(19)国家知识产权局

(12)发明专利

(10)授权公告号 CN 110875925 B
(45)授权公告日 2022.07.19

(21)申请号 201910789691.0

(22)申请日 2019.08.26

(65)同一申请的已公布的文献号
申请公布号 CN 110875925 A

(43)申请公布日 2020.03.10

(30)优先权数据
2018-162103 2018.08.30 JP

(73)专利权人 佳能株式会社
地址 日本国东京都大田区下丸子3丁目30-2

(72)发明人 岸本凉

(74)专利代理机构 北京怡丰知识产权代理有限公司 11293
专利代理师 迟军 李艳丽

(51)Int.Cl.
H04L 9/40(2022.01)
H04L 9/08(2006.01)
H04L 9/32(2006.01)
H04L 67/02(2022.01)

(56)对比文件
CN 108432180 A,2018.08.21
CN 108432180 A,2018.08.21
CN 107659406 A,2018.02.02
US 2014007198 A1,2014.01.02
CN 103109499 A,2013.05.15
JP 2018093407 A,2018.06.14
JP 2018032085 A,2018.03.01
JP 2014142732 A,2014.08.07
US 2014380428 A1,2014.12.25
慕鹏."面向服务架构的工作流系统的设计与实现".《中国优秀硕士学位论文全文数据库(电子期刊) 信息科技辑》.2017,(第3期),I138-1842.
李怀明等.基于时间和环境约束的四层访问控制模型研究.《计算机应用与软件》.2018,(第01期),

审查员 王务鹏

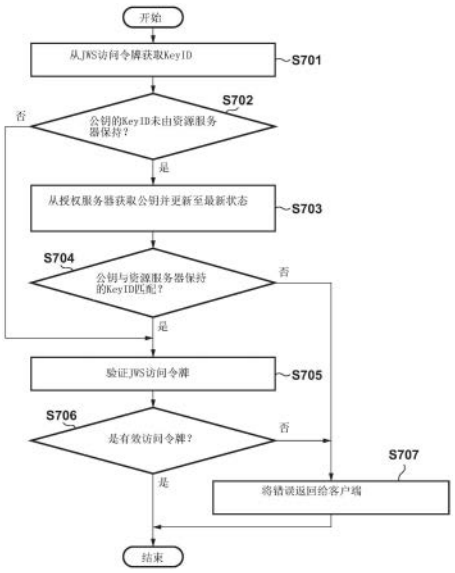
权利要求书1页 说明书13页 附图10页

(54)发明名称

信息处理装置、授权系统及验证方法

(57)摘要

本发明公开一种信息处理装置、授权系统及验证方法。根据本发明,提供一种验证签名令牌的信息处理装置。所述装置包括:用于保持用以验证签名令牌的密钥信息的保持单元;用于在用以验证接收到的签名令牌的密钥信息未被保持在保持单元中的情况下、从提供密钥信息的服务器中获取新的密钥信息并将新的密钥信息保持在保持单元中的获取单元;以及用于在用以验证接收到的签名令牌的密钥信息被保持在保持单元中的情况下利用密钥信息验证签名令牌的验证单元。



CN 110875925 B

1.一种作为资源服务器进行签名令牌的验证的信息处理装置,所述信息处理装置包括:

存储器,用于保持用以验证的第一密钥信息;

接收单元,用于接收签名令牌;

获取单元,用于在针对接收到的签名令牌的密钥信息未被保持在存储器中并且在第一密钥信息的获取时间之后发布接收到的签名令牌的情况下,从提供第二密钥信息的授权服务器中获取第二密钥信息;

保持单元,用于保持存储器中的第二密钥信息;以及

验证单元,用于在针对接收到的签名令牌的密钥信息保持在存储器中的情况下,利用密钥信息进行接收到的签名令牌的验证,

其中,在与请求一起接收到的签名令牌的验证成功的情况下,响应于请求而提供资源。

2.根据权利要求1所述的信息处理装置,

其中,所述信息处理装置还包括:

保存单元,用于保存从授权服务器获取第二密钥信息的获取时间。

3.根据权利要求1所述的信息处理装置,

其中,接收到的签名令牌包括密钥信息的识别信息,以及

其中,在针对接收到的签名令牌的密钥信息未被保持在存储器中并且包含在接收到的签名令牌中的识别信息已由授权服务器发布的情况下,从授权服务器获取第二密钥信息。

4.根据权利要求1所述的信息处理装置,

其中,信息处理装置与授权服务器、以及与授权服务器提供的签名令牌一起发送资源请求的客户端相连接。

5.一种由作为资源服务器进行签名令牌的验证并且在存储器中保持用以验证的第一密钥信息的信息处理装置执行的签名令牌验证方法,所述签名令牌验证方法包括:

在针对接收到的签名令牌的密钥信息未被保持在存储器中并且在第一密钥信息的获取时间之后发布接收到的签名令牌的情况下,从提供第二密钥信息的授权服务器中获取第二密钥信息;

在存储器中保持第二密钥信息;以及

在针对接收到的签名令牌的密钥信息被保持在存储器中的情况下,利用密钥信息进行接收到的签名令牌的验证,

其中,在与请求一起接收到的签名令牌的验证成功的情况下,响应于请求而提供资源。

信息处理装置、授权系统及验证方法

技术领域

[0001] 本发明涉及一种用于验证使用诸如签名令牌授权令牌等的信息处理装置、授权系统及验证方法。

背景技术

[0002] 近年来，通过网络提供的基于web的服务对用户进行认证，以确定他们是否被授权使用这些服务。基于web的服务本身变得更加多样化，并且随着多个基于Web的服务相互结合使用，提供基于web的服务的资源服务器和进行认证/授权的认证/授权服务器被配置为单独的服务器。利用这种配置，认证/授权服务器和资源服务器通过认证/授权信息的交换来连接。认证/授权服务器发布包含授权信息的访问令牌，客户端使用访问令牌接收资源服务器提供的服务。OAuth 2.0是以这种方式交换授权信息的规范的示例。

[0003] 当实施传统的OAuth规范时，大体上，针对资源服务器从客户端接收的所有访问令牌，资源服务器基本上向认证/授权服务器(保持令牌状态的服务器)进行验证查询。同样，对于与令牌关联的用户信息，资源服务器通常向保持与令牌相关联的用户信息的授权服务器进行信息查询。因此，随着客户端和资源服务器的数量增加，从资源服务器向认证/授权服务器(保持已经发布的令牌的令牌信息和与令牌相关联的用户信息的服务器)进行的令牌验证和信息查询的数量也会增加，这会增加认证/授权服务器的负荷。另外，针对从客户端接收的所有令牌向认证/授权服务器进行验证和信息查询，这导致资源服务器的性能下降。

[0004] 附带地，可以想到将授权信息(令牌ID、范围、有效期等)和与令牌相关联的信息(用户ID、客户端ID、用户名、电子邮件地址等)添加到由认证/授权服务器发布的各个令牌，以使资源服务器本身能够验证从客户端接收的访问令牌。资源服务器中的验证可以通过认证/授权服务器向令牌添加签名且资源服务器验证该签名来实现。使用JWS(JSON Web Signature)是表达签名令牌的一种可能方法。JWS是用于表达受数字签名、消息认证码(MACs)等保护的JWT(JSON Web Token)内容的手段。JWT也是一种使用基于JSON(JavaScript Object Notation(Java是注册商标))的数据结构来表达URL安全声明的方法。

[0005] 诸如日本特开2007-149010号公报中公开的方法等方法可以作为即使在已经接受了令牌转让的系统使用该令牌时的时间点也能够可靠地验证令牌的权利管理系统的示例。

[0006] 认证/授权服务器可以保持多对私钥和公钥。如果私钥泄露、使用的加密算法已被破坏等，则认证/授权服务器还可以通过注册和使用新密钥来提高安全性。此外，通过为各个要授权的服务或客户端使用不同的私钥，可以增加各个服务的安全性。然而，认证/授权服务器和资源服务器由单独的服务器构成，因此，即使在认证/授权服务器中注册了新的密钥对，资源服务器也不知道这一点，且不能获得公钥。因此，在使用资源服务器验证签名令牌的系统中，可能发布使用与未由资源服务器保持的公钥对应的私钥签名的访问令牌。在这种情况下，资源服务器将无法验证令牌，即使客户端使用由认证/授权服务器发布的访问

