(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 110851851 B
(45)授权公告日 2020.11.06

(21)申请号 202010039770.2

(22)申请日 2020.01.15

(65)同一申请的已公布的文献号
申请公布号 CN 110851851 A

(43)申请公布日 2020.02.28

(73)专利权人 蚂蚁区块链科技(上海)有限公司
地址 200025 上海市黄浦区黄陂南路838弄1号4幢A座20层(实际楼层17层)02单元

(72)发明人 朱明 李亿泽 张渊 管亚阳 杨新颖 闫文远 俞本权 陈思峰 诸威 余廷钊

(74)专利代理机构 北京博思佳知识产权代理有限公司 11415
代理人 陈冲

(51)Int.Cl.
G06F 21/60(2013.01)
G06F 21/62(2013.01)

审查员 崔成东

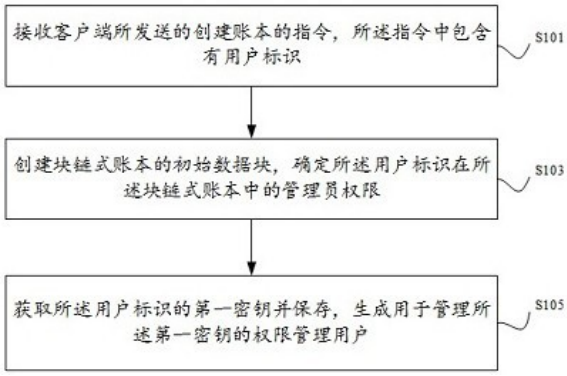
权利要求书2页 说明书8页 附图2页

(54)发明名称

一种块链式账本中的权限管理方法、装置及设备

(57)摘要

公开了一种块链式账本中的权限管理方法、装置及设备。通过本说明书实施例所提供的方案,在用户创建账本时,除根据用户的指令生成一个与用户标识所对应的管理员用户,还会同时生成一个用于管理用户密钥的权限管理用户,如果用户的第一密钥丢失,则即可以基于该权限管理用户重新建立新的第二密钥,并创建新的具有一定权限的角色基于第二密钥在账本中对数据记录进行数字签名。



CN 110851851 B

1.一种块链式账本中的权限管理方法,应用于通过块链式账本存储数据的中心化的数据库服务端中,包括:

接收客户端所发送的创建账本的指令,所述指令中包含有用户标识;

创建块链式账本的初始数据块,确定所述用户标识在所述块链式账本中的管理员权限;所述用户标识对应的用户为账本管理用户;

在所述数据库服务端的可信执行环境中创建并获取关联所述用户标识的第一密钥并保存,生成用于管理所述第一密钥的权限管理用户,其中,所述第一密钥用于所述账本管理用户在所述块链式账本中进行数字签名,所述第一密钥包括第一私钥和/或第一公钥,所述权限管理用户仅对于用户权限方面具有管理功能,所述权限管理用户与所述账本管理用户不是同一用户;

当所述账本管理用户的第一密钥丢失之后,接收客户端所发送的在所述块链式账本中请求授权的指令,调用所述权限管理用户,在所述可信执行环境中重新创建关联所述用户标识的第二密钥,所述第二密钥用于所述账本管理用户在所述账本中进行数字签名;

调用所述权限管理用户,在所述可信执行环境中对所述块链式账本中包含的历史数据记录进行验证,其中,所述历史数据为需要使用已经丢失的第一密钥进行解密的历史数据。

2.如权利要求1所述的方法,在所述块链式账本中,数据块通过如下方式生成:

接收用户所发送的待存储的数据记录,确定所述数据记录的哈希值;

当达到预设的成块条件时,确定待写入数据块中的各数据记录,生成包含数据块的哈希值和数据记录的第N个数据块:

当N=1时,初始数据块的哈希值和块高基于预设方式给定;

当N>1时,根据待写入数据块中的各数据记录和第N-1个数据块的哈希值确定第N个数据块的哈希值,生成包含第N个数据块的哈希值和各数据记录的第N个数据块,其中,数据块的块高基于成块时间的先后顺序单调递增。

3.如权利要求2所述的方法,所述预设的成块条件包括:

待存储的数据记录数量达到数量阈值;或者,

距离上一次成块时刻的时间间隔达到时间阈值。

4.一种块链式账本中的权限管理装置,应用于通过块链式账本存储数据的中心化的数据库服务端中,包括:

接收模块,接收客户端所发送的创建账本的指令,所述指令中包含有用户标识;

