(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 109617692 A
(43)申请公布日 2019.04.12

(21)申请号 201811526788.4

(22)申请日 2018.12.13

(71)申请人 郑州师范学院
地址 450044 河南省郑州市英才街6号
申请人 上海朝夕网络技术有限公司

(72)发明人 刘云霞 李汝佳 王永浩

(74)专利代理机构 武汉东喻专利代理事务所（普通合伙）42224
代理人 李佑宏

(51)Int.Cl.
H04L 9/32(2006.01)
H04L 29/06(2006.01)
H04L 29/08(2006.01)

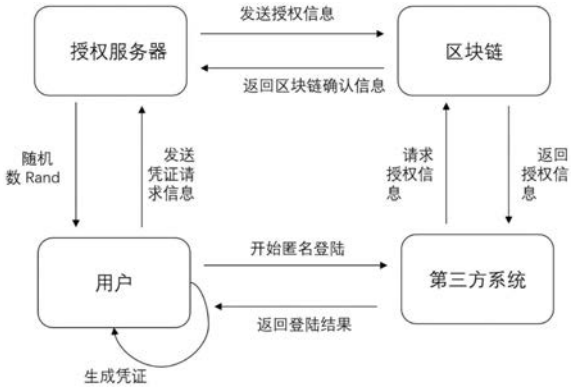
权利要求书2页 说明书6页 附图2页

(54)发明名称

一种基于区块链的匿名登陆方法及系统

(57)摘要

本发明公开了一种基于区块链的匿名登陆方法，包括对授权服务器进行初始化，获得的公钥生成对应的区块链地址，并将授权凭证发送给用户；枚举出用户的全部属性信息，获得用户的身份参数和校验参数，并利用授权凭证将其固化到区块链中；根据系统发布的登陆条件，选择所需要属性信息生成用户登陆凭证；系统接收用户登陆凭证，根据用户身份参数和/或校验参数对用户登陆凭证进行验证，确认当前用户是否满足登陆条件。本发明还公开了一种基于区块链的匿名登陆系统。本发明技术方案针对目前的匿名登陆系统的不足，采用哈希算法、非对称加密算法等对个人属性信息进行加密，并将其保存到区块链上，可以在保证用户身份安全认证的前提下最大限度保护用户的个人属性信息。



CN 109617692 A

1.一种基于区块链的匿名登陆方法,其特征在于,包括

S1对授权服务器进行初始化,根据初始化后获得的公钥生成对应的区块链地址,并将授权凭证发送给用户;

S2枚举出用户的全部属性信息,对属性信息进行哈希运算,获得用户的身份参数和校验参数,并利用授权凭证将其固化到区块链中;

S3根据系统发布的登陆条件,选择所需要属性信息生成用户登陆凭证,并将其提交给请求登陆的系统;

S4系统接收用户登陆凭证,根据用户身份参数和/或校验参数对用户登陆凭证进行验证,确认当前用户是否满足登陆条件。

2.根据权利要求1所述的一种基于区块链的匿名登陆方法,其中,所述步骤S1包括,

S11确定安全参数和/或加密算法,对授权服务器的公钥进行初始化并公开;

S12根据公钥获取对应的区块链地址,对应存储获得授权的用户信息;

S13授权服务器随机生成若干随机数并分配给用户,作为授权凭证。

3.根据权利要求1或2所述的一种基于区块链的匿名登陆方法,其中,所述步骤S2包括,

S21枚举出用户的全部属性信息,获取用户的属性信息列表;

S22根据安全参数,结合属性信息获取用户的校验参数;

S23将分配获得的授权凭证、身份参数和校验参数发送到授权服务器;

S24对授权通过的身份参数和校验参数进行签名处理,并固化保存到区块链中。

4.根据权利要求1～3任一项所述的一种基于区块链的匿名登陆方法,其中,所述步骤S3包括,

S31根据业务需求,公开登陆条件,所述登陆条件对用户的一个或多个属性信息提出要求;

S32请求登陆的用户根据登陆条件,选择对应的属性信息生成其所对应的哈希值和/或验证参数;

