(19)国家知识产权局

(12)发明专利

(10)授权公告号 CN 112491843 B
(45)授权公告日 2022.06.21

(21)申请号 202011288784.4
(22)申请日 2020.11.17
(65)同一申请的已公布的文献号
申请公布号 CN 112491843 A
(43)申请公布日 2021.03.12
(73)专利权人 苏州浪潮智能科技有限公司
地址 215100 江苏省苏州市吴中区吴中经济开发区郭巷街道官浦路1号9幢
(72)发明人 吴标强
(74)专利代理机构 济南舜源专利事务所有限公司 37205
专利代理师 孙玉营
(51)Int.Cl.
H04L 9/40(2022.01)
H04L 9/32(2006.01)
H04L 9/08(2006.01)
审查员 王姣

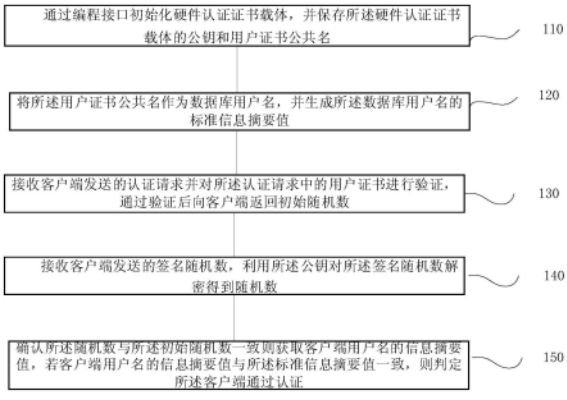
权利要求书2页 说明书8页 附图3页

(54)发明名称

一种数据库多重认证方法、系统、终端及存储介质

(57)摘要

本发明提供一种数据库多重认证方法、系统、终端及存储介质，包括：通过编程接口初始化硬件认证证书载体，并保存所述硬件认证证书载体的公钥和用户证书公共名；将所述用户证书公共名作为数据库用户名，并生成所述数据库用户名的标准信息摘要值；接收客户端发送的认证请求并对所述认证请求中的用户证书进行验证，通过验证后向客户端返回初始随机数；接收客户端发送的签名随机数，利用所述公钥对所述签名随机数解密得到随机数；确认所述随机数与所述初始随机数一致则获取客户端用户名的信息摘要值，若客户端用户名的信息摘要值与所述标准信息摘要值一致，则判定所述客户端通过认证。本发明增加了数据库的安全性和用户的保密性。



CN 112491843 B

1.一种数据库多重认证方法,其特征在于,包括:

通过编程接口初始化硬件认证证书载体,并保存所述硬件认证证书载体的公钥和用户证书公共名;

将所述用户证书公共名作为数据库用户名,并生成所述数据库用户名的标准信息摘要值;

接收客户端发送的认证请求并对所述认证请求中的用户证书进行验证,通过验证后向客户端返回初始随机数;

接收客户端发送的签名随机数,利用所述公钥对所述签名随机数解密得到随机数;

确认所述随机数与所述初始随机数一致则获取客户端用户名的信息摘要值,若客户端用户名的信息摘要值与所述标准信息摘要值一致,则判定所述客户端通过认证。

2.根据权利要求1所述的方法,其特征在于,所述通过编程接口初始化硬件认证证书载体,包括:

向所述硬件认证证书载体认证管理函数输入用户信息,登录所述认证管理函数;

利用所述认证管理函数生成密钥对,所述密钥对包括私钥和公钥,将所述私钥保存至硬件认证证书载体的存储介质;

利用证书生成工具根据所述密钥对生成用户证书;

指定所述硬件认证证书载体的管理员账户,并设置管理员和用户登录所述认证管理函数的最大尝试次数。

3.根据权利要求1所述的方法,其特征在于,所述客户端发送认证请求的方法包括:

所述客户端识别硬件认证证书载体;

所述客户端向所述硬件认证证书载体输入用户信息,登录所述硬件认证证书载体;

所述客户端从所述硬件认证证书载体读取用户证书,并根据所述用户证书生成认证请求。

4.根据权利要求1所述的方法,其特征在于,所述接收客户端发送的认证请求并对所述认证请求中的用户证书进行验证通过验证后向客户端返回初始随机数,包括:

从所述认证请求中提取用户证书,并判断所述用户证书与本地存储的合法用户证书是否匹配:

若是,则生成指定位数的随机数,并将所述随机数发送客户端的同时作为初始随机数保存至本地;

若否,则判定认证失败。

5.根据权利要求1所述的方法,其特征在于,所述确认随机数与所述初始随机数一致则获取客户端用户名的信息摘要值,包括:

在所述随机数与所述初始随机数一致性验证通过后,获取客户端从硬件认证证书载体读取的用户证书公共名;

利用信息摘要算法计算所述用户证书公共名的信息摘要值。

6.一种数据库多重认证系统,其特征在于,包括:

初始设置单元,配置用于通过编程接口初始化硬件认证证书载体,并保存所述硬件认证证书载体的公钥和用户证书公共名;

本地设置单元,配置用于将所述用户证书公共名作为数据库用户名,并生成所述数据

库用户名的标准信息摘要值；

证书验证单元，配置用于接收客户端发送的认证请求并对所述认证请求中的用户证书进行验证，通过验证后向客户端返回初始随机数；

动态验证单元，配置用于接收客户端发送的签名随机数，利用所述公钥对所述签名随机数解密得到随机数；

摘要验证单元，配置用于确认所述随机数与所述初始随机数一致则获取客户端用户名的信息摘要值，若客户端用户名的信息摘要值与所述标准信息摘要值一致，则判定所述客户端通过认证。

7.根据权利要求6所述的系统，其特征在于，所述初始设置单元包括：

用户登录模块，配置用于向所述硬件认证证书载体认证管理函数输入用户信息，登录所述认证管理函数；

密钥生成模块，配置用于利用所述认证管理函数生成密钥对，所述密钥对包括私钥和公钥，将所述私钥保存至硬件认证证书载体的存储介质；

证书生成模块，配置用于利用证书生成工具根据所述密钥对生成用户证书；

次数设置模块，配置用于指定所述硬件认证证书载体的管理员账户，并设置管理员和用户登录所述认证管理函数的最大尝试次数。

8.根据权利要求6所述的系统，其特征在于，所述证书验证单元包括：

证书匹配模块，配置用于从所述认证请求中提取用户证书，并判断所述用户证书与本地存储的合法用户证书是否匹配；

口令生成模块，配置用于若所述用户证书与本地存储的合法用户证书匹配，则生成指定位数的随机数，并将所述随机数发送客户端的同时作为初始随机数保存至本地；

失败判定模块，配置用于若所述用户证书与本地存储的合法用户证书不匹配，则判定认证失败。

9.一种终端，其特征在于，包括：

处理器；

用于存储处理器的执行指令的存储器；

其中，所述处理器被配置为执行权利要求1-5任一项所述的方法。

10.一种存储有计算机程序的计算机可读存储介质，其特征在于，该程序被处理器执行时实现如权利要求1-5中任一项所述的方法。

一种数据库多重认证方法、系统、终端及存储介质

技术领域

[0001] 本发明涉及数据库认证技术领域，具体涉及一种数据库多重认证方法、系统、终端及存储介质。

背景技术

[0002] 伴随信息化的持续推进，数据的安全和业务运行的可靠性越来越重要。身份鉴别做完登录数据库用户的合法认证，计算机网络世界中一切信息包括用户的身份信息都是用一组特定的数据来表示的，计算机只能识别用户的数字身份，所有对用户的授权也是针对用户数字身份的授权。

[0003] 如何保证以数字身份进行操作的操作者就是这个数字身份合法拥有者，也就是说保证操作者的物理身份与数字身份相对应，身份认证就是为了解决这个问题，作为防护网络资产的第一道关口，身份认证有着举足轻重的作用。

[0004] PostgreSQL目前只支持设置一种认证方式，大部分使用md5口令认证，md5口令的安全性并不强，因此具有密码泄露的风险。

发明内容

[0005] 针对现有技术的上述不足，本发明提供一种数据库多重认证方法、系统、终端及存储介质，以解决上述技术问题。

[0006] 第一方面，本发明提供一种数据库多重认证方法，包括：

[0007] 通过编程接口初始化硬件认证证书载体，并保存所述硬件认证证书载体的公钥和用户证书公共名；

[0008] 将所述用户证书公共名作为数据库用户名，并生成所述数据库用户名的标准信息摘要值；

[0009] 接收客户端发送的认证请求并对所述认证请求中的用户证书进行验证，通过验证后向客户端返回初始随机数；

[0010] 接收客户端发送的签名随机数，利用所述公钥对所述签名随机数解密得到随机数；

[0011] 确认所述随机数与所述初始随机数一致则获取客户端用户名的信息摘要值，若客户端用户名的信息摘要值与所述标准信息摘要值一致，则判定所述客户端通过认证。

[0012] 进一步的，所述通过编程接口初始化硬件认证证书载体，包括：

[0013] 向所述硬件认证证书载体认证管理函数输入用户信息，登录所述认证管理函数；

[0014] 利用所述认证管理函数生成密钥对，所述密钥对包括私钥和公钥，将所述私钥保存至硬件认证证书载体的存储介质；

[0015] 利用证书生成工具根据所述密钥对生成用户证书；

[0016] 指定所述硬件认证证书载体的管理员账户，并设置管理员和用户登录所述认证管理函数的最大尝试次数。

[0017] 进一步的,所述客户端发送认证请求的方法包括:

[0018] 所述客户端识别硬件认证证书载体;

[0019] 所述客户端向所述硬件认证证书载体输入用户信息,登录所述硬件认证证书载体;

[0020] 所述客户端从所述硬件认证证书载体读取用户证书,并根据所述用户证书生成认证请求。

[0021] 进一步的,所述接收客户端发送的认证请求并对所述认证请求中的用户证书进行验证通过验证后向客户端返回初始随机数,包括:

[0022] 从所述认证请求中提取用户证书,并判断所述用户证书与本地存储的合法用户证书是否匹配:

[0023] 若是,则生成指定位数的随机数,并将所述随机数发送客户端的同时作为初始随机数保存至本地;

[0024] 若否,则判定认证失败。

[0025] 进一步的,所述确认随机数与所述初始随机数一致则获取客户端用户名的信息摘要值,包括:

[0026] 在所述随机数与所述初始随机数一致性验证通过后,获取客户端从硬件认证证书载体读取的用户证书公共名;

[0027] 利用信息摘要算法计算所述用户证书公共名的信息摘要值。

[0028] 第二方面,本发明提供一种数据库多重认证系统,包括:

[0029] 初始设置单元,配置用于通过编程接口初始化硬件认证证书载体,并保存所述硬件认证证书载体的公钥和用户证书公共名;

[0030] 本地设置单元,配置用于将所述用户证书公共名作为数据库用户名,并生成所述数据库用户名的标准信息摘要值;

[0031] 证书验证单元,配置用于接收客户端发送的认证请求并对所述认证请求中的用户证书进行验证,通过验证后向客户端返回初始随机数;

[0032] 动态验证单元,配置用于接收客户端发送的签名随机数,利用所述公钥对所述签名随机数解密得到随机数;

[0033] 摘要验证单元,配置用于确认所述随机数与所述初始随机数一致则获取客户端用户名的信息摘要值,若客户端用户名的信息摘要值与所述标准信息摘要值一致,则判定所述客户端通过认证。

[0034] 进一步的,所述初始设置单元包括:

[0035] 用户登录模块,配置用于向所述硬件认证证书载体认证管理函数输入用户信息,登录所述认证管理函数;

[0036] 密钥生成模块,配置用于利用所述认证管理函数生成密钥对,所述密钥对包括私钥和公钥,将所述私钥保存至硬件认证证书载体的存储介质;

[0037] 证书生成模块,配置用于利用证书生成工具根据所述密钥对生成用户证书;

[0038] 次数设置模块,配置用于指定所述硬件认证证书载体的管理员账户,并设置管理员和用户登录所述认证管理函数的最大尝试次数。

[0039] 进一步的,所述证书验证单元包括:

[0040] 证书匹配模块,配置用于从所述认证请求中提取用户证书,并判断所述用户证书与本地存储的合法用户证书是否匹配;

[0041] 口令生成模块,配置用于若所述用户证书与本地存储的合法用户证书匹配,则生成指定位数的随机数,并将所述随机数发送客户端的同时作为初始随机数保存至本地;

[0042] 失败判定模块,配置用于若所述用户证书与本地存储的合法用户证书不匹配,则判定认证失败。

[0043] 第三方面,提供一种终端,包括:

[0044] 处理器、存储器,其中,

[0045] 该存储器用于存储计算机程序,

[0046] 该处理器用于从存储器中调用并运行该计算机程序,使得终端执行上述的终端的方法。

[0047] 第四方面,提供了一种计算机存储介质,所述计算机可读存储介质中存储有指令,当其在计算机上运行时,使得计算机执行上述各方面所述的方法。

[0048] 本发明的有益效果在于,

[0049] 本发明提供的数据库多重认证方法、系统、终端及存储介质,通过增加硬件认证证书载体(USBKey)USBKey的认证方式,并且与md5口令认证相结合,实现数据库的多重认证。本发明针对PostgreSQL数据库的身份鉴别功能,增加了usb key的认证方法,并结合了usb key认证以及md5认证,增加了数据库的安全性和用户的保密性。

[0050] 此外,本发明设计原理可靠,结构简单,具有非常广泛的应用前景。

附图说明

[0051] 为了更清楚地说明本发明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,对于本领域普通技术人员而言,在不付出创造性劳动的前提下,还可以根据这些附图获得其他的附图。

[0052] 图1是本发明一个实施例的方法的示意性流程图。

[0053] 图2是本发明一个实施例的方法的另一示意性流程图。

[0054] 图3是本发明一个实施例的系统的示意性框图。

[0055] 图4为本发明实施例提供的一种终端的结构示意图。

具体实施方式

[0056] 为了使本技术领域的人员更好地理解本发明中的技术方案,下面将结合本发明实施例中的附图,对本发明实施例中的技术方案进行清楚、完整地描述,显然,所描述的实施例仅仅是本发明一部分实施例,而不是全部的实施例。基于本发明中的实施例,本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例,都应当属于本发明保护的范围。