确定模块,创建块链式账本的初始数据块,确定所述用户标识在所述块链式账本中的管理员权限;所述用户标识对应的用户为账本管理用户;

生成模块,在所述数据库服务端的可信执行环境中创建并获取关联所述用户标识的第一密钥并保存,生成用于管理所述第一密钥的权限管理用户,其中,所述第一密钥用于所述账本管理用户在所述块链式账本中进行数字签名,所述第一密钥包括第一私钥和/或第一公钥,所述权限管理用户仅对于用户权限方面具有管理功能,所述权限管理用户与所述账本管理用户不是同一用户;

创建模块,当所述账本管理用户的第一密钥丢失之后,接收客户端所发送的在所述块链式账本中请求授权的指令,调用所述权限管理用户,在所述可信执行环境中重新创建关联所述用户标识的第二密钥,所述第二密钥用于所述账本管理用户在所述账本中进行数字

签名；

验证模块，调用所述权限管理用户，在所述可信执行环境中对所述块链式账本中包含的历史数据记录进行验证，其中，所述历史数据为需要使用已经丢失的第一密钥进行解密的历史数据。

5.如权利要求4所述的装置，还包括数据块生成模块：接收用户所发送的待存储的数据记录，确定所述数据记录的哈希值；当达到预设的成块条件时，确定待写入数据块中的各数据记录，生成包含数据块的哈希值和数据记录的第N个数据块：

当N＝1时，初始数据块的哈希值和块高基于预设方式给定；

当N>1时，根据待写入数据块中的各数据记录和第N－1个数据块的哈希值确定第N个数据块的哈希值，生成包含第N个数据块的哈希值和各数据记录的第N个数据块，其中，数据块的块高基于成块时间的先后顺序单调递增。

6.如权利要求5所述的装置，所述预设的成块条件包括：待存储的数据记录数量达到数量阈值；或者，距离上一次成块时刻的时间间隔达到时间阈值。

7.一种计算机设备，包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机程序，其中，所述处理器执行所述程序时实现如权利要求1至3任一项所述的方法。

一种块链式账本中的权限管理方法、装置及设备

技术领域

[0001] 本说明书实施例涉及信息技术领域，尤其涉及一种块链式账本中的权限管理方法、装置及设备。

背景技术

[0002] 在中心化的数据库服务端以块链式账本对外提供服务时，用户通常需要对自己所产生的数据记录进行数字签名，具体而言即为公钥加密，私钥解密；或者私钥加密，公钥解密，具体的表现即为用户需要一定级别的授权才可以使用相关的密钥。而如果在块链式账本中用户的密钥丢失，则会对于账本的使用产生很大的影响。

[0003] 基于此，需要一种可以在块链式账本中的用户权限进行有效管理的方案。

发明内容

[0004] 本申请实施例的目的是提供一种可以在块链式账本中的用户权限进行有效管理的方案。

[0005] 为解决上述技术问题，本申请实施例是这样实现的：

[0006] 一种块链式账本中的权限管理方法，应用于通过块链式账本存储数据的中心化的数据库服务端中，包括：

[0007] 接收客户端所发送的创建账本的指令，所述指令中包含有用户标识；

[0008] 创建块链式账本的初始数据块，确定所述用户标识在所述块链式账本中的管理员权限；

[0009] 获取所述用户标识的第一密钥并保存，生成用于管理所述第一密钥的权限管理用户，其中，所述第一密钥用于所述用户标识在所述块链式账本中进行数字签名，所述第一密钥包括第一私钥和/或第一公钥。

[0010] 对应的，本说明书实施例还提供一种块链式账本中的权限管理装置，应用于通过块链式账本存储数据的中心化的数据库服务端中，包括：

[0011] 接收模块，接收客户端所发送的创建账本的指令，所述指令中包含有用户标识；

[0012] 确定模块，创建块链式账本的初始数据块，确定所述用户标识在所述块链式账本中的管理员权限；

[0013] 生成模块，获取所述用户标识的第一密钥并保存，生成用于管理所述第一密钥的权限管理用户，其中，所述第一密钥用于所述用户标识在所述块链式账本中进行数字签名，所述第一密钥包括第一私钥和/或第一公钥。

[0014] 通过本说明书实施例所提供的方案，在用户创建账本时，除根据用户的指令生成一个与用户标识所对应的管理员用户，还会同时生成一个用于管理用户密钥的权限管理用户，如果用户的第一密钥丢失，则即可以基于该权限管理用户重新建立新的第二密钥，并创建新的具有一定权限的角色基于第二密钥在账本中对数据记录进行数字签名，从而实现对于用户权限的有效管理。

