有当前的登录时间戳，则服务端依据登录凭证以及预先获取的本地端的唯一标识符对登录请求进行验证的具体操作过程还可以是：若登录请求中的登录凭证与服务端中预先存储的登录凭证相同，登录请求中的唯一标识符与服务端预先获取的本地端的唯一标识符相同，以及登录请求的发送时间与服务端中预先存储的登录时间戳之间的差值小于时间阈值，则确定验证成功；否则，验证失败。

[0078] 为了避免登录状态存在被窃取的风险，可选的，上述方法还可以包括：将绑定请求上报给服务端，以指示服务端对该绑定请求进行验证，以获取并存储绑定请求中的本地端的唯一标识符。

[0079] 本发明实施例提供的应用程序自动登录的方法，当客户机器上的本地目标应用程序检测到本地浏览器调用该本地目标应用程序发出的调用请求时，从本地浏览器获取加密的登录凭证，并对其进行解密，同时将解密的登录凭证和本地端的唯一标识符进行加密，生成登录请求，且将登录请求发送给服务端，以指示服务端对登录请求进行验证，若验证成

功，则进行登录授权。可实现安全的将登录状态从浏览器同步到客户端程序上，减少用户的重复输入。

[0080] 实施例四

[0081] 图5为本发明实施例四提供的一种应用程序自动登录的方法的示意图，该方法是在上述实施例的基础上，对服务端以及客户机器上的本地浏览器和本地目标应用程序三个之间的交互提供的一个优选示例。如图5所示，整体交互过程具体包括如下：

[0082] A，本地目标应用程序即客户端程序安装在客户机器上后，在操作系统公开的相关位置上注册URL协议头对应的唤起程序。当用户第一次使用自己账户登录成功之后，本地目标应用程序弹框提示用户，是否绑定该客户机器，以放入信任列表中。如果用户选择是，则将绑定请求发送给服务端。其中，绑定请求是对本地端的唯一标识符和用户名进行加密后得到的。如图中的标号1。

[0083] B，服务端接收到本地目标应用程序上报的绑定请求后，对该绑定请求进行解密，获取并存储本地目标应用程序上报的本地端的唯一标识符和用户名。如图中的标号2。

[0084] C，服务端向本地目标应用程序发送绑定成功确认信息。此时，用户可正常使用该本地目标应用程序。如图中的标号3。

[0085] D，在服务端检测到用户通过浏览器访问目标应用程序，并输入用户名和密码进行登录。如图中的标号4。

[0086] E，服务端对接收到的用户名和密码进行验证，当验证通过后，获取并存储当前的登录时间戳以及该用户的IP地址，同时生成、存储并加密登录凭证，且将加密的登录凭证发送给本地浏览器。如图中的标号5和6。

[0087] F，本地浏览器在检测到用户通过本地浏览器调用本地目标应用程序时，将加密的登录凭证发送给本地目标应用程序。如图中的标号7。

[0088] G，本地目标应用程序对本地浏览器发送的加密登录凭证进行解密，取得用户名及登录凭证，并从客户机器上相应的存储位置上获取本地端的唯一标识符，对登录凭证和本地端的唯一标识符进行重新加密后生成登录请求并发送给服务端。如图中的标号8和9。

[0089] H，服务端接收到该本地目标应用程序发送的登录请求后，首先将当前的IP地址与E记录的IP地址进行比对，若相同，则对获取的登录请求进行解密，获得用户名、登录凭证和本地端的唯一标识符，并查看登录请求中的登录凭证是否与服务端中预先存储的登录凭证相同，登录请求中的唯一标识符是否与服务端预先获取的本地端的唯一标识符相同，以及登录请求的发送时间与服务端中预先存储的登录时间戳之间的差值是否小于时间阈值。如图中的标号10。I，当服务器验证本地应用程序发送的登录请求同时满足H中的条件后，将生成新的登录凭证，并将该新的登录凭证发送给本地目标应用程序，完成登录授权。同时，服务端将会把先前存储的登录凭证作废，当再次收到先前存储的登录凭证的更新登录凭证请求时，则拒绝以防止其他客户机器重试，且后续交互正常使用新的登录凭证进行校验。如图中的标号11。

[0090] 本发明实施例提供的应用程序自动登录的方法，当用户通过浏览器登录目标应用程序时，服务器检测到此操作后，生成、存储并加密登录凭证，同时将加密的登录凭证发送给本地浏览器；本地浏览器将加密的登录凭证发送给本地目标应用程序，本地目标应用程序对解密的登录凭证和本地端的唯一标识符进行加密，从而生成登录请求，且将登录请求

发送给服务端;服务端对登录请求进行验证,若验证成功,则进行登录授权。解决了在客户机器上记住密客户端软件的登录密码这种方式,如果记住密码所在的文件/文件夹被不小心擦除,或者用户在其他地方修改过密码,则需要重新输入密码进行登录的问题。可实现安全的将登录状态从浏览器同步到客户端程序上,减少用户的重复输入。