[0057] 下面对本发明中出现的关键术语进行解释。

[0058] 1)PostgreSQL数据库

[0059] 是一种特性非常齐全的自由软件的对象-关系型数据库管理系统(ORDBMS)PostgreSQL支持大部分的SQL标准并且提供了很多其他现代特性,如复杂查询、外键、触发

器、视图、事务完整性、多版本并发控制等。同样，PostgreSQL也可以用许多方法扩展，例如通过增加新的数据类型、函数、操作符、聚集函数、索引方法、过程语言等。另外，因为许可证的灵活，任何人都可以以任何目的免费使用、修改和分发PostgreSQL。

[0060] 2)身份鉴别

[0061] 是指在计算机及计算机网络系统中确认操作者身份的过程，从而确定该用户是否具有对某种资源的访问和使用权限，进而使计算机和网络系统的访问策略能够可靠、有效地执行，防止攻击者假冒合法用户获得资源的访问权限，保证系统和数据的安全，以及授权访问者的合法利益。

[0062] 3)USB Key

[0063] 是一种USB接口的硬件设备。它内置单片机或智能卡芯片，有一定的存储空间，可以存储用户的私钥以及数字证书，利用USB Key内置的公钥算法实现对用户身份的认证。由于用户私钥保存在密码锁中，理论上使用任何方式都无法读取，因此保证了用户认证的安全性。

[0064] 4)PKCS#11标准

[0065] 定义了与密码令牌(如硬件安全模块(HSM)和智能卡)的独立于平台的API，并将API本身命名为“Cryptoki”(来自“加密令牌接口”，发音为“crypto-key”-但是“PKCS#11”通常用于指代API以及定义它的标准)。API定义了最常用的加密对像类型(RSA密钥，X.509证书，DES/三重DES密钥等)以及使用，创建/生成，修改和删除这些对象所需的所有功能。

[0066] 图1是本发明一个实施例的方法的示意性流程图。其中，图1执行主体可以为一种数据库多重认证系统。

[0067] 如图1所示，该方法包括：

[0068] 步骤110，通过编程接口初始化硬件认证证书载体，并保存所述硬件认证证书载体的公钥和用户证书公共名；

[0069] 步骤120，将所述用户证书公共名作为数据库用户名，并生成所述数据库用户名的标准信息摘要值；

[0070] 步骤130，接收客户端发送的认证请求并对所述认证请求中的用户证书进行验证，通过验证后向客户端返回初始随机数；

[0071] 步骤140，接收客户端发送的签名随机数，利用所述公钥对所述签名随机数解密得到随机数；

[0072] 步骤150，确认所述随机数与所述初始随机数一致则获取客户端用户名的信息摘要值，若客户端用户名的信息摘要值与所述标准信息摘要值一致，则判定所述客户端通过认证。

[0073] 具体的，如图2所示，所述数据库多重认证方法包括：

[0074] S1、通过编程接口初始化硬件认证证书载体，并保存所述硬件认证证书载体的公钥和用户证书公共名。

[0075] 硬件认证证书载体初始化方法包括以下步骤：

[0076] 1)本发明选择的编程接口为pkcs#11(cryptoki)。使用pkcs#11库流程，首先是初始化C_Initialize，获取当前连接的槽信息C_GetSlotList，打开会话C_OpenSession，获取token的信息C_GetTokenInfo，然后用用户的pin登录C_Login，接着进行相应的操作(生成

密钥对,签名,加密,读取key中的证书,写入证书),操作完成关闭会话,完成退出。

[0077] 2)为需要使用数据库的用户提供一个usb_key,key中包括一对rsa密钥,是通过pkcs#11的接口C_GnerateKeyPair生成的,保证私钥不出key。还包括由这个密钥对生成的个人数字证书。

[0078] 3)通过调用C_InitToke初始化一个key,指定key的名称输入key的管理员pin(SO pin)、设置用户pin。设置管理员pin和用户pin的最大尝试次数,划分key中的公开存储区和隐蔽存储区的大小。

[0079] 4)通过调用C_GnerateKeyPair为使用这个key的用户生成一对rsa密钥,这对密钥在key中保存,保证私钥永不出key。然后需要利用这对密钥完成用户证书的制作。制作证书的思路是用工具生成用户证书请求,再把这个证书请求交个CA签名成为用户证书。为了利用key中生成的密钥对制作证书请求,需要用到openssl的证书请求生成工具,修改它的处理流程,通过pkcs#11的库访问usb_key获取生成证书请求所需要的公钥和用相应的私钥签名。

[0080] 5)再把生成的证书请求提交给CA签名生成证书,将生成的数字证书写回usb_key中,这样用于PostgreSQL客户身份鉴别的usb_key初始化完成,可以交给用户了。