S33生成登陆凭证,所述登陆凭证包括符合登陆条件的属性信息、该属性信息的哈希值、验证参数以及用户的身份参数和校验参数。

5.根据权利要求1～4任一项所述的一种基于区块链的匿名登陆方法,其中,所述步骤S4包括,

S41接收用户登陆凭证,确定用户的身份参数和校验参数存储在区块链中;

S42对用户的属性信息进行哈希计算,确定其与验证参数中对应的属性哈希值相吻合;

S43根据用户的属性信息、校验参数以及验证参数,确定当前用户提供的登陆凭证与保存在区块链中的身份参数一致,即为验证通过。

6.一种基于区块链的匿名登陆系统,其特征在于,包括

初始模块,用于对授权服务器进行初始化,根据初始化后获得的公钥生成对应的区块链地址,并将授权凭证发送给用户;

授权模块,用于枚举出用户的全部属性信息,对属性信息进行哈希运算,获得用户的身份参数和校验参数,并利用授权凭证将其固化到区块链中;

登陆模块,用于根据系统发布的登陆条件,选择所需要属性信息生成用户登陆凭证,并将其提交给请求登陆的系统;

验证模块，用于系统接收用户登陆凭证，根据用户身份参数和/或校验参数对用户登陆凭证进行验证，确认当前用户是否满足登陆条件。

7.根据权利要求6所述的一种基于区块链的匿名登陆方法，其中，所述步骤初始模块包括，

初始化模块，用于确定安全参数和/或加密算法，对授权服务器的公钥进行初始化并公开；

区块链模块，用于根据公钥获取对应的区块链地址，对应存储获得授权的用户信息；

授权凭证模块，用于授权服务器随机生成若干随机数并分配给用户，作为授权凭证。

8.根据权利要求6或7所述的一种基于区块链的匿名登陆方法，其中，所述授权模块包括，

属性模块，用于枚举出用户的全部属性信息，获取用户的属性信息列表；

参数模块，用于根据安全参数，结合属性信息获取用户的校验参数；

请求模块，用于将分配获得的授权凭证、身份参数和校验参数发送到授权服务器；

签名模块，用于对授权通过的身份参数和校验参数进行签名处理，并固化保存到区块链中。

9.根据权利要求6～8任一项所述的一种基于区块链的匿名登陆方法，其中，所述登陆模块包括，

条件模块，用于根据业务需求，公开登陆条件，所述登陆条件对用户的一个或多个属性信息提出要求；

属性信息模块，用于请求登陆的用户根据登陆条件，选择对应的属性信息生成其所对应的哈希值和/或验证参数；

登陆凭证模块，用于生成登陆凭证，所述登陆凭证包括符合登陆条件的属性信息、该属性信息的哈希值、验证参数以及用户的身份参数和校验参数。

10.根据权利要求6～9任一项所述的一种基于区块链的匿名登陆方法，其中，所述验证模块包括，

参数验证模块，用于接收用户登陆凭证，确定用户的身份参数和校验参数存储在区块链中；

属性验证模块，用于对用户的属性信息进行哈希计算，确定其与验证参数中对应的属性哈希值相吻合；

身份验证模块，用于根据用户的属性信息、校验参数以及验证参数，确定当前用户提供的登陆凭证与保存在区块链中的身份参数一致，即为验证通过。

一种基于区块链的匿名登陆方法及系统

技术领域

[0001] 本发明属于计算机系统安全领域，具体涉及一种基于区块链的匿名登陆方法及系统。

背景技术

[0002] 进入21世纪，随着信息技术的不断发展，信息安全问题也日显突出。如何确保信息系统的安全已成为全社会关注的问题。信息安全主要包括以下五方面的内容，即需保证信息的保密性、真实性、完整性、未授权拷贝和所寄生系统的安全性。信息安全本身包括的范围很大，其中包括如何防范商业企业机密泄露、防范青少年对不良信息的浏览、个人信息的泄露等。