[0091] 实施例五

[0092] 图6为本发明实施例五提供的一种应用程序自动登录的系统的结构框图,该系统可用于执行本发明任意实施例提供的应用程序自动登录的方法,具备执行方法相应的功能模块和有益效果。具体是该系统100包括:服务端101、本地浏览器102和本地目标应用程序103;其中,

[0093] 服务端101用于在检测到用户端通过浏览器登录目标应用程序时,生成、存储并加密登录凭证,且将加密的登录凭证发送给本地浏览器20;

[0094] 本地浏览器102用于在检测到用户通过本地浏览器102调用本地目标应用程序103时,将加密的登录凭证发送给本地目标应用程序103;

[0095] 本地目标应用程序103用于解密获得的登录凭证,对解密的登录凭证和本地端的唯一标识符进行加密,并依据加密结果生成登录请求,且将生成的登录请求发送给服务端101;

[0096] 服务端101还用于依据存储的登录凭证以及预先获取的本地端的唯一标识符对登录请求进行验证,若验证成功,则进行登录授权。

[0097] 本发明实施例提供应用程序自动登录的系统,当用户通过浏览器登录目标应用程序时,服务器检测到此操作后,生成、存储并加密登录凭证,同时将加密的登录凭证发送给本地浏览器;本地浏览器将加密的登录凭证发送给本地目标应用程序,本地目标应用程序对解密的登录凭证和本地端的唯一标识符进行加密,从而生成登录请求,且将登录请求发送给服务端;服务端对登录请求进行验证,若验证成功,则进行登录授权。解决了在客户机器上记住密客户端软件的登录密码这种方式,如果记住密码所在的文件/文件夹被不小心擦除,或者用户在其他地方修改过密码,则需要重新输入密码进行登录的问题。可实现安全的将登录状态从浏览器同步到客户端程序上,减少用户的重复输入。

[0098] 示例性的,服务端101还用于在检测到用户端通过浏览器登录目标应用程序时,生成、存储并加密登录凭证之前,若检测到本地目标应用程序30上报的绑定请求,则获取并存储本地目标应用程序103上报的本地端的唯一标识符。

[0099] 可选的,服务端101还用于通过如下方式对登录请求进行验证:

[0100] 若登录请求中的登录凭证与服务端101中预先存储的登录凭证相同,且登录请求中的唯一标识符与服务端101预先获取的本地端的唯一标识符相同,则确定验证成功;否则验证失败。

[0101] 示例性的,服务端101还用于获取并存储当前的登录时间戳;

[0102] 对应的,服务端101还用于通过如下方式对登录请求进行验证:

[0103] 若登录请求中的登录凭证与服务端101中预先存储的登录凭证相同,登录请求中的唯一标识符与服务端预先获取的本地端的唯一标识符相同,以及登录请求的发送时间与服务端中预先存储的登录时间戳之间的差值小于时间阈值,则确定验证成功;否则,验证失败。

[0104] 实施例六

[0105] 图7为本发明实施例六提供的一种应用程序自动登录的装置的结构框图，该装置可执行本发明实施例三所提供的应用程序自动登录的方法，具备执行方法相应的功能模块和有益效果。如图7所示，该装置可以包括：

[0106] 登录凭证获取模块710，用于在检测到本地浏览器的调用请求时，从本地浏览器获取加密的登录凭证，其中加密的登录凭证是在服务端检测到用户端通过浏览器登录目标应用程序时，生成、存储、加密并下发到本地浏览器中的；

[0107] 登录请求发送模块720，用于解密获取的登录凭证，并对登录凭证和本地端的唯一标识符进行加密，依据加密结果生成登录请求，且将登录请求发送给服务端，其中登录请求用于指示服务端依据存储的登录凭证以及预先获取的本地端的唯一标识符对登录请求进行验证，若验证成功，则进行登录授权。

[0108] 本发明实施例提供应用程序自动登录的装置，当客户机器上的本地目标应用程序检测到本地浏览器调用该本地目标应用程序发出的调用请求时，从本地浏览器获取加密的登录凭证，并对其进行解密，同时将解密的登录凭证和本地端的唯一标识符进行加密，生成登录请求，且将登录请求发送给服务端，以指示服务端对登录请求进行验证，若验证成功，则进行登录授权。可实现安全的将登录状态从浏览器同步到客户端程序上，减少用户的重复输入。

[0109] 实施例七

[0110] 图8为本发明实施例七提供的一种设备的结构示意图。图8示出了适于用来实现本发明实施方式的示例性设备12的框图。图8显示的设备12仅仅是一个示例，不应对本发明实施例的功能和使用范围带来任何限制。

[0111] 如图8所示，该设备12以通用计算设备的形式表现。该设备12的组件可以包括但不限于：一个或者多个处理器或者处理单元16，系统存储器28，连接不同系统组件(包括系统存储器28和处理单元16)的总线18。

[0112] 总线18表示几类总线结构中的一种或多种，包括存储器总线或者存储器控制器，外围总线，图形加速端口，处理器或者使用多种总线结构中的任意总线结构的局域总线。举例来说，这些体系结构包括但不限于工业标准体系结构(ISA)总线，微通道体系结构(MAC)总线，增强型ISA总线、视频电子标准协会(VESA)局域总线以及外围组件互连(PCI)总线。