[0015] 应当理解的是,以上的一般描述和后文的细节描述仅是示例性和解释性的,并不能限制本说明书实施例。

[0016] 此外,本说明书实施例中的任一实施例并不需要达到上述的全部效果。

附图说明

[0017] 为了更清楚地说明本说明书实施例或现有技术中的技术方案,下面将对实施例或现有技术描述中所需要使用的附图作简单地介绍,显而易见地,下面描述中的附图仅仅是本说明书实施例中记载的一些实施例,对于本领域普通技术人员来讲,还可以根据这些附图获得其他的附图。

[0018] 图1是本说明书实施例提供的一种块链式账本中的权限管理方法的流程示意图;

[0019] 图2为本说明书实施例所提供的一种数据块的块头的示意图;

[0020] 图3为本说明书实施例所提供的一种链式账本中的数据块生成方法的流程示意图;

[0021] 图4是本说明书实施例提供的一种块链式账本中的权限管理装置的结构示意图;

[0022] 图5是用于配置本说明书实施例方法的一种设备的结构示意图。

具体实施方式

[0023] 为了使本领域技术人员更好地理解本说明书实施例中的技术方案,下面将结合本说明书实施例中的附图,对本说明书实施例中的技术方案进行详细地描述,显然,所描述的实施例仅仅是本说明书的一部分实施例,而不是全部的实施例。基于本说明书中的实施例,本领域普通技术人员所获得的所有其他实施例,都应当属于保护的范围。

[0024] 以下结合附图,详细说明本说明书各实施例提供的技术方案。如图1所示,图1是本说明书实施例提供的一种块链式账本中的权限管理方法的流程示意图,应用于通过块链式账本存储数据的中心化的数据库服务端中,该流程具体包括如下步骤:

[0025] S101,接收客户端所发送的创建账本的指令,所述指令中包含有用户标识。

[0026] 在数据库服务端中,用户可以发送创建账本的指令。例如,NEW(LGNAME,Admin,UserID),其中LGNAME为账本名称,“Admin”表征用户指定管理员权限,“UserID”即为用户指定为账本中具有管理员权限的用户标识,包括身份证号、手机号码或者客户端唯一标识等等。需要说明的是,在指令中,“UserID”可以是包含多份用户标识的集合,即可以在一个账本中同时指定多个具有管理员权限的用户。

[0027] 同时需要说明的是,在账本被创建之后,在一个用户标识下,可以对应于多个角色Role,而多个角色可以有不同等级的权限。

[0028] S103,创建块链式账本的初始数据块,确定所述用户标识在所述块链式账本中的管理员权限。

[0029] 数据库服务端在接收到创建账本的指令后。即创建一份名为“LGNAME”的账本。在本说明书实施例中,由于账本是块链式(即多个数据块依序链式连接)的,因此,实际上对于一份新创建的账本而言,此时仅需创建一个初始数据块即可。后续的数据块将会在达到一定成块条件后再依序生成。

[0030] 块链式账本中的数据块,可以包括块头和块体两个部分。块体中可以用于存储拼

接数据的明文，或者拼接数据的哈希值等等；块头中可以用于存储有关本数据块的元数据，例如，账本的版本号，前一数据块的哈希值，自身数据块中的拼接数据所组成的默克尔树的根哈希值，自身数据块的哈希值，用于记录拼接数据的被操作状态的状态数组等等。如图2所示，图2为本说明书实施例所提供的一种数据块的块头的示意图。

[0031] 在初始数据块被创建之后，用户后续的数据记录即可以按照如下方式生成，从而生成相应的块链式账本，如图3所示，图3为本说明书实施例所提供的一种链式账本中的数据块生成方法的流程示意图，该流程具体包括如下步骤：

[0032] S301，接收待存储的数据记录，确定各数据记录的哈希值。此处的待存储的数据记录，可以是客户端个人用户的各种消费记录，也可以是应用服务器基于用户的指令，在执行业务逻辑时产生的业务结果、中间状态以及操作记录等等。具体的业务场景可以包括消费记录、审计日志、供应链条、政府监管记录、医疗记录等等。S303，当达到预设的成块条件时，确定待写入数据块中的各数据记录，生成包含数据块的哈希值和数据记录的第N个数据块。

[0033] 所述预设的成块条件包括：待存储的数据记录数量达到数量阈值，例如，每接收到一千条数据记录时，生成一个新数据块，将一千条数据记录写入块中；或者，距离上一次成块时刻的时间间隔达到时间阈值，例如，每隔5分钟，生成一个新数据块，将在这5分钟内接收到的数据记录写入块中。