[0003] 因此，网络环境下的信息安全体系是保证信息安全的关键，包括计算机安全操作系统、各种安全协议、安全机制(数字签名、消息认证、数据加密等)，直至安全系统，如UniNAC、DLP等，只要存在安全漏洞便可以威胁全局安全。信息安全是指信息系统(包括硬件、软件、数据、人、物理环境及其基础设施)受到保护，不受偶然的或者恶意的原因而遭到破坏、更改、泄露，系统连续可靠正常地运行，信息服务不中断，最终实现业务连续性。

[0004] 在这种需求下，匿名登陆技术就显得十分必要了。匿名登陆是以匿名方式进入操作系统或者应用程序的过程，匿名登陆的情况下，访问请求人无须提交全部的个人信息。目前的匿名登陆方式有两种：有授权匿名登陆(如密码口令验证方案)和非授权的匿名登陆(如非法入侵)。有授权登陆的核心思想在于将用户的认证与用户的登陆进行分离，以OAuth 2.0用户开放授权的协议为例，用户的信息会被放置在中心化的授权服务器中，当用户需要访问第三方系统时，第三方系统要求用户像中心化的授权服务器获取授权令牌。

[0005] 但是在此过程中，存在以下问题：(1)中心化的授权服务器存储了用户的所有信息，该数据一旦被黑客攻陷所有信息将被暴露，目前因中心化服务器被攻陷而导致的用户信息泄露的事件近年来层出不穷；(2)此类登陆方案不提供匿名登陆的功能，如果授权服务器和第三方系统串通，则可以很轻易的溯源到此用户。也就是说，用户的信息仍然具有多个不受控的披露途径。

发明内容

[0006] 针对现有技术的以上缺陷或改进需求，本发明提供了一种基于区块链的匿名登陆方法，至少可以部分解决上述问题。本发明技术方案针对目前的匿名登陆系统仍然无法实现的情况，采用哈希算法，非对称加密算法等对个人属性信息进行加密，并将其保存到区块链上，可以在保证用户身份安全认证的前提下最大限度保护用户的个人属性信息。