[0113] 设备12典型地包括多种计算机系统可读介质。这些介质可以是任何能够被设备12访问的可用介质，包括易失性和非易失性介质，可移动的和不可移动的介质。

[0114] 系统存储器28可以包括易失性存储器形式的计算机系统可读介质，例如随机存取存储器(RAM)30和/或高速缓存存储器32。设备12可以进一步包括其它可移动/不可移动的、易失性/非易失性计算机系统存储介质。仅作为举例，存储系统34可以用于读写不可移动的、非易失性磁介质(图8未显示，通常称为“硬盘驱动器”)。尽管图8中未示出，可以提供用于对可移动非易失性磁盘(例如“软盘”)读写的磁盘驱动器，以及对可移动非易失性光盘(例如CD-ROM，DVD-ROM或者其它光介质)读写的光盘驱动器。在这些情况下，每个驱动器可以通过一个或者多个数据介质接口与总线18相连。系统存储器28可以包括至少一个程序产品，该程序产品具有一组(例如至少一个)程序模块，这些程序模块被配置以执行本发明各实施例的功能。

[0115] 具有一组(至少一个)程序模块42的程序/实用工具40,可以存储在例如系统存储器28中,这样的程序模块42包括但不限于操作系统、一个或者多个应用程序、其它程序模块以及程序数据,这些示例中的每一个或某种组合中可能包括网络环境的实现。程序模块42通常执行本发明所描述的实施例中的功能和/或方法。

[0116] 设备12也可以与一个或多个外部设备14(例如键盘、指向设备、显示器24等)通信,还可与一个或者多个使得用户能与该设备交互的设备通信,和/或与使得该设备12能与一个或多个其它计算设备进行通信的任何设备(例如网卡,调制解调器等等)通信。这种通信可以通过输入/输出(I/O)接口22进行。并且,设备12还可以通过网络适配器20与一个或者多个网络(例如局域网(LAN),广域网(WAN)和/或公共网络,例如因特网)通信。如图所示,网络适配器20通过总线18与设备12的其它模块通信。应当明白,尽管图5中未示出,可以结合设备12使用其它硬件和/或软件模块,包括但不限于:微代码、设备驱动器、冗余处理单元、外部磁盘驱动阵列、RAID系统、磁带驱动器以及数据备份存储系统等。

[0117] 处理单元16通过运行存储在系统存储器28中的程序,从而执行各种功能应用以及数据处理,例如实现本发明实施例三所提供的应用程序自动登录的方法。

[0118] 实施例八

[0119] 本发明实施例八还提供了一种计算机可读存储介质,其上存储有计算机程序,该程序被处理器执行时可实现上述实施例三中应用程序自动登录的方法。

[0120] 本发明实施例的计算机存储介质,可以采用一个或多个计算机可读的介质的任意组合。计算机可读介质可以是计算机可读信号介质或者计算机可读存储介质。计算机可读存储介质例如可以是但不限于电、磁、光、电磁、红外线、或半导体的系统、装置或器件,或者任意以上的组合。计算机可读存储介质的更具体的例子(非穷举的列表)包括:具有一个或多个导线的电连接、便携式计算机磁盘、硬盘、随机存取存储器(RAM)、只读存储器(ROM)、可擦式可编程只读存储器(EPROM或闪存)、光纤、便携式紧凑磁盘只读存储器(CD-ROM)、光存储器件、磁存储器件、或者上述的任意合适的组合。在本文件中,计算机可读存储介质可以是任何包含或存储程序的有形介质,该程序可以被指令执行系统、装置或者器件使用或者与其结合使用。

[0121] 计算机可读的信号介质可以包括在基带中或者作为载波一部分传播的数据信号,其中承载了计算机可读的程序代码。这种传播的数据信号可以采用多种形式,包括但不限于电磁信号、光信号或上述的任意合适的组合。计算机可读的信号介质还可以是计算机可读存储介质以外的任何计算机可读介质,该计算机可读介质可以发送、传播或者传输用于由指令执行系统、装置或者器件使用或者与其结合使用的程序。

[0122] 计算机可读介质上包含的程序代码可以用任何适当的介质传输,包括但不限于无线、电线、光缆、RF等等,或者上述的任意合适的组合。

[0123] 可以以一种或多种程序设计语言或其组合来编写用于执行本发明操作的计算机程序代码,所述程序设计语言包括面向对象的程序设计语言—诸如Java、Smalltalk、C++,还包括常规的过程式程序设计语言—诸如“C”语言或类似的程序设计语言。程序代码可以完全地在用户计算机上执行、部分地在用户计算机上执行、作为一个独立的软件包执行、部分在用户计算机上部分在远程计算机上执行、或者完全在远程计算机或服务器上执行。在涉及远程计算机的情形中,远程计算机可以通过任意种类的网络包括局域网(LAN)或广域