[0081] 将上述初始化过程中生成的公钥、用户证书和用户证书公共名保存至服务端数据库。

[0082] S2、将所述用户证书公共名作为数据库用户名,并生成所述数据库用户名的标准信息摘要值。

[0083] 将用户的证书公共名作为数据库用户名,利用MD5信息摘要算法计算数据库用户名的MD5值作为初始MD5值。

[0084] S3、接收客户端发送的认证请求并对所述认证请求中的用户证书进行验证,通过验证后向客户端返回初始随机数。

[0085] 客户端判断usb_key是否连接,让用户输入pin码,pin码验证通过再向服务器发送认证请求,请求中包括用户的证书。

[0086] 服务器验证认证请求中的用户证书的真伪,如果是合法的用户证书则证书认证通过;服务器端生成一个随机数,并用的证书公钥加密随机数发给客户端。同时将该随机数保存为初始随机数。

[0087] S4、接收客户端发送的签名随机数,利用所述公钥对所述签名随机数解密得到随机数。

[0088] 客户端将服务器发送的随机数交给usb_key解密并签名,然后发送给服务器认证。

[0089] 服务器利用证书的公钥对客户端签名的随机数进行验证,将解密后得到的随机数与初始随机数比对,若两者一致,即解密后的随机数如果是服务器发送的随机数则通过,否则认证失败。

[0090] S5、确认所述随机数与所述初始随机数一致则获取客户端用户名的信息摘要值,若客户端用户名的信息摘要值与所述标准信息摘要值一致,则判定所述客户端通过认证。

[0091] 客户端收到证书验证通过的消息进行获取客户端的用户名和口令操作,从USBKey中读取名称。服务端从客户端获取该读取的名称,并计算其MD5值,若该MD5值与初始MD5值一致,则判定该客户端的认证请求通过认证,成功连接数据库。

[0092] 如图3所示，该系统300包括：

[0093] 初始设置单元310，配置用于通过编程接口初始化硬件认证证书载体，并保存所述硬件认证证书载体的公钥和用户证书公共名；

[0094] 本地设置单元320，配置用于将所述用户证书公共名作为数据库用户名，并生成所述数据库用户名的标准信息摘要值；

[0095] 证书验证单元330，配置用于接收客户端发送的认证请求并对所述认证请求中的用户证书进行验证，通过验证后向客户端返回初始随机数；

[0096] 动态验证单元340，配置用于接收客户端发送的签名随机数，利用所述公钥对所述签名随机数解密得到随机数；

[0097] 摘要验证单元350，配置用于确认所述随机数与所述初始随机数一致则获取客户端用户名的信息摘要值，若客户端用户名的信息摘要值与所述标准信息摘要值一致，则判定所述客户端通过认证。

[0098] 可选地，作为本发明一个实施例，所述初始设置单元包括：

[0099] 用户登录模块，配置用于向所述硬件认证证书载体认证管理函数输入用户信息，登录所述认证管理函数；

[0100] 密钥生成模块，配置用于利用所述认证管理函数生成密钥对，所述密钥对包括私钥和公钥，将所述私钥保存至硬件认证证书载体的存储介质；

[0101] 证书生成模块，配置用于利用证书生成工具根据所述密钥对生成用户证书；

[0102] 次数设置模块，配置用于指定所述硬件认证证书载体的管理员账户，并设置管理员和用户登录所述认证管理函数的最大尝试次数。

[0103] 可选地，作为本发明一个实施例，所述证书验证单元包括：

[0104] 证书匹配模块，配置用于从所述认证请求中提取用户证书，并判断所述用户证书与本地存储的合法用户证书是否匹配；

[0105] 口令生成模块，配置用于若所述用户证书与本地存储的合法用户证书匹配，则生成指定位数的随机数，并将所述随机数发送客户端的同时作为初始随机数保存至本地；

[0106] 失败判定模块，配置用于若所述用户证书与本地存储的合法用户证书不匹配，则判定认证失败。

[0107] 图4为本发明实施例提供的一种终端400的结构示意图，该终端400可以用于执行本发明实施例提供的数据库多重认证方法。

[0108] 其中，该终端400可以包括：处理器410、存储器420及通信单元430。这些组件通过一条或多条总线进行通信，本领域技术人员可以理解，图中示出的服务器的结构并不构成对本发明的限定，它既可以是总线形结构，也可以是星型结构，还可以包括比图示更多或更少的部件，或者组合某些部件，或者不同的部件布置。

[0109] 其中，该存储器420可以用于存储处理器410的执行指令，存储器420可以由任何类型的易失性或非易失性存储终端或者它们的组合实现，如静态随机存取存储器(SRAM)，电可擦除可编程只读存储器(EEPROM)，可擦除可编程只读存储器(EPROM)，可编程只读存储器(PROM)，只读存储器(ROM)，磁存储器，快闪存储器，磁盘或光盘。当存储器420中的执行指令由处理器410执行时，使得终端400能够执行以下上述方法实施例中的部分或全部步骤。