[0034] 此处的N指的是数据块的序号，换言之，在本说明书实施例中，数据块是以块链的形式，基于成块时间的顺序先后排列，具有很强的时序特征。其中，数据块的块高基于成块时间的先后顺序单调递增。块高可以是序号，此时第N个数据块的块高即为N；块高也可以其它方式生成，例如，将数据块的成块时间戳转换为单调递增的大整型数据，以该大整型数据作为数据块的块高。

[0035] 当N=1时，即此时的数据块为为初始数据块。初始数据块的哈希值和块高基于预设方式给定。例如，初始数据块中不包含数据记录，哈希值则为任一给定的哈希值，块高blknum=0；又例如，初始数据块的生成触发条件与其它数据块的触发条件一致，但是初始数据块的哈希值由对初始数据块中的所有内容取哈希确定。

[0036] 当N>1时，由于前一数据块的内容和哈希值已经确定，则此时，可以基于前一数据块(即第N－1个数据块)的哈希值生成当前数据块(第N个数据块)的哈希值。

[0037] 具体而言，可以确定每一条将要写入第N个块中的数据记录的哈希值，按照在块中的排列顺序，生成一个默克尔树，将默克尔树的根哈希值和前一数据块的哈希值拼接在一起，再次采用哈希算法，生成当前块的哈希值，以及还可以根据默克尔树的根哈希值和其它一些元数据(例如版本号、数据块的生成时间戳等等)生成当前块的哈希值。并且，将所述数据记录写入数据块的块体中，将所述根哈希写入数据块的块头中，其中，数据块的块高基于成块时间的先后顺序单调递增.

[0038] 通过前述的数据块的生成方式，每一个数据块通过哈希值确定，数据块的哈希值由数据块中的数据记录的内容、顺序以及前一数据块的哈希值决定。用户可以随时基于数据块的哈希值或者数据记录的哈希值发起验证，对于数据块中任何内容(包括对于数据块中数据记录内容或者顺序的修改)的修改都会造成在验证时计算得到的数据块的哈希值和数据块生成时的哈希值不一致，而导致验证失败，从而实现了中心化下的不可篡改。

[0039] 用户在上传数据成功后，即可以得到对应的数据记录的哈希值以及所处的数据块

的哈希值,并保存,并且可以基于该哈希值发起完整性验证。

[0040] 完整性验证包括对于一个数据块的完整性验证,即,根据数据块中数据记录的哈希值重新组成默克尔树,计算默克尔树的根哈希值,并且根据默克尔树的根哈希值与前一数据块的哈希值重新计算该数据块的哈希值,与事先保存的数据块的哈希值进行一致性对比。

[0041] 完整性验证还可以包括对于若干连续数据块的完整性验证,即根据数据块的块头中所保存的默克尔树的根哈希值与前一数据块的哈希值重新计算该数据块的哈希值,并与事先保存的数据块的哈希值进行对比。

[0042] S105,获取所述用户标识的第一密钥并保存,生成用于管理所述第一密钥的权限管理用户。

[0043] 如前所述,在该账本中,指令中所包含的用户标识将会被作为账本的创始人,被分配相应的管理员权限。具体而言,管理员权限至少拥有查询、验证、清除以及隐藏等权限。而一般用户则只有查询以及验证权限,没有清除以及隐藏权限。

[0044] 换言之,一个用户标识可能对应于多个不同权限的用户。其中有些用户可能可以使用密钥,而另一些而不能使用密钥(具有只读权限的用户)。

[0045] 而在中心化的块链式账本中,为了防止篡改,用户方所上传的数据记录以及任一对于数据记录所进行的操作,都会相应的由用户方进行数字签名。例如,在创建管理员用户时即确定第一密钥(包括第一私钥和/或第一公钥)用于在该账本中进行数字签名。

[0046] 例如,用户首先将第一公钥上传至服务端并保存,然后用户上传的数据均为经过第一私钥签名的加密数据,服务端可以对于用户所上传加密数据采用第一公钥进行解密。又例如,服务端返回数据时,首先采用用户的公钥加密生成加密数据,用户在接收到加密数据之后采用私钥解密,从而得到解密数据,防止数据泄露等等。

[0047] 在本说明书实施例中,服务端获取用户第一密钥的方式有如下几种:

[0048] 第一种,由用户在获取管理员权限之后上传自身的第一公钥,但是在第一私钥保存在用户本地。

[0049] 第二种,在创建管理员用户之后,即在可信执行环境(Trusted Execution Environment, TEE)中生成对应的第一公钥和私钥对。TEE可以起到硬件中的黑箱作用,在TEE中执行的代码和数据操作系统层都无法偷窥,只有代码中预先定义的接口才能对其进行操作。可信执行环境是基于 CPU 硬件的安全扩展,且与外部完全隔离的可信执行环境。TEE最早是由Global Platform提出的概念,用于解决移动设备上资源的安全隔离,平行于操作系统为应用程序提供可信安全的执行环境。ARM的Trust Zone技术最早实现了真正商用的TEE技术。

[0050] 伴随着互联网的高速发展,安全的需求越来越高,不仅限于移动设备,云端设备,数据中心都对TEE提出了更多的需求。TEE的概念也得到了高速的发展和扩充。现在所说的TEE相比与最初提出的概念已经是更加广义的TEE。例如,服务器芯片厂商Intel,AMD等都先后推出了硬件辅助的TEE并丰富了TEE的概念和特性,在工业界得到了广泛的认可。现在提起的TEE通常更多指这类硬件辅助的TEE技术。不同于移动端,云端访问需要远程访问,终端用户对硬件平台不可见,因此使用TEE的第一步就是要确认TEE的真实可信。因此现在的TEE技术都引入了远程证明机制,由硬件厂商(主要是CPU厂商) 背书并通过数字签名技术确保

用户对TEE状态可验证。换言之,在TEE中执行的结果可以得到硬件厂商的数字签名。

[0051] 同时仅仅是安全的资源隔离也无法满足的安全需求,进一步的数据隐私保护也被提出。包括Intel SGX, AMD SEV在内的商用TEE也都提供了内存加密技术,将可信硬件限定在CPU内部,总线和内存的数据均是密文防止恶意用户进行窥探。例如,英特尔的软件保护扩展(SGX)等 TEE 技术隔离了代码执行、远程证明、安全配置、数据的安全存储以及用于执行代码的可信路径。在 TEE 中运行的应用程序受到安全保护,几乎不可能被第三方访问。

[0052] 以Intel SGX技术为例,SGX提供了围圈(enclave,也称为飞地),即内存中一个加密的可信执行区域,由 CPU 保护数据不被窃取。以服务端采用支持SGX的CPU为例,利用新增的处理器指令,在内存中可以分配一部分区域 EPC(Enclave Page Cache,围圈页面缓存或飞地页面缓存),通过 CPU 内的加密引擎 MEE(Memory Encryption Engine)对其中的数据进行加密。EPC 中加密的内容只有进入 CPU 后才会被解密成明文。因此,在 SGX 中,用户可以不信任操作系统、VMM(Virtual Machine Monitor,虚拟机监控器)、甚至 BIOS(Basic Input Output System,基本输入输出系统),只需要信任 CPU 便能确保代码的执行。

[0053] 在本说明书实施例中,可以通过在可信执行环境中执行预设的密钥生成算法,并将第一私钥在TEE中进行保存,并且只需对外公开第一公钥即可。第一密钥的真实可靠性由可信执行环境的硬件提供方所保障。

[0054] 在确定了第一密钥之后,服务端即可以生成用于管理所述第一密钥的权限管理用户,该权限管理用户主要用于是服务端提供给用户的一层密钥权限保障。

[0055] 具体而言,用户对于该权限管理用户是无感知的,该权限管理用户仅仅具有对于用户权限方面具有管理功能,而没有其它权限,例如,不能对于账本中的数据记录进行任何操作。

[0056] 该权限管理用户对于所述第一密钥具有管理功能的具体表现可以包括,该权限管理用户可以进行诸如GRANT或者creat的相应功能。例如,服务端可以通过该权限管理用户输入指令,GRANT(userid, &v) : 给与userid所对应的用户权重值v;即分配给某个用户一定的权限值,从而创建一个具有一定权限的新用户,而该新用户如果权重超过一定值,则可以赋予该新用户对于密钥的使用权限,以便该用户可以基于密钥进行数字签名。

[0057] 又例如,如果用户的密钥丢失,那么用户则可以通客户端向服务端发起在所述块链式账本中请求重新授权可用密钥的指令,服务端可以通过调用该权限管理用户输入指令creat(userid ,pubkey),在TEE环境中给userid创建一个可供用户使用的第二密钥对等等,从而用户可以基于第二密钥在账本中进行数字签名。