网(WAN)连接到用户计算机,或者,可以连接到外部计算机(例如利用因特网服务提供商来通过因特网连接)。

[0124] 上述实施例序号仅仅为了描述,不代表实施例的优劣。

[0125] 本领域普通技术人员应该明白,上述的本发明的各模块或各可以用通用的计算装置来实现,它们可以集中在单个计算装置上,或者分布在多个计算装置所组成的网络上,可选地,他们可以用计算机装置可执行的程序代码来实现,从而可以将它们存储在存储装置中由计算装置来执行,或者将它们分别制作成各个集成电路模块,或者将它们中的多个模块或制作成单个集成电路模块来实现。这样,本发明不限制于任何特定的硬件和软件的结合。

[0126] 本说明书中的各个实施例均采用递进的方式描述,每个实施例重点说明的都是与其他实施例的不同之处,各个实施例之间的相同或相似的部分互相参见即可。

[0127] 以上所述仅为本发明的优选实施例,并不用于限制本发明,对于本领域技术人员而言,本发明可以有各种改动和变化。凡在本发明的精神和原理之内所作的任何修改、等同替换、改进等,均应包含在本发明的保护范围之内。

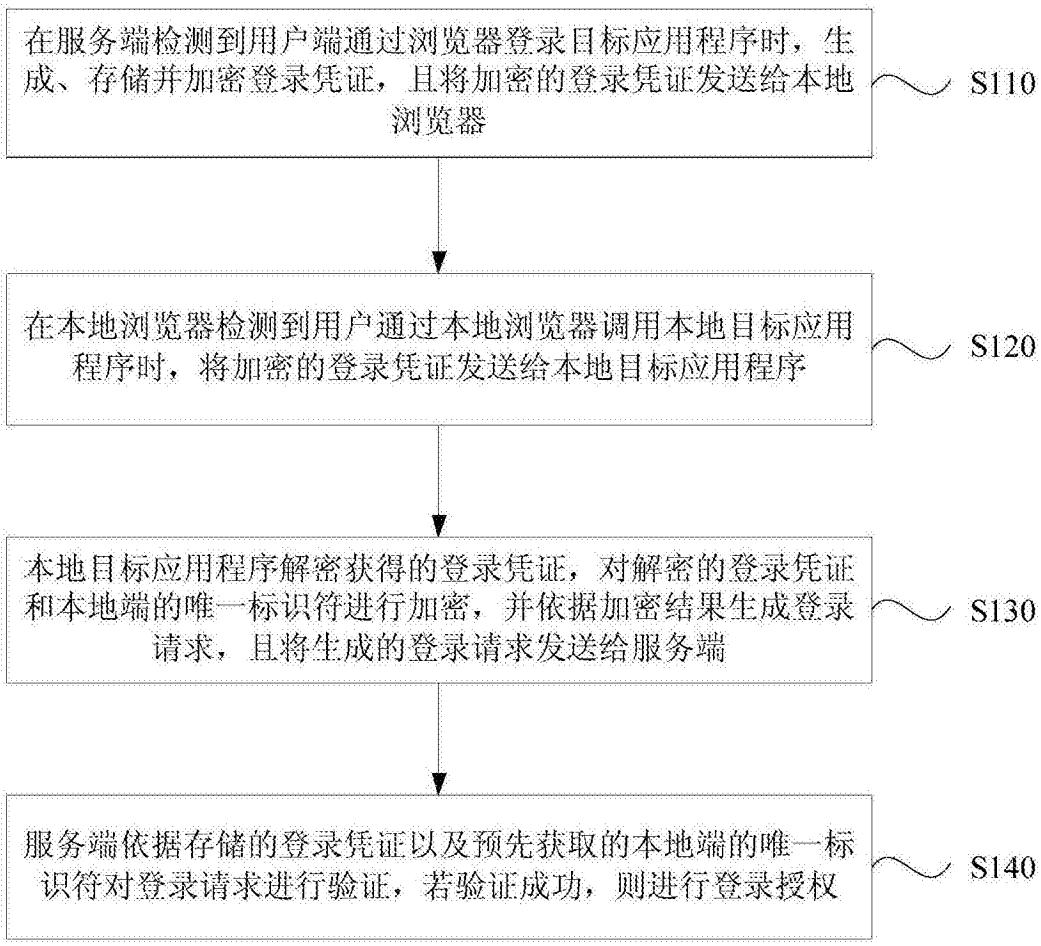

在服务端检测到用户端通过浏览器登录目标应用程序时，生成、存储并加密登录凭证，且将加密的登录凭证发送给本地浏览器
S110
在本地浏览器检测到用户通过本地浏览器调用本地目标应用程序时，将加密的登录凭证发送给本地目标应用程序
S120
本地目标应用程序解密获得的登录凭证，对解密的登录凭证和本地端的唯一标识符进行加密，并依据加密结果生成登录请求，且将生成的登录请求发送给服务端
S130
服务端依据存储的登录凭证以及预先获取的本地端的唯一标识符对登录请求进行验证，若验证成功，则进行登录授权
S140