[0110] 处理器410为存储终端的控制中心，利用各种接口和线路连接整个电子终端的各

个部分，通过运行或执行存储在存储器420内的软件程序和/或模块，以及调用存储在存储器内的数据，以执行电子终端的各种功能和/或处理数据。所述处理器可以由集成电路(Integrated Circuit，简称IC)组成，例如可以由单颗封装的IC所组成，也可以由连接多颗相同功能或不同功能的封装IC而组成。举例来说，处理器410可以仅包括中央处理器(Central Processing Unit，简称CPU)。在本发明实施方式中，CPU可以是单运算核心，也可以包括多运算核心。

[0111] 通信单元430，用于建立通信信道，从而使所述存储终端可以与其它终端进行通信。接收其他终端发送的用户数据或者向其他终端发送用户数据。

[0112] 本发明还提供一种计算机存储介质，其中，该计算机存储介质可存储有程序，该程序执行时可包括本发明提供的各实施例中的部分或全部步骤。所述的存储介质可为磁碟、光盘、只读存储记忆体(英文：read-only memory，简称：ROM)或随机存储记忆体(英文：random access memory，简称：RAM)等。

[0113] 因此，本发明通过增加硬件认证证书载体(USBKey)USBKey的认证方式，并且与md5口令认证相结合，实现数据库的多重认证。本发明针对PostgreSQL数据库的身份鉴别功能，增加了usb key的认证方法，并结合了usb key认证以及md5认证，增加了数据库的安全性和用户的保密性，本实施例所能达到的技术效果可以参见上文中的描述，此处不再赘述。

[0114] 本领域的技术人员可以清楚地了解到本发明实施例中的技术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，本发明实施例中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，该计算机软件产品存储在一个存储介质中如U盘、移动硬盘、只读存储器(ROM，Read-Only Memory)、随机存取存储器(RAM，Random Access Memory)、磁碟或者光盘等各种可以存储程序代码的介质，包括若干指令用以使得一台计算机终端(可以是个人计算机，服务器，或者第二终端、网络终端等)执行本发明各个实施例所述方法的全部或部分步骤。

[0115] 本说明书中各个实施例之间相同相似的部分互相参见即可。尤其，对于终端实施例而言，由于其基本相似于方法实施例，所以描述的比较简单，相关之处参见方法实施例中的说明即可。

[0116] 在本发明所提供的几个实施例中，应该理解到，所揭露的系统和方法，可以通过其它的方式实现。例如，以上所描述的系统实施例仅仅是示意性的，例如，所述单元的划分，仅仅为一种逻辑功能划分，实际实现时可以有另外的划分方式，例如多个单元或组件可以结合或者可以集成到另一个系统，或一些特征可以忽略，或不执行。另一点，所显示或讨论的相互之间的耦合或直接耦合或通信连接可以是通过一些接口，系统或单元的间接耦合或通信连接，可以是电性，机械或其它的形式。

[0117] 所述作为分离部件说明的单元可以是或者也可以不是物理上分开的，作为单元显示的部件可以是或者也可以不是物理单元，即可以位于一个地方，或者也可以分布到多个网络单元上。可以根据实际的需要选择其中的部分或者全部单元来实现本实施例方案的目的。

[0118] 另外，在本发明各个实施例中的各功能单元可以集成在一个处理单元中，也可以是各个单元单独物理存在，也可以两个或两个以上单元集成在一个单元中。

[0119] 尽管通过参考附图并结合优选实施例的方式对本发明进行了详细描述，但本发明

并不限于此。在不脱离本发明的精神和实质的前提下，本领域普通技术人员可以对本发明的实施例进行各种等效的修改或替换，而这些修改或替换都应在本发明的涵盖范围内/任何熟悉本技术领域的技术人员在本发明揭露的技术范围内，可轻易想到变化或替换，都应涵盖在本发明的保护范围之内。因此，本发明的保护范围应所述以权利要求的保护范围为准。