令牌,资源服务器也将确定签名不正确。

发明内容

[0007] 本发明使得,当将密钥添加到认证/授权服务器时,即使没有通知资源服务器已经添加了密钥,资源服务器也可以验证利用所添加的密钥签名的签名令牌。

[0008] 本发明具有以下配置。即:根据本发明的第一方面,提供了一种验证签名令牌的信息处理装置,所述装置包括:用于保持用以验证签名令牌的密钥信息的保持单元;用于在用以验证接收到的签名令牌的密钥信息未被保持在保持单元中的情况下、从提供密钥信息的服务器中获取新的密钥信息并将新的密钥信息保持在保持单元中的获取单元;以及用于在用以验证接收到的签名令牌的密钥信息保持在保持单元中的情况下利用密钥信息验证签名令牌的验证单元。

[0009] 根据本发明的第二方面,提供一种授权系统,所述授权系统包括:提供密钥信息和签名令牌的服务器;发送资源请求以及服务器提供的签名令牌的客户端;以及验证签名令牌的信息处理装置,所述装置包括:用于保持用以验证签名令牌的密钥信息的保持单元;用于在用以验证接收到的签名令牌的密钥信息未被保持在保持单元中的情况下、从提供密钥信息的服务器中获取新的密钥信息并将新的密钥信息保持在保持单元中的获取单元;以及用于在用以验证接收到的签名令牌的密钥信息被保持在保持单元中的情况下利用密钥信息验证签名令牌的验证单元。

[0010] 根据本发明的第三方面,提供一种由信息处理装置执行的签名令牌验证方法,所述方法包括:在保持单元保持用以验证签名令牌的密钥信息;在用以验证接收到的签名令牌的密钥信息未被保持在保持单元中的情况下,从提供密钥信息的服务器中获取新的密钥信息,并将新的密钥信息保持在保持单元中;以及在用以验证接收到的签名令牌的密钥信息被保持在保持单元中的情况下,利用密钥信息验证签名令牌。

[0011] 根据本发明,当将密钥添加到授权服务器时,即使没有通知资源服务器已经添加了该密钥,资源服务器也可以验证利用所添加的密钥签名的签名令牌。

[0012] 根据以下参照附图对示例性实施例的详细描述,本发明的其他特征将变得清楚。

附图说明

[0013] 图1是示出根据本发明实施例的计算机的配置的图。

[0014] 图2是示出根据本发明实施例的网络的配置的图。

[0015] 图3是示出访问令牌发布和验证的序列图。

[0016] 图4是示出根据本发明实施例的认证/授权服务器的功能框图。

[0017] 图5是示出根据本发明实施例的资源服务器的功能框图。

[0018] 图6是示出根据本发明实施例的JWS访问令牌发布和验证的序列图。

[0019] 图7是示出根据本发明第一实施例的JWS访问令牌验证的流程图。

[0020] 图8是示出根据本发明第二实施例的JWS访问令牌验证的流程图。

[0021] 图9是说明根据本发明第三实施例的JWS访问令牌验证的流程图。

[0022] 图10是示出根据本发明第四实施例的JWS访问令牌验证的流程图。

具体实施方式

[0023] 下面将参考附图描述本发明的实施例。首先，将描述由多个实施例共享的配置。

[0024] 系统配置

[0025] 图1是示出根据本发明实施例的信息处理装置的系统配置的图。信息处理装置是用作服务器、客户端等的计算机。如图1中所示，信息处理装置包括CPU 102、存储器103、存储设备104、视频接口105、输入/输出(下文中缩写为“I/O”)接口106和通信接口107。信息处理装置中的信息处理装置通过系统总线101彼此连接。CPU 102是中央处理单元，其控制系统总线101上的组成元件、计算和处理数据等。存储器103是存储数据、程序等的设备，并由RAM(随机存取存储器)、ROM(只读存储器)等构成。存储设备104写入和读出存储的数据。存储设备104包括硬盘驱动器(HDD)111、用作非易失性数据源的DVD-ROM驱动器112以及使用半导体存储器的固态驱动器(SSD)113。尽管未在图1中示出，但磁带驱动器、Floppy(注册商标)磁盘驱动器(FDD)、CD-ROM驱动器、CD/DVD-RAM驱动器、USB闪存驱动器等也可用作存储设备104。

[0026] 根据实施例的程序被从存储设备104中读取，存储在存储器103中，并由CPU 102执行。尽管根据实施例，配置为从存储设备104读取程序，但是也可以配置为从ROM(未示出)、经由通信接口107从外部等读取程序。

[0027] 视频接口105控制到显示设备114的显示输出。显示设备114可以是CRT类型、液晶类型等。诸如键盘115、指示设备116等的输入设备连接到I/O接口106。操作者通过操纵键盘115向信息处理装置发布操作命令等。指示设备116移动显示设备114中的光标以选择和操作菜单项、对象等。使用触摸面板等的操作输入也可以通过显示设备114进行。在这种情况下，显示设备114同时用作输出设备和输入设备。