图1

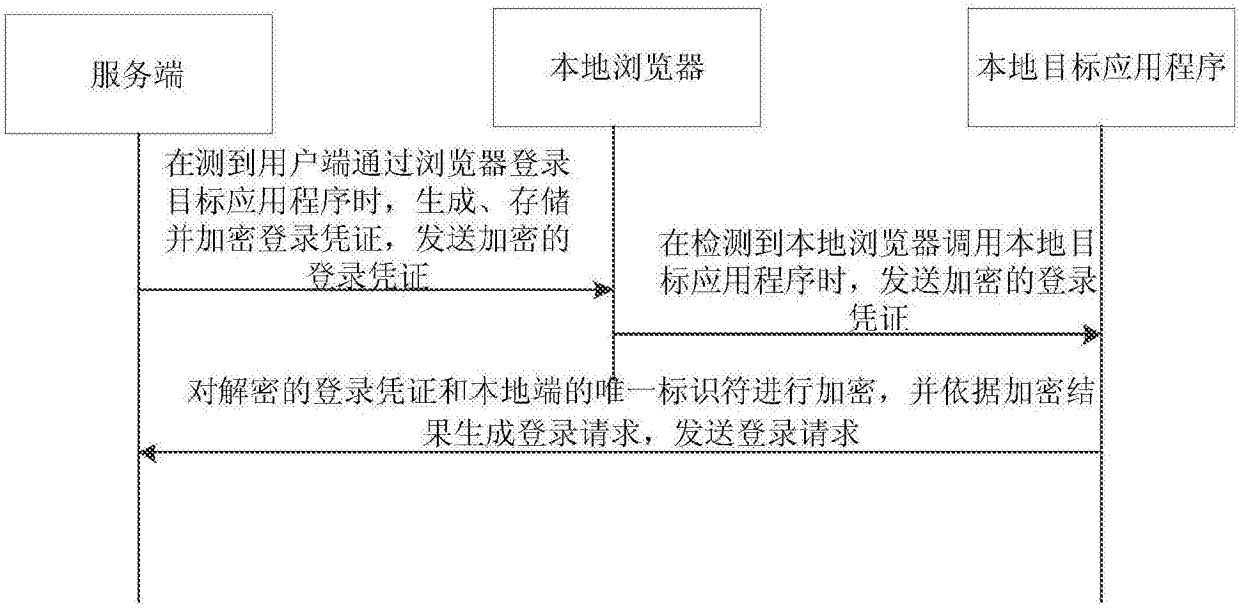

服务端
本地浏览器
本地目标应用程序
在测到用户端通过浏览器登录目标应用程序时，生成、存储并加密登录凭证，发送加密的登录凭证
在检测到本地浏览器调用本地目标应用程序时，发送加密的登录凭证
对解密的登录凭证和本地端的唯一标识符进行加密，并依据加密结果生成登录请求，发送登录请求