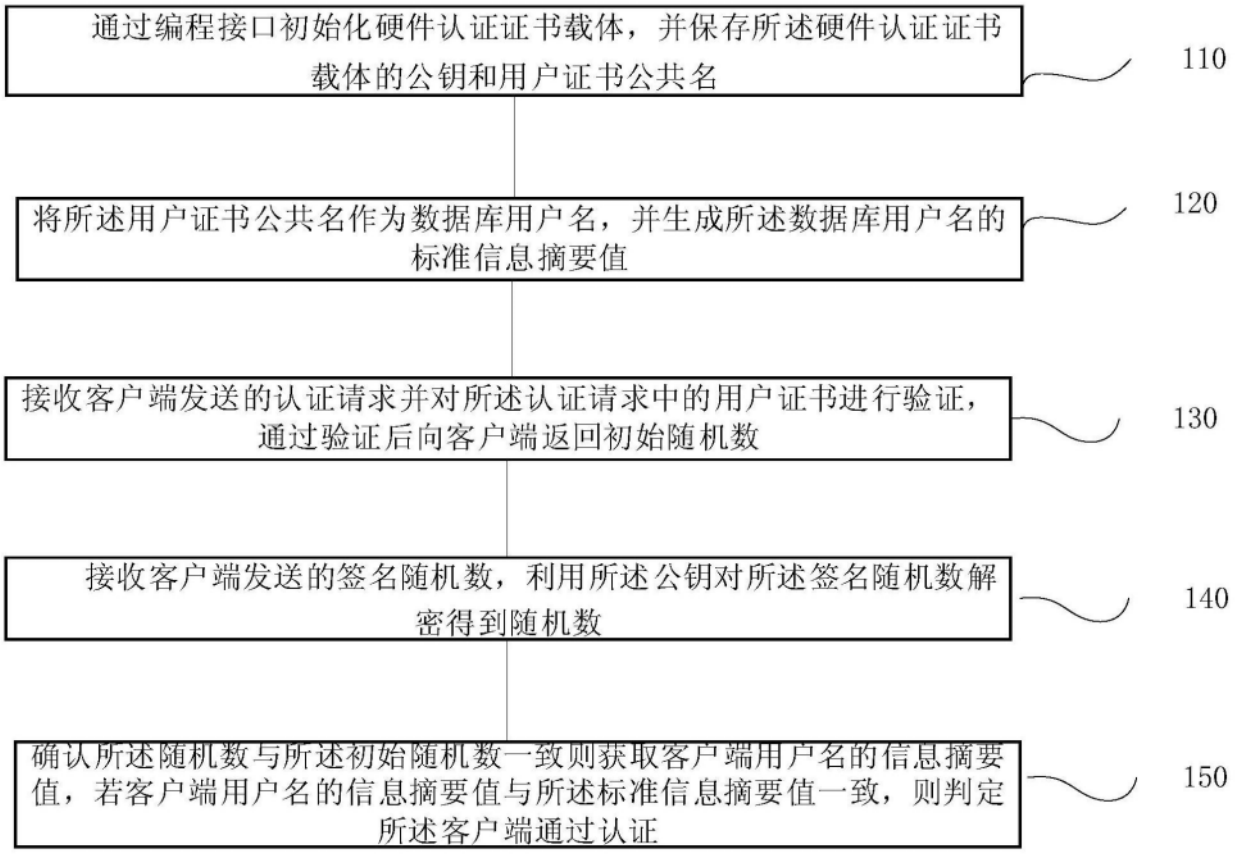

通过编程接口初始化硬件认证证书载体，并保存所述硬件认证证书载体的公钥和用户证书公共名
110
将所述用户证书公共名作为数据库用户名，并生成所述数据库用户名的标准信息摘要值
120
接收客户端发送的认证请求并对所述认证请求中的用户证书进行验证，通过验证后向客户端返回初始随机数
130
接收客户端发送的签名随机数，利用所述公钥对所述签名随机数解密得到随机数
140
确认所述随机数与所述初始随机数一致则获取客户端用户名的信息摘要值，若客户端用户名的信息摘要值与所述标准信息摘要值一致，则判定所述客户端通过认证
150