[0007] 为实现上述目的，按照本发明的一个方面，提供了一种基于区块链的匿名登陆方法，其特征在于，包括

[0008] S1对授权服务器进行初始化，根据初始化后获得的公钥生成对应的区块链地址，并将授权凭证发送给用户；

[0009] S2枚举出用户的全部属性信息，对属性信息进行哈希运算，获得用户的身份参数和校验参数，并利用授权凭证将其固化到区块链中；

[0010] S3根据系统发布的登陆条件，选择所需要属性信息生成用户登陆凭证，并将其提交给请求登陆的系统；

[0011] S4系统接收用户登陆凭证，根据用户身份参数和/或校验参数对用户登陆凭证进行验证，确认当前用户是否满足登陆条件。

[0012] 作为本发明技术方案的一个优选，步骤S1包括，

[0013] S11确定安全参数和/或加密算法，对授权服务器的公钥进行初始化并公开；

[0014] S12根据公钥获取对应的区块链地址，对应存储获得授权的用户信息；

[0015] S13授权服务器随机生成若干随机数并分配给用户，作为授权凭证。

[0016] 作为本发明技术方案的一个优选，步骤S2包括，

[0017] S21枚举出用户的全部属性信息，获取用户的属性信息列表；

[0018] S22根据安全参数，结合属性信息获取用户的校验参数；

[0019] S23将分配获得的授权凭证、身份参数和校验参数发送到授权服务器；

[0020] S24对授权通过的身份参数和校验参数进行签名处理，并固化保存到区块链中。

[0021] 作为本发明技术方案的一个优选，步骤S3包括，

[0022] S31根据业务需求，公开登陆条件，所述登陆条件对用户的一个或多个属性信息提出要求；

[0023] S32请求登陆的用户根据登陆条件，选择对应的属性信息生成其所对应的哈希值和/或验证参数；

[0024] S33生成登陆凭证，所述登陆凭证包括符合登陆条件的属性信息、该属性信息的哈希值、验证参数以及用户的身份参数和校验参数。

[0025] 作为本发明技术方案的一个优选，步骤S4包括，

[0026] S41接收用户登陆凭证，确定用户的身份参数和校验参数存储在区块链中；

[0027] S42对用户的属性信息进行哈希计算，确定其与验证参数中对应的属性哈希值相吻合；

[0028] S43根据用户的属性信息、校验参数以及验证参数，确定当前用户提供的登陆凭证与保存在区块链中的身份参数一致，即为验证通过。

[0029] 按照本发明的一个方面，提供了一种基于区块链的匿名登陆系统，其特征在于，包括

[0030] 初始模块，用于对授权服务器进行初始化，根据初始化后获得的公钥生成对应的区块链地址，并将授权凭证发送给用户；

[0031] 授权模块，用于枚举出用户的全部属性信息，对属性信息进行哈希运算，获得用户的身份参数和校验参数，并利用授权凭证将其固化到区块链中；

[0032] 登陆模块，用于根据系统发布的登陆条件，选择所需要属性信息生成用户登陆凭证，并将其提交给请求登陆的系统；

[0033] 验证模块，用于系统接收用户登陆凭证，根据用户身份参数和/或校验参数对用户登陆凭证进行验证，确认当前用户是否满足登陆条件。

[0034] 作为本发明技术方案的一个优选，步骤初始模块包括，

[0035] 初始化模块，用于确定安全参数和/或加密算法，对授权服务器的公钥进行初始化并公开；

[0036] 区块链模块，用于根据公钥获取对应的区块链地址，对应存储获得授权的用户信息；

[0037] 授权凭证模块，用于授权服务器随机生成若干随机数并分配给用户，作为授权凭证。

[0038] 作为本发明技术方案的一个优选，授权模块包括，

[0039] 属性模块，用于枚举出用户的全部属性信息，获取用户的属性信息列表；

[0040] 参数模块，用于根据安全参数，结合属性信息获取用户的校验参数；

[0041] 请求模块，用于将分配获得的授权凭证、身份参数和校验参数发送到授权服务器；

[0042] 签名模块，用于对授权通过的身份参数和校验参数进行签名处理，并固化保存到区块链中。

[0043] 作为本发明技术方案的一个优选，登陆模块包括，

[0044] 条件模块，用于根据业务需求，公开登陆条件，所述登陆条件对用户的一个或多个属性信息提出要求；

[0045] 属性信息模块，用于请求登陆的用户根据登陆条件，选择对应的属性信息生成其所对应的哈希值和/或验证参数；

[0046] 登陆凭证模块，用于生成登陆凭证，所述登陆凭证包括符合登陆条件的属性信息、该属性信息的哈希值、验证参数以及用户的身份参数和校验参数。

[0047] 作为本发明技术方案的一个优选，验证模块包括，

[0048] 参数验证模块，用于接收用户登陆凭证，确定用户的身份参数和校验参数存储在区块链中；

[0049] 属性验证模块，用于对用户的属性信息进行哈希计算，确定其与验证参数中对应的属性哈希值相吻合；

[0050] 身份验证模块，用于根据用户的属性信息、校验参数以及验证参数，确定当前用户提供的登陆凭证与保存在区块链中的身份参数一致，即为验证通过。

[0051] 总体而言，通过本发明所构思的以上技术方案与现有技术相比，具有以下有益效果：

[0052] 1)本发明技术方案，与一般等登录方法相比，提供了一种匿名的登陆方法，第三方资源系统仅知道该用户是否有权登陆，而不知道该用户的具体信息。即使授权服务器和第三方系统串通，也无法获知用户的隐私信息。

[0053] 2)本发明技术方案，与一般等登录方法相比，用户数据采用分散式存储，用户的数据有用户自己存储，授权服务器无需保存任何内容，攻击者即使侵入了授权服务器依然无法对用户的隐私造成伤害。

[0054] 3)本发明技术方案，与一般的多次运用授权服务器登录方法相比，用户自己生成匿名登陆凭证，用户自行决定何时何地生成何用需求的匿名凭证，无需与授权服务器进行交互，大大优化了登陆流程。