图2A

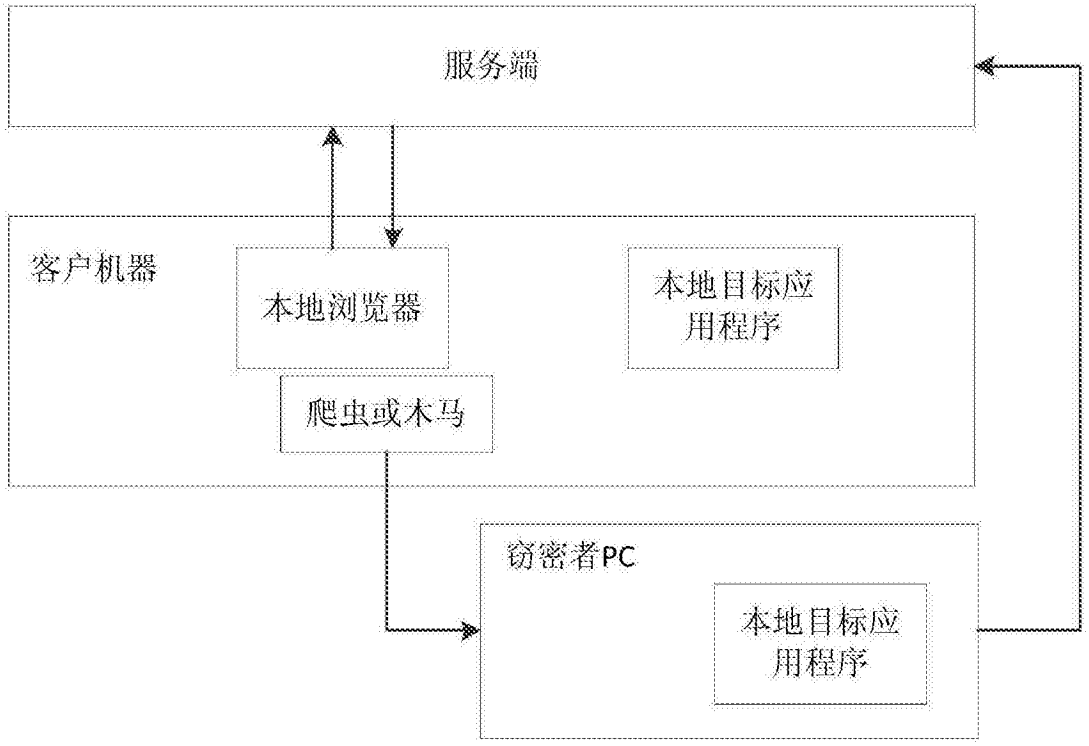


图2B

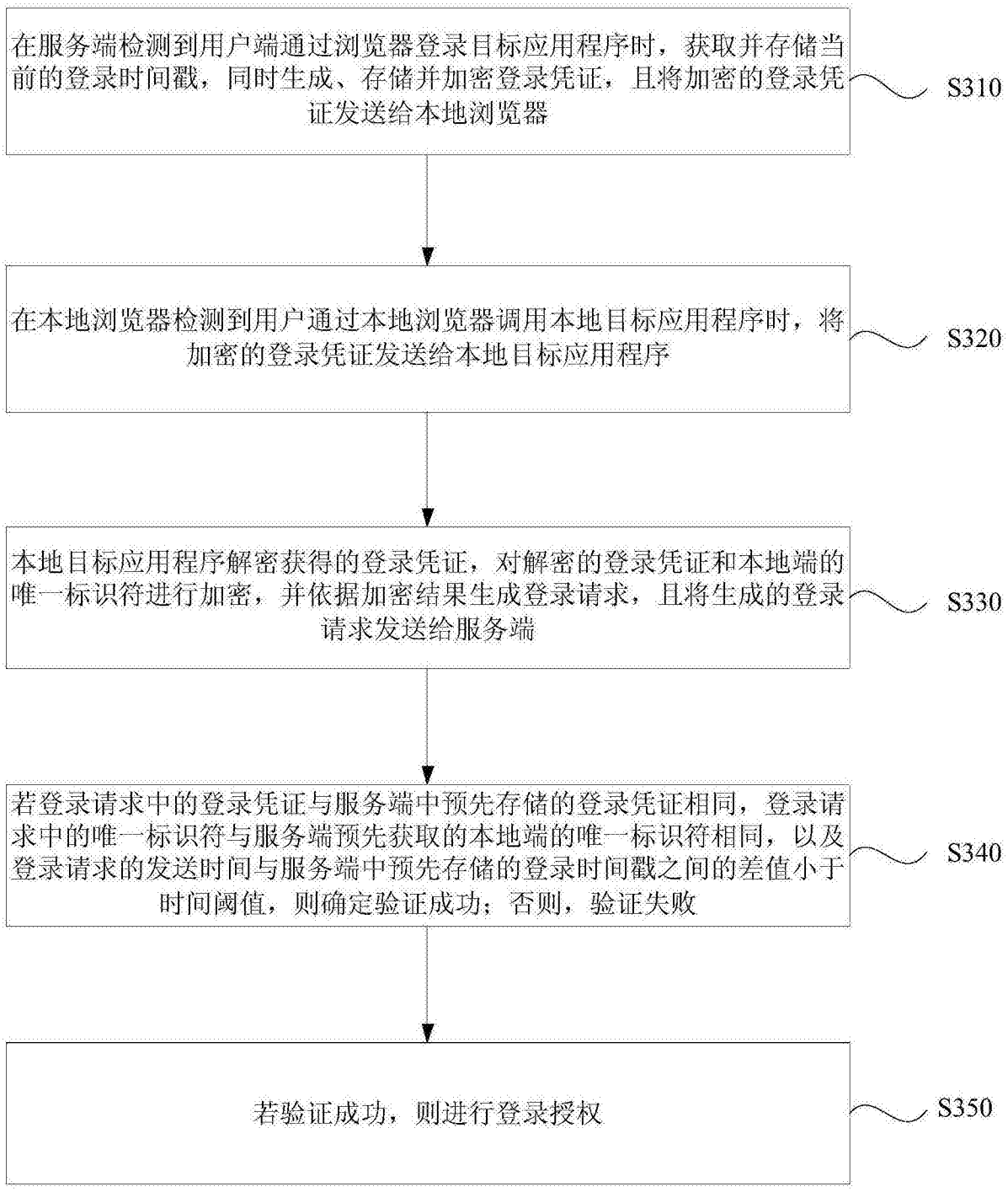

在服务端检测到用户端通过浏览器登录目标应用程序时，获取并存储当前的登录时间戳，同时生成、存储并加密登录凭证，且将加密的登录凭证发送给本地浏览器
S310
在本地浏览器检测到用户通过本地浏览器调用本地目标应用程序时，将加密的登录凭证发送给本地目标应用程序
S320
本地目标应用程序解密获得的登录凭证，对解密的登录凭证和本地端的唯一标识符进行加密，并依据加密结果生成登录请求，且将生成的登录请求发送给服务端
S330
若登录请求中的登录凭证与服务端中预先存储的登录凭证相同，登录请求中的唯一标识符与服务端预先获取的本地端的唯一标识符相同，以及登录请求的发送时间与服务端中预先存储的登录时间戳之间的差值小于时间阈值，则确定验证成功；否则，验证失败
S340
若验证成功，则进行登录授权
S350