图1

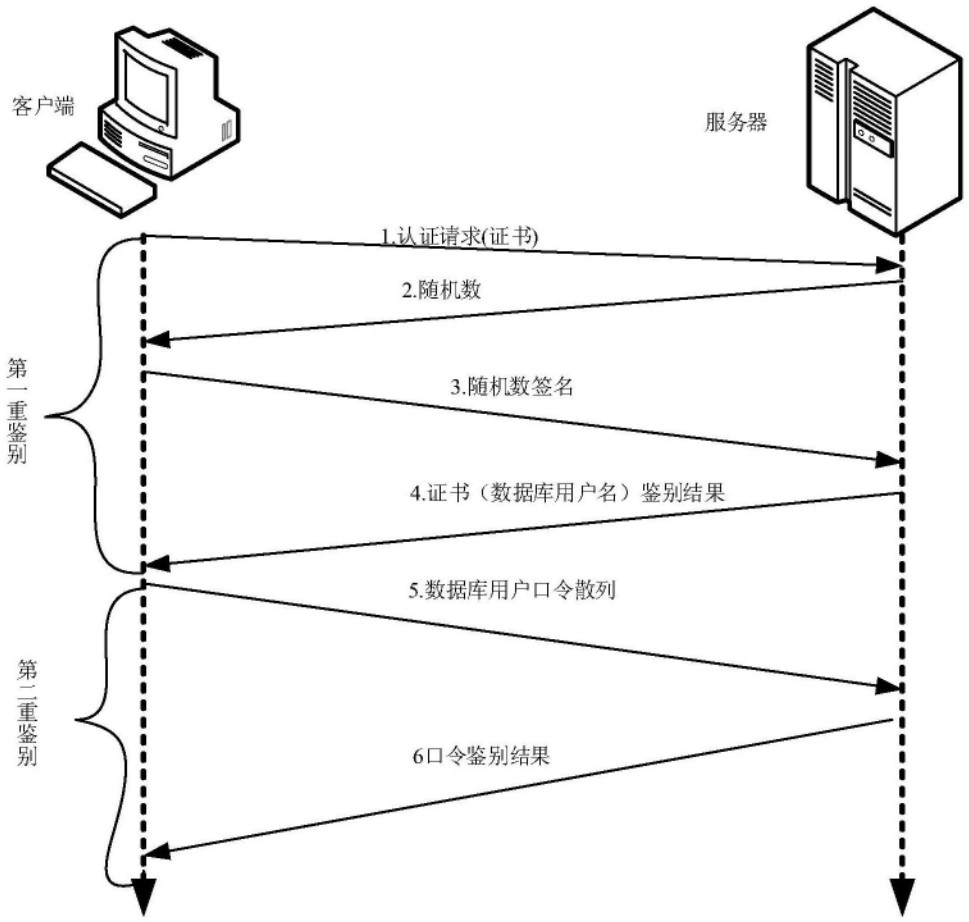

客户端
服务器
1.认证请求(证书)
2.随机数
第一重鉴别
3.随机数签名
4.证书（数据库用户名）鉴别结果
5.数据库用户口令散列
第二重鉴别
6口令鉴别结果

图2

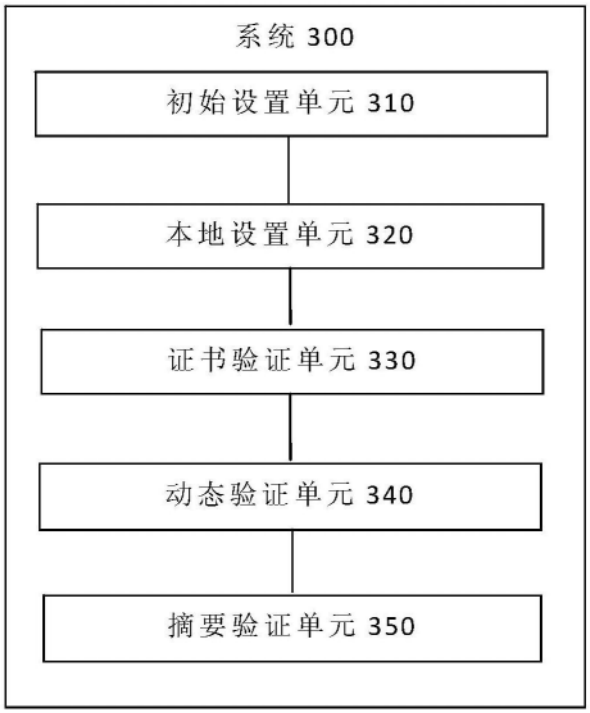

系统 300
初始设置单元 310
本地设置单元 320
证书验证单元 330
动态验证单元 340
摘要验证单元 350

图3

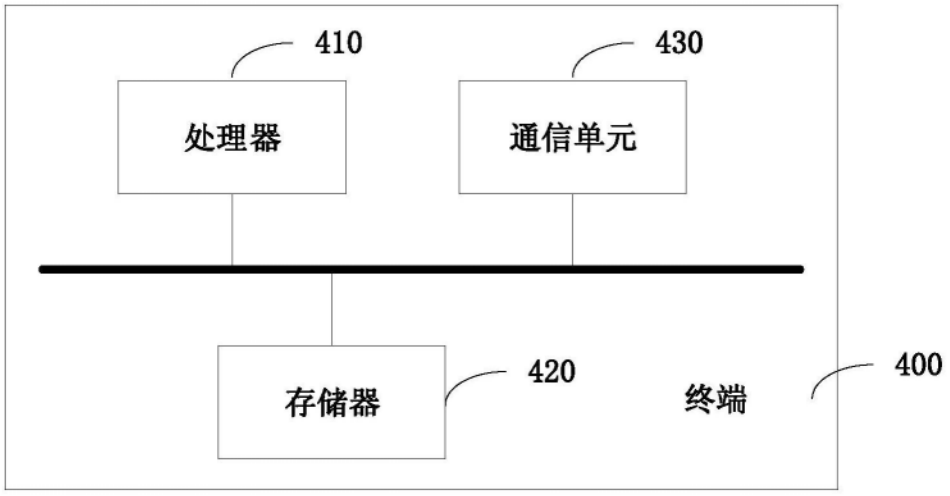

410
处理器
430
通信单元
存储器
420
终端
400

图4