[0028] 通信接口107通过计算机网络117与外部设备通信。连接到的网络是LAN、WAN，诸如因特网等的公共线路等。

[0029] 注意，服务器不需要包括诸如显示设备114、键盘115、鼠标11等用户接口设备以及用于那些设备的接口。

[0030] 图2是示出根据本发明实施例的网络配置的图。计算机网络211、212和213连接到互联网210，且认证/授权服务器201连接到计算机网络211。资源服务器202连接到计算机网络212，且客户端203连接到计算机网络213。在这些实施例中，资源服务器202通过因特网210连接到计算机网络，但是可以替代为与认证/授权服务器201位于同一LAN上。客户端203不限于如图2所示的连接到计算机网络213，其可以连接到计算机网络211或212。

[0031] 客户端203从认证/授权服务器201获得访问令牌。然后，客户端203使用获得的访问令牌从资源服务器202接受提供基于web的服务(下文中简称为“服务”)。在这些实施例中，认证/授权服务器201、资源服务器202和客户端203都具有图1所示的信息处理装置的配置。然而，该配置不限于图1中所示的配置，可以省略显示设备114，可以添加其他功能等。服务器也不限于单个计算机，而是可以替代为由多个机架式计算机构成的系统。

[0032] 签名访问令牌的发布和验证(传统)

[0033] 图3示出了根据传统技术的访问令牌发布和验证的流程。图3是根据传统技术与签名令牌(例如，签名访问令牌)的发布和验证有关的序列图。该流程在图2所示的认证/授权服务器201、客户端203和资源服务器202之间执行。这里，“签名”是指数字签名或电子签名。

例如，当要验证的数据被给予用于利用私钥加密该数据的数字签名并被发送时，数据的接收者用公钥解密该签名，并将解密结果与对象数据进行比较。如果数据匹配，则确定验证成功。尽管这仅仅是一个示例，但是通常根据诸如此类的过程来验证签名数据。在实施例中，“密钥”是数字信息，也可以称为“密钥信息”。此外，在以下实施例中，“认证”是例如使用ID、秘密信息等来确认对象(例如，用户)的权限的过程。“授权”是用于确认由已认证对象持有的权限转移给另一方(例如，客户等)的过程。这些仅是示例，定义不限于此。提供与该认证和授权有关的服务的服务器被称为“认证/授权服务器”，并且在以下描述中，根据所提供的服务，可以简称为“认证服务器”或“授权服务器”。

[0034] 首先，在步骤S301中，资源服务器202向认证/授权服务器201请求公钥，并获得公钥。接下来，在步骤S302中，客户端203将认证信息或授权代码传递到认证/授权服务器201，以请求并获得签名的访问令牌。然后，在步骤S303中，客户端203使用获得的签名访问令牌来请求资源服务器202提供服务。在步骤S304中，资源服务器202基于所接收的签名来验证访问令牌是否有效。此时，通过使用从认证/授权服务器201获得的公钥验证签名，资源服务器202可以确认访问令牌已经由有效的认证/授权服务器发布。然后，在步骤S305中，如果访问令牌有效，则资源服务器202执行用于提供服务的处理，并将处理结果传递给客户端203。注意，如果在步骤S304的验证中访问令牌无效，则资源服务器202不执行处理，而是将错误返回给客户端203。

[0035] 这里，可以在向客户端203提供服务之前(例如当资源服务器202被重置时等)执行一次图3中的步骤S301的公钥获取请求。然而，在步骤S301中获得公钥之后，存在对认证/授权服务器201添加新的私钥-公钥对的情况。在这种情况下，可以向客户端203发布已利用由授权/认证服务器201保持但未由资源服务器202保持的密钥签名的访问令牌。结果是，尽管访问令牌实际上是有效的，但在资源服务器202中验证签名时确定访问令牌无效。为了防止这种情况，当资源服务器202验证签名时，资源服务器202必须保持与认证/授权服务器201用于对访问令牌签名的私钥形成配对的公钥。