[0058] 综上所述,生成的权限管理用户可以直接或者间接的对于用户的第一私钥和/或第一公钥进行相应的创建或者授权,并且创建新的第二密钥对和用户授权,从而实现对应的权限管理。

[0059] 进一步地,基于用户的第一密钥(特别是私钥)已经丢失,而由于用户账本中的历史数据记录都是基于第一密钥进行加密的,如果需要对于用户的历史数据记录进行验证,则会发生在用户方没有私钥而无法解密的情形。

[0060] 基于此,由于第一密钥可以是在TEE环境中生成,因此,在TEE生成第一密钥时,可以同时关联用户标识的管理员用户和所创建的权限管理用户,从而即使用户标识所对应的

管理员用户的第一私钥发生了丢失，但是在TEE中第一私钥仍然和权限管理用户存在关联关系，此时如果需要对于用户那些历史数据进行验证，则可以调用权限管理用户在TEE中进行解密验证。

[0061] 在验证完成之后，服务端即对于验证结果进行数字签名，从而可以证明即使这些历史数据记录的验证过程中，并不能再被用户所解密，但是，这些历史数据记录的验证结果仍然是得到了服务端的签名背书，保障了用户的历史数据记录的真实和可用性。

[0062] 通过本说明书实施例所提供的方案，在用户创建账本时，除根据用户的指令生成一个与用户标识所对应的管理员用户，还会同时生成一个用于管理用户密钥的权限管理用户，如果用户的第一密钥丢失，则即可以基于该权限管理用户重新建立新的第二密钥，并创建新的具有一定权限的角色基于第二密钥在账本中对数据记录进行数字签名，从而实现对于用户权限的有效管理。

[0063] 对应的，本说明书实施例还提供一种块链式账本中的权限管理装置，应用于通过块链式账本存储数据的中心化的数据库服务端中，如图4所示，图4是本说明书实施例提供的一种块链式账本中的权限管理装置的结构示意图，包括：

[0064] 接收模块401，接收客户端所发送的创建账本的指令，所述指令中包含有用户标识；

[0065] 确定模块403，创建块链式账本的初始数据块，确定所述用户标识在所述块链式账本中的管理员权限；

[0066] 生成模块405，获取所述用户标识的第一密钥并保存，生成用于管理所述第一密钥的权限管理用户，其中，所述第一密钥用于所述用户标识在所述块链式账本中进行数字签名，所述第一密钥包括第一私钥和/或第一公钥。

[0067] 进一步地，所述生成模块405，在所述服务端的可信执行环境中创建并获取关联所述用户标识的第一密钥；或者，接收客户端所发送的第一公钥。

[0068] 进一步地，所述装置还包括创建模块407，接收客户端所发送的在所述块链式账本中请求授权的指令；调用所述权限管理用户，在所述块链式账本中重新创建关联所述用户标识的第二密钥。

[0069] 进一步地，所述装置还包括验证模块409，调用所述权限管理用户，对所述块链式账本中包含第一私钥签名的数据记录进行验证。

[0070] 进一步地，还包括数据块生成模块411，接收用户所发送的待存储的数据记录，确定所述数据记录的哈希值；当达到预设的成块条件时，确定待写入数据块中的各数据记录，生成包含数据块的哈希值和数据记录的第N个数据块：

[0071] 当N=1时，初始数据块的哈希值和块高基于预设方式给定；

[0072] 当N>1时，根据待写入数据块中的各数据记录和第N－1个数据块的哈希值确定第N个数据块的哈希值，生成包含第N个数据块的哈希值和各数据记录的第N个数据块，其中，数据块的块高基于成块时间的先后顺序单调递增。

[0073] 进一步地，在所述装置中所述预设的成块条件包括：待存储的数据记录数量达到数量阈值；或者，距离上一次成块时刻的时间间隔达到时间阈值。

[0074] 本说明书实施例还提供一种计算机设备，其至少包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机程序，其中，处理器执行所述程序时实现图1所示的权

限管理方法。

[0075] 图5示出了本说明书实施例所提供的一种更为具体的计算设备硬件结构示意图，该设备可以包括：处理器1010、存储器1020、输入/输出接口1030、通信接口1040和总线1050。其中处理器1010、存储器1020、输入/输出接口1030和通信接口1040通过总线1050实现彼此之间在设备内部的通信连接。

[0076] 处理器1010可以采用通用的CPU(Central Processing Unit，中央处理器)、微处理器、应用专用集成电路(Application Specific Integrated Circuit，ASIC)、或者一个或多个集成电路等方式实现，用于执行相关程序，以实现本说明书实施例所提供的技术方案。