图3

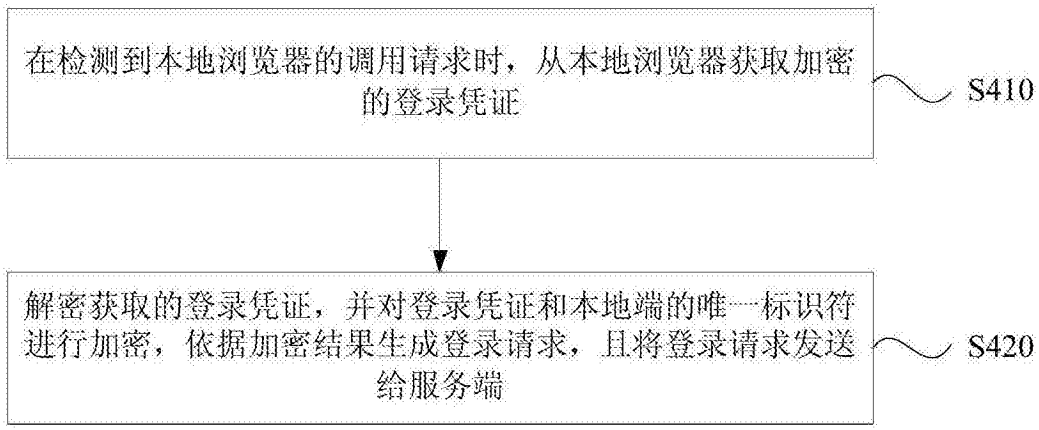

在检测到本地浏览器的调用请求时，从本地浏览器获取加密的登录凭证
S410
解密获取的登录凭证，并对登录凭证和本地端的唯一标识符进行加密，依据加密结果生成登录请求，且将登录请求发送给服务端
S420