[0036] 这里，可以想到认证/授权服务器201在每次将新的私钥-公钥对添加到认证/授权服务器201时都通知资源服务器202已经添加了密钥。然而，这样做需要构建用于发送和接收通知的系统，并且认证/授权服务器201和资源服务器202二者都必须符合通知系统，这使得系统整体复杂化。即使进行了通知，在资源服务器获得公钥之前也会出现同样的问题，并且没有方法使得认证/授权服务器201可以通过该方法确认所有连接的资源服务器都已获得公钥。下面描述的用于处理该问题的方法不需要来自认证/授权服务器201的通知，并且使得可以正确地验证利用与资源服务器202未保持的公钥形成配对的私钥所添加的签名。此外，即使配置为资源服务器202周期性地向认证/授权服务器201检查是否已更新密钥，使用下文描述的方法仍使得可以比不使用该方法的情况设置更长的更新检查间隔。注意，下面描述的方法是用于处理已经将新密钥添加到资源服务器202的情况的方法，因此配置可以为使得仅当保持在资源服务器202中的密钥已被删除时才通知资源服务器202。接下来将描述根据第一实施例的用于处理上述问题的方法。

[0037] 第一实施例

[0038] JWS访问令牌

[0039] 接下来将详细描述在本实施例中使用的JWS保持访问令牌信息(以下称为“JWS访

问令牌")。

[0040] JWS访问令牌由JWS头、JWS有效载荷和JWS签名构成。接下来将描述在本实施例中使用的JWS访问令牌的组成元素。

[0041] 表格1

[0042]

	值	声明名称	声明详情
1	JWS 头	"alg"(Algorithm)	识别用于JWS签名的加密算法
2		"typ"(Type)	包含"JWT"以指示JWT格式
3		"kid"(Key ID)	指示哪个密钥曾用于保护JWS
4	有效载荷	"iss"(Issuer)	JWT 发布者的标识符
5		"sub"(Subject)	JWT 主体的标识符
6		"aud"(Audience)	预计使用JWT的主体的标识符列表
7		"exp"(Expiration Time)	JWT 的有效期
8		"nbf"(Not Before)	JWT 生效的日期/时间
9		"iat"(Issued At)	JWT 发布的时间

[0043]

[0044] 表1:包括在JWS访问令牌中的声明

[0045] 表1中具有值"JWS头"的声明是JWS RFC7515中定义的"注册声明",如下所述。注意,"声明"是主体主张的信息,并且表示为由声明名称和声明值构成的名称/值对。换句话说,"JWS头"中的声明包括识别JWS签名中使用的加密算法的"alg"(算法)、其中写入"JWT"以指示JWT格式的"typ"(类型)以及指示哪个密钥曾被用来保护JWS的"kid"(Key ID)。

[0046] 此外,表1中的声明是在JWT RFC7519中定义的"注册声明",如下所述。"iss"(Issuer)是JWT发布人的标识符;"sub"(Subject)是JWT主体的标识符;"aud"(Audience)是预计使用JWT的主体的标识符列表;"exp"(Expiration time)是JWT的有效期;"nbf"(Not Before)是JWT生效的日期/时间;以及"iat"(Issued At)是JWT的唯一标识符。在上述"exp"、"nbt"和"iat"中指定的日期/时间是表示从1970-01-01T0:0:0Z UTC到指定的UTC日期/时间的秒数的JSON数值。尽管可以根据需要使用"注册声明",但是根据本实施例,JWS头"kid"和JWS有效载荷"iat"被设置为授权服务器中的必需项目。

[0047] 根据本实施例,在发布具有诸如表1中所示的内容的JWS访问令牌的授权服务器

中,根据作为JWT规范的RFC7519将表1中的声明编码为JSON对象。根据作为JWS规范(表1中的声明的JSON表达,即JWS有效载荷)的RFC7515中的紧凑序列化规范进行数字签名的内容被表达并被编码为紧凑的URL-安全字符串。根据本实施例的JWS访问令牌是根据JWS紧凑序列化规范的字符串,其中,编码的JWS头、编码的JWS有效载荷和编码的JWS签名按照这样的顺序使用句点(.)作为分隔符相连接。以下是连接JWS头(“头”部分)、JWS有效载荷(“有效载荷”部分)和编码的JWS头、编码的JWS有效负载以及编码的JWS签名(“编码”部分)的结果的示例。

[0048] 头

[0049] {

[0050] “alg”:“RS256”,

[0051] “typ”:“JWT”,

[0052] “kid”:“3e4aed8ee5914d9e82ca0ae762fc84ef”