附图说明

[0055] 图1是本发明技术方案实施例的信息结构概览图；

[0056] 图2是本发明技术方案实施例的Merkle Tree示例图；

[0057] 图3是本发明技术方案实施例的登陆验证流程图。

具体实施方式

[0058] 为了使本发明的目的、技术方案及优点更加清楚明白，以下结合附图及实施例，对本发明进行进一步详细说明。应当理解，此处所描述的具体实施例仅用以解释本发明，并不用于限定本发明。此外，下面所描述的本发明各个实施方式中所涉及到的技术特征只要彼此之间未构成冲突就可以相互组合。下面结合具体实施方式对本发明进一步详细说明。

[0059] 如图3所示，本发明技术方案的实施例中提供了一种基于区块链的匿名登陆方法，其主要特点在于，用户首先通过授权服务器进行验证授权，然后对个人的属性信息进行哈希运算，生成特定的哈希值，并将相应的信息保存到区块链上。用户在需要登陆系统(或者说是第三方系统)的时候，只需要提供部分属性信息和部分属性信息的哈希值，通过保存在区块链上的验证信息即可对用户的身份进行验证，从而实现用户的匿名登陆。

[0060] 需要特别说明的是，本实施例中的匿名登陆，并不是单纯的隐去姓名，而是对用户的身份进行部分覆盖加密，在进行登陆的时候只需要提供部分用户属性信息，使得通过用户提供的部分身份信息即可确定其是否满足登陆条件。从而根据用户提供的部分身份信息只能确定一定的人群范围，并不能实际确定具体的用户身份。也就是说，通过将请求登陆的用户隐藏在满足一定条件范围的人群里，以达到匿名的目的。

[0061] 具体来说，如图1所示，本实施例中，首先要根据安全参数和非对称加密算法等，对授权服务器的公钥和私钥进行初始化，将该授权服务器的公钥公开，并根据该公钥获取对应的区块链地址ADR。经过初始化的授权服务器随机生成若干个随机数，并随机分配给用户。本实施例中，优选将随机数表达为{$Rand_0$,$Rand_1$,.....,$Rand_{m-1}$,$Rand_m$}。

[0062] 进一步地，用户枚举出自身的所有属性信息，如姓名、出生日期、性别、国籍.....，从而获得用户所有属性的集合{$attr_0$,$attr_1$,.....,$attr_{n-1}$,$attr_n$}，根据用户的属性集合计算获得Merkle Rootφ_i。本实施例中的Merkle Root如图2所示。

[0063] 对任意的用户来说，本实施例中的Merkle Rootφ_i(用户的身份参数)优选具有如下计算公式：

[0064] $\varphi_i = \mathrm{Merkle}(\mathrm{attr}_0 + \mathrm{attr}_1 + \cdots\ldots, \mathrm{attr}_{n-1} + \mathrm{attr}_n\})$。

[0065] 具体来说就是，通过对该用户的全部属性信息进行哈希运算，最后获得一个代表用户身份的Merkle Root(哈希值)。其原理在于，对相邻的属性连续进行哈希运算，并将得到的结果进行迭代哈希运算。因为任何字符串进行连续哈希运算，获取相同的哈希值的(哈希碰撞)概率极小，从而属性的哈希值可以作为这个用户的身份校验标识。

[0066] 以图2为例来对上述方案进行说明，假设根据用户的姓名、出生日期、性别和国籍这几项属性信息即可确认用户的身份，那么将这些属性信息分为两组，分别进行迭代哈希运算，直至获得最后的Merkle Root。如图2中即为，将代表姓名的$attr_0$和代表出生日期的$attr_1$进行哈希运算，获得Hash($attr_0$,$attr_1$)，类似的，获得性别属性信息和国籍属性信息的Hash($attr_2$,$attr_3$)，然后再次对Hash($attr_0$,$attr_1$)和Hash($attr_2$,$attr_3$)进行哈希运算，

所获得的哈希值即为图2中的Merkle Root。

[0067] 由于对于任何字符串进行哈希运算,获取相同的哈希值的(哈希碰撞)概率极小,从而这个Merkle Root可以作为用户的身份识别凭证。优选的,本实施例中对每个属性信息都进行了哈希运算,即对于用户i来说,其第j个属性信息的哈希值具有如下表达:

[0068] $hash_{ij}=Hash(attr_{ij})$。

[0069] 进一步地,在图2中Merkle Tree的基础上,用户按照要求选择参数生成属于自己的校验参数,本实施例中的校验参数(commitment ω_i)具有如下表达:

[0070] $$\omega_i = h^r g_0^{aux} \prod_{0}^{j} g_{j+1}^{hash_{ij}}$$

[0071] 其中G为Z_p上的q阶子群,选择随机生成器$G=<g_0>=...<g_j>$,h为g_0^a,r为随机数,aux为任意数,将来用来替换登陆验证码。参数a为隐私参数,仅为用户所拥有,其他参数为公开参数。

[0072] 然后,用户将Merkle Root、分配获得的随机数以及校验参数一起发送到授权服务器。授权服务器首先读取用户提供的随机数,判断该随机数是否在授权服务器分发给用户的随机数列表中,如果在,则对用户提供的数据进行进一步的处理后发送到区块链中,否则拒绝该用户的授权请求。

[0073] 对于随机数符合判断要求的用户,授权服务器首先对其进行签名处理,然后再进行发送,其具体过程如下:

[0074] $$userdata = \varphi i + Rand_i + \omega_i$$

[0075] $$signature = SIGN_{sk_s}(userdata)$$

[0076] transaction=GenTran(version,input,output,data:userdata+signature)

[0077] 上述数据参数经过一定数据的区块链节点确认后被永久固化到区块链中,任何人都无法对该用户的上述身份认证信息进行修改,即用户完成了自身属性信息的绑定过程。作为本实施例的一个优选,为了增加不可追踪性和不可关联性,可以选择多次执行上述步骤,即多次对根据用户属性信息获得的身份参数和校验参数进行签名处理,并将其固化保存到区块链中。换而言之,本实施例中的技术方案允许一个用户使用多套哈希值(如对相同的属性信息采用不同的哈希算法以获得不同的哈希值等)进行身份验证,其中每个哈希值可以互不相同,但是都是该用户的准确真实身份参数。

[0078] 同时需要强调的是,在以上方案实施的过程中,用户连续使用多次相同的匿名凭证可能会导致该用户匿名机制丧失,更具体的说,如果同一个用户使用同一类匿名凭证(如只显示年龄)对同一个系统联系登陆多次,该系统可能会反推出,此同一类匿名凭证属于同一个用户,为了更好的实现登陆系统的反追踪性和反关联性,本实施例中允许一个用户在不同的授权服务器上进行认证,并在不同的登陆请求下使用有不同的授权服务器背书的匿名凭证。

[0079] 在此技术上,用户可以开始对第三方系统进行匿名访问,其具体过程如下:

[0080] 首先,第三方系统发布实际的业务需求,标明系统的登陆条件,如仅允许某个群体的人员登陆,又或者不允许一定年龄段的人员登陆等,其可以根据需求自由设定。也就是说,其可以为用户登陆设置一定的阈值条件,可以是用户的单个属性信息(如年龄、性别或

者国籍)，也可以是多个属性信息的组合(如年龄+性别)，本发明技术方案中不对此作具体的限定，本实施例中的具体属性类别也仅作说明本发明技术方案之用，不视为对本发明技术方案的具体限制。

[0081] 然后，用户根据第三方系统的实际需求，选择所需要的属性信息生成登陆凭证，然后将该登陆凭证提供给第三方系统。本实施例中，属性信息登陆凭证优选如下：

[0082] $\{attr_{ij}, \mathrm{hash}_{ij}, \mathrm{authcode}, (\tau_{ij}, \varphi_i), (\omega_i, \Gamma_i)\}$。

[0083] 其中，authcode为登陆验证码，需要说明的是，hash_{ij}和τ_{ij}为属性attr_j的验证参数，此处的τ_{ij}计算为标准的Merkle proof的算法，τ_{ij}计算公式优选如下，

[0084] $\tau_{ij}=\{\mathrm{attr}_j, \mathrm{Proof}(\varphi_i, \mathrm{attr}_j)\}$。