[0077] 存储器1020可以采用ROM(Read Only Memory，只读存储器)、RAM(Random Access Memory，随机存取存储器)、静态存储设备，动态存储设备等形式实现。存储器1020可以存储操作系统和其他应用程序，在通过软件或者固件来实现本说明书实施例所提供的技术方案时，相关的程序代码保存在存储器1020中，并由处理器1010来调用执行。

[0078] 输入/输出接口1030用于连接输入/输出模块，以实现信息输入及输出。输入/输出模块可以作为组件配置在设备中(图中未示出)，也可以外接于设备以提供相应功能。其中输入设备可以包括键盘、鼠标、触摸屏、麦克风、各类传感器等，输出设备可以包括显示器、扬声器、振动器、指示灯等。

[0079] 通信接口1040用于连接通信模块(图中未示出)，以实现本设备与其他设备的通信交互。其中通信模块可以通过有线方式(例如USB、网线等)实现通信，也可以通过无线方式(例如移动网络、WIFI、蓝牙等)实现通信。

[0080] 总线1050包括一通路，在设备的各个组件(例如处理器1010、存储器1020、输入/输出接口1030和通信接口1040)之间传输信息。

[0081] 需要说明的是，尽管上述设备仅示出了处理器1010、存储器1020、输入/输出接口1030、通信接口1040以及总线1050，但是在具体实施过程中，该设备还可以包括实现正常运行所必需的其他组件。此外，本领域的技术人员可以理解的是，上述设备中也可以仅包含实现本说明书实施例方案所必需的组件，而不必包含图中所示的全部组件。

[0082] 本说明书实施例还提供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图1所示的权限管理方法。

[0083] 计算机可读介质包括永久性和非永久性、可移动和非可移动媒体可以由任何方法或技术来实现信息存储。信息可以是计算机可读指令、数据结构、程序的模块或其他数据。计算机的存储介质的例子包括，但不限于相变内存(PRAM)、静态随机存取存储器(SRAM)、动态随机存取存储器(DRAM)、其他类型的随机存取存储器(RAM)、只读存储器(ROM)、电可擦除可编程只读存储器(EEPROM)、快闪记忆体或其他内存技术、只读光盘只读存储器(CD-ROM)、数字多功能光盘(DVD)或其他光学存储、磁盒式磁带，磁带磁盘存储或其他磁性存储设备或任何其他非传输介质，可用于存储可以被计算设备访问的信息。按照本文中的界定，计算机可读介质不包括暂存电脑可读媒体(transitory media)，如调制的数据信号和载波。

[0084] 通过以上的实施方式的描述可知，本领域的技术人员可以清楚地了解到本说明书实施例可借助软件加必需的通用硬件平台的方式来实现。基于这样的理解，本说明书实施例的技术方案本质上或者说对现有技术做出贡献的部分可以以软件产品的形式体现出来，

该计算机软件产品可以存储在存储介质中，如ROM/RAM、磁碟、光盘等，包括若干指令用以使得一台计算机设备(可以是个人计算机，服务器，或者网络设备等)执行本说明书实施例各个实施例或者实施例的某些部分所述的方法。

[0085] 上述实施例阐明的系统、方法、模块或单元，具体可以由计算机芯片或实体实现，或者由具有某种功能的产品来实现。一种典型的实现设备为计算机，计算机的具体形式可以是个人计算机、膝上型计算机、蜂窝电话、相机电话、智能电话、个人数字助理、媒体播放器、导航设备、电子邮件收发设备、游戏控制台、平板计算机、可穿戴设备或者这些设备中的任意几种设备的组合。

[0086] 本说明书中的各个实施例均采用递进的方式描述，各个实施例之间相同相似的部分互相参见即可，每个实施例重点说明的都是与其他实施例的不同之处。尤其，对于方法实施例而言，由于其基本相似于方法实施例，所以描述得比较简单，相关之处参见方法实施例的部分说明即可。以上所描述的方法实施例仅仅是示意性的，其中所述作为分离部件说明的模块可以是或者也可以不是物理上分开的，在实施本说明书实施例方案时可以把各模块的功能在同一个或多个软件和/或硬件中实现。也可以根据实际的需要选择其中的部分或者全部模块来实现本实施例方案的目的。本领域普通技术人员在不付出创造性劳动的情况下，即可以理解并实施。

[0087] 以上所述仅是本说明书实施例的具体实施方式，应当指出，对于本技术领域的普通技术人员来说，在不脱离本说明书实施例原理的前提下，还可以做出若干改进和润饰，这些改进和润饰也应视为本说明书实施例的保护范围。