图4

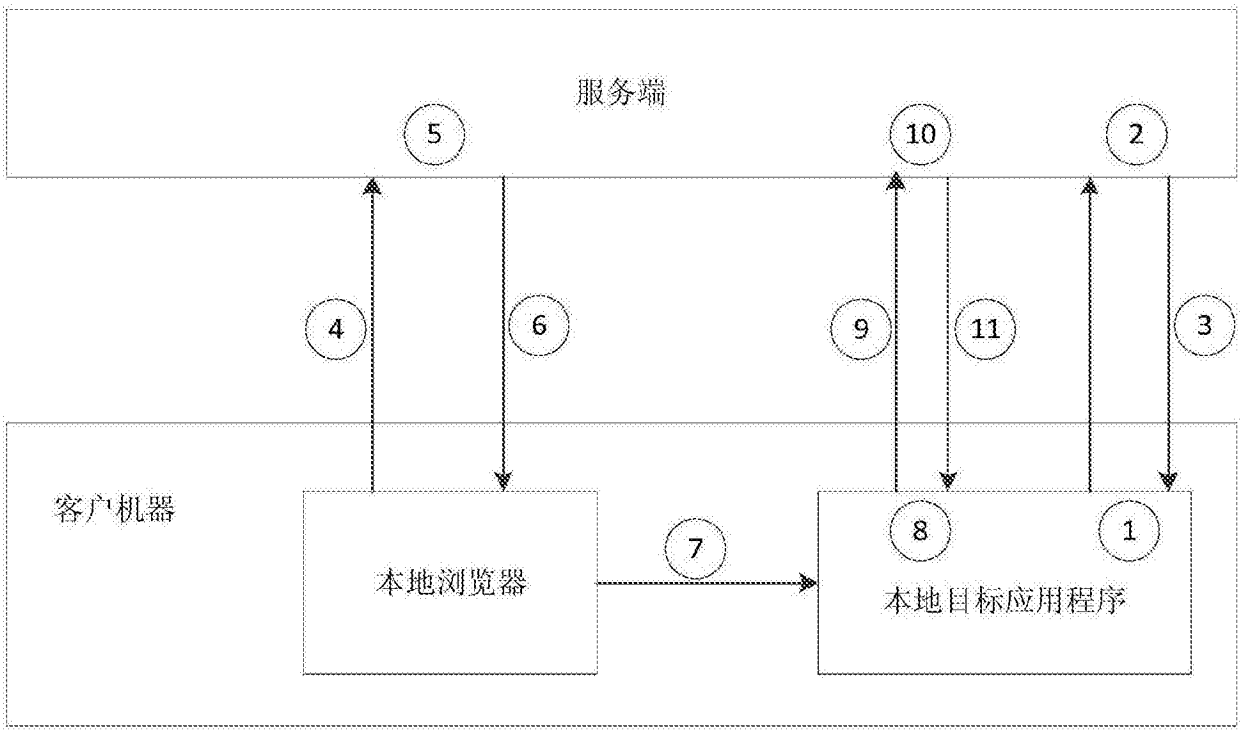

服务端
5
10
2
4
6
9
11
3
客户机器
本地浏览器
7
8
1
本地目标应用程序

图5

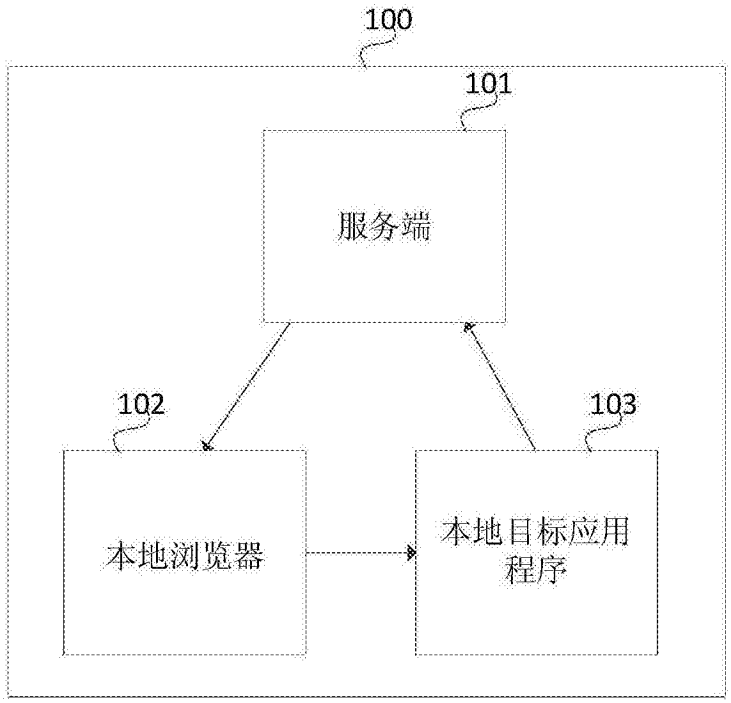

100
101
服务端
102
103
本地浏览器
本地目标应用程序

图6

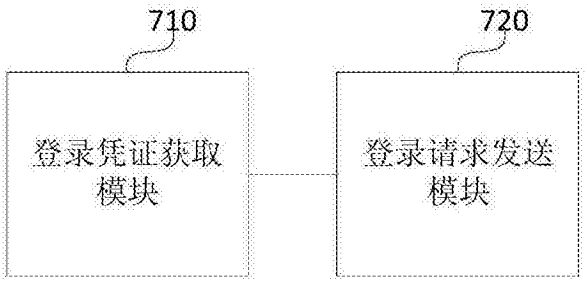

710
720
登录凭证获取模块
登录请求发送模块

图7

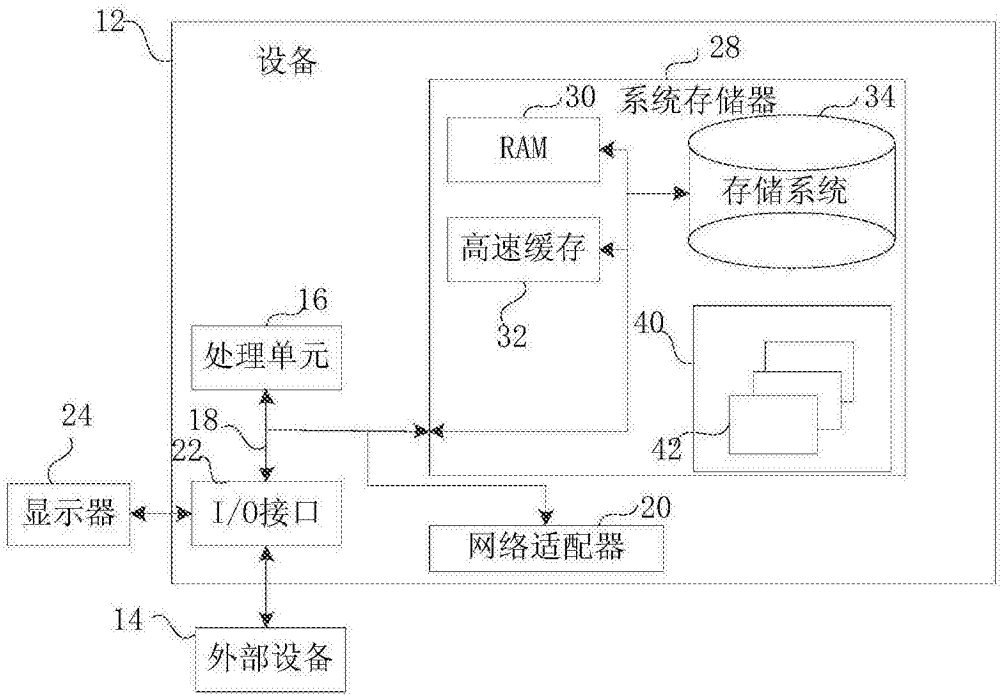

12
设备
28
30
系统存储器
34
RAM
存储系统
高速缓存
32
16
处理单元
40
42
24
18
22
显示器
I/O接口
20
网络适配器
14
外部设备

图8