[0085] 其中，Γ_i的计算采用了佩德森承诺算法，因为仅有用户知道隐私的参数a和离散对数的分解难题，因此仅有用户可以迅速计算出Γ_i，计算公式优选如下：

[0086] $h^r g_0^{aux} = h^r g_0^{authcode}$

[0087] $g_0^{ar} g_0^{aux} = g_0^{ar'} g_0^{authcode}$

[0088] $\Gamma_i=\{\mathrm{authcode}, \frac{(\mathrm{aux}+\mathrm{ar})-\mathrm{authcode}}{a}\}$

[0089] 换句话说，上述登陆凭证中包含有如下信息：属性信息、该属性信息对应的哈希值、该属性信息的验证参数、用户的校验参数以及用户的身份参数，第三方根据上述信息对用户的身份进行验证。

[0090] 本实施例中，第三方系统的验证过程优选如下：

[0091] 第三方资源系统首先根据登陆凭证中的信息扫描区块链，获取φ_i和ω_i，比较其是否与登陆凭证中的一致，不一致的则直接拒绝登陆请求，一致则进入下一步骤。然后对hash_{ij}进行验证，即验证$\mathrm{hash}_{ij}=\mathrm{Hash}(\mathrm{attr}_{ij})$是否成立，若成立则继续对$\tau_{ij}$和$\varphi_i$进行验证，进一步的对$(\omega_i, \Gamma_i)$进行验证，任意一个不满足则拒绝用户登陆，只有登陆凭证中的上述参数信息均验证通过，才允许当前请求的用户进行匿名登陆。

[0092] 本领域的技术人员容易理解，以上所述仅为本发明的较佳实施例而已，并不用以限制本发明，凡在本发明的精神和原则之内所作的任何修改、等同替换和改进等，均应包含在本发明的保护范围之内。

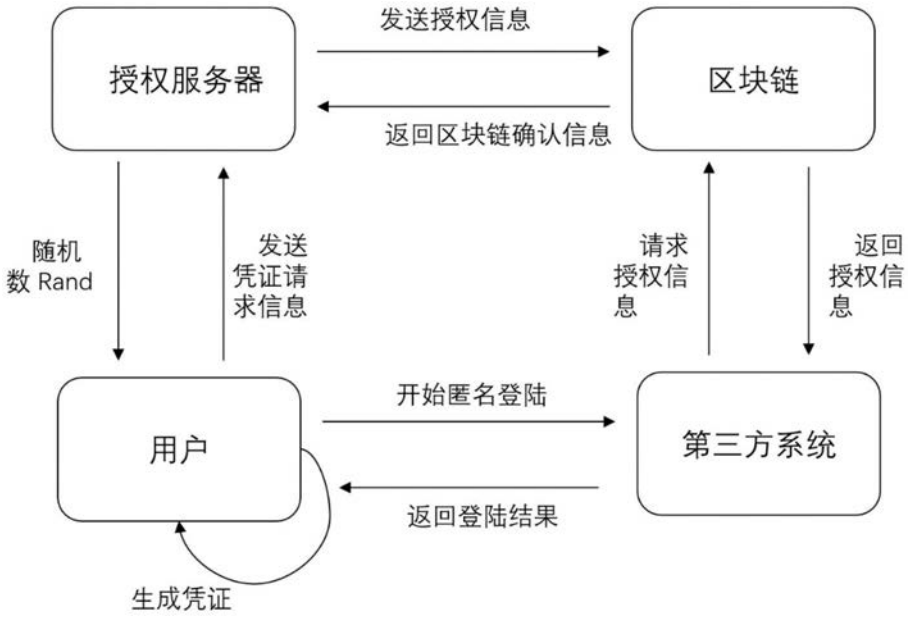

授权服务器
发送授权信息
区块链
返回区块链确认信息
随机数 Rand
发送凭证请求信息
请求授权信息
返回授权信息
用户
开始匿名登陆
第三方系统
返回登陆结果
生成凭证