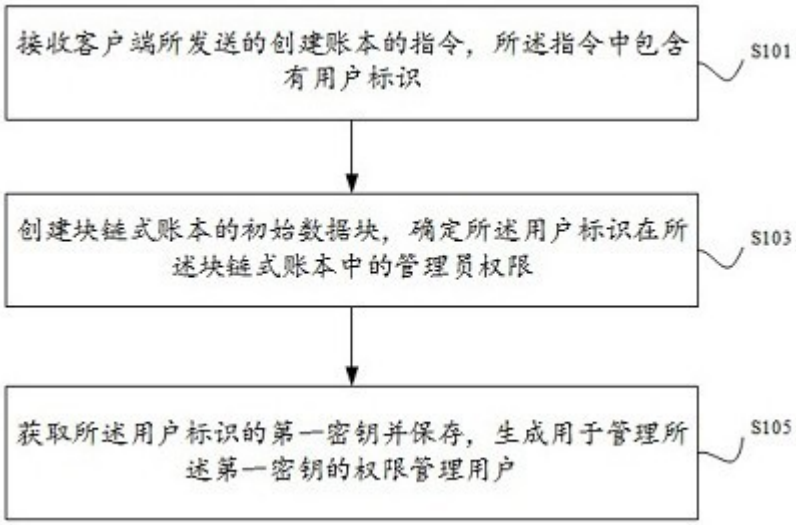


图1

字段名称	字段类型	字段说明
Hash	h256	数据块的哈希值
Version	Uint32_t	版本号
Number	Uint64_t	块高
Parent_hash	h256	父数据块的哈希值
Tx_root	h256	块体中的数据的默克尔树根哈希
Time_stamp	Uint64_t	时间戳

图2

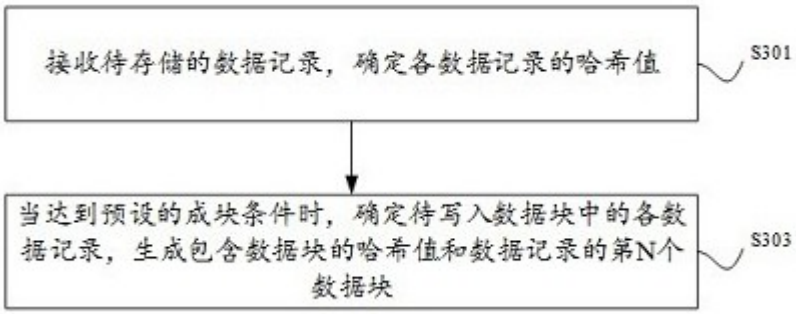


图3

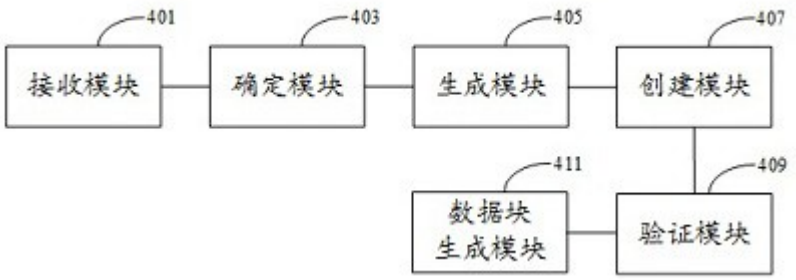

401
403
405
407
接收模块
确定模块
生成模块
创建模块
411
409
数据块
生成模块
验证模块

图4

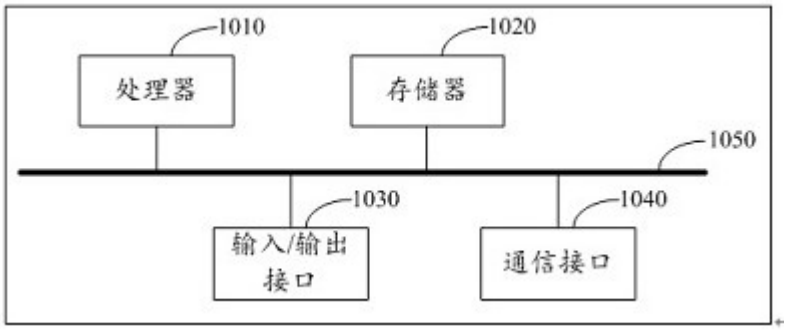

1010
1020
处理器
存储器
1050
1030
1040
输入/输出
接口
通信接口

图5