图1

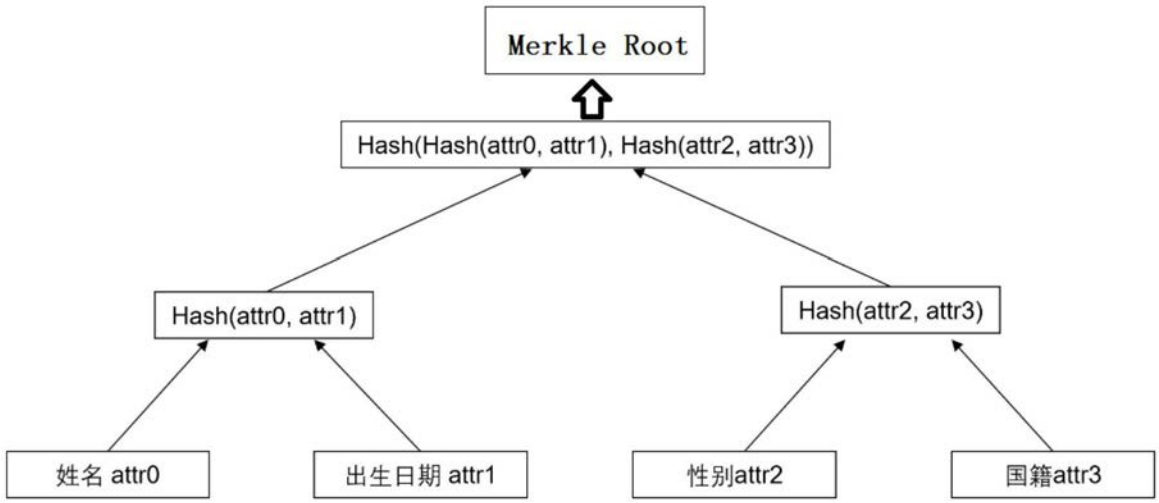

Merkle Root
Hash(Hash(attr0, attr1), Hash(attr2, attr3))
Hash(attr0, attr1)
Hash(attr2, attr3)
姓名 attr0
出生日期 attr1
性别attr2
国籍attr3

图2

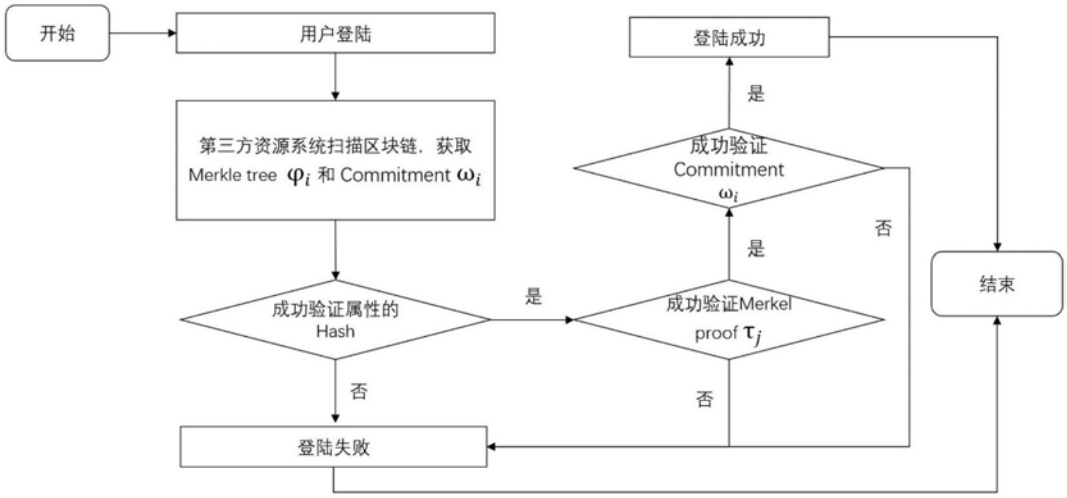

开始
用户登陆
第三方资源系统扫描区块链，获取 Merkle tree φ_i 和 Commitment ω_i
成功验证属性的 Hash
是
否
成功验证Merkel proof τ_j
是
否
成功验证 Commitment ω_i
是
否
登陆成功
结束
登陆失败

图3