[0053] }

[0054] 有效载荷

[0055] {

[0056] “iss”:“https://auth.example.com”,

[0057] “sub”:“1ce42f74-1225-4cbb-b23f-f525dabc3cfd”,

[0058] “aud”:“[https://print.srv.example.com https://form.srv.example.com]”,

[0059] “exp”:1472713413,

[0060] “nbf”:1472709813,

[0061] “iat”:1472709813

[0062] }

[0063] 编码

[0064] eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9...(base64编码头).eyJpc3MiOiJjbGllbnQwMDEiLCJzdWIiOiJodHRwczovL3h4eC5jb20v IiwiYXVkIjoiaHR0cHM6Ly94eHguY29tL2F1dGhyaXphdGlvbiIsImV4cCI6MTQ3MjcxMDQxMywiaWF0IjoxNDcyNzAwNDEzfQ...(base64编码有效载荷).k_SbabNV...(签名)

[0065] 认证/授权服务器的功能块

[0066] 图4是示出根据本发明的实施例的认证/授权系统的功能框图。这些功能块在认证/授权服务器201中操作。各个块由例如认证/授权服务器201的CPU执行存储在存储器中的程序来实现。客户端管理单元401管理客户端信息。资源服务器信息管理单元402管理与和认证/授权服务器201连接的资源服务器202有关的信息。私钥/公钥对管理单元403管理用于对JWS访问令牌进行签名的密钥。响应于来自资源服务器202的请求,私钥/公钥对管理单元403将所需公钥传递到资源服务器202。表2是由私钥/公钥对管理单元403管理的私钥/公钥对管理表。

[0067] 表2

[0068]

	Key ID	私钥	公钥
1	3e4aed8ee5914d9e82ca0ae762fc84ef	xxx	xxx
2	f2bcf072e23142b0909163cadf0c2347	yyy	yyy

3	60730fd46abf4d51908057f7ef059050	zzz	zzz

[0069] 表2:密钥/公钥对管理表

[0070] 表2中的私钥/公钥对管理表由表示用于唯一识别私钥/公钥对的Key ID的Key ID项、表示私钥的细节的私钥项以及表示公钥的细节的公钥项构成。在私钥项和公钥项中设置的值可以是实际文件或文件路径。

[0071] JWS访问令牌发布单元404发布JWS访问令牌,并且由发布的JWS访问令牌管理单元405管理所发布的JWS访问令牌的信息。JWS访问令牌验证单元406验证JWS访问令牌是否有效。注意,由JWS访问令牌验证单元504在资源服务器中执行JWS访问令牌验证,稍后将参考图5对其进行描述。

[0072] 资源服务器的功能块

[0073] 图5是示出根据本发明的实施例的认证/授权系统的功能框图。这些功能块在资源服务器202中操作。各个块由例如资源服务器202的CPU执行存储在存储器中的程序来实现。资源管理单元501管理资源信息,JWS访问令牌管理单元503管理JWS访问令牌信息,并且JWS访问令牌验证单元504验证JWS访问令牌。公钥管理单元502管理从认证/授权服务器201获得的公钥。公钥管理单元502还向认证/授权服务器201进行获取公钥的请求,并根据需要获得公钥。表3是由公钥管理单元502管理的公钥管理表。

[0074] 表3:

[0075]

	Key ID	公钥
1	3e4aed8ee5914d9e82ca0ae762fc84ef	xxx
2	f2bcf072e23142b0909163cadf0c2347	yyy
3	60730fd46abf4d51908057f7ef059050	zzz

[0076] 表3:公钥管理表

[0077] 表3中的公钥管理表是用于管理从认证/授权服务器201获得的公钥的表,并且由表示用于唯一识别公钥的Key ID的Key ID项和表示公钥的细节的公钥项构成。在私钥项和公钥项中设置的值可以是实际文件或文件路径。认证/授权服务器信息管理单元505管理与和资源服务器202连接的认证/授权服务器201有关的信息。

[0078] 签名访问令牌的发布和验证(第一至第四实施例)

[0079] 图6是与根据本发明实施例的认证/授权系统中的签名访问令牌的发布和验证有关的序列图。本实施例将描述使用根据OAuth2(RFC6749)的授权代码授权流程来发布和验证JWS访问令牌。然而,JWS访问令牌的发布和验证可以根据OAuth2(RFC6749)使用客户端凭证授权、资源所有者密码凭证授权、隐式授权等,而不是授权代码授权。

[0080] 在OAuth2协议中,在开始访问令牌发布等流程之前,在认证/授权服务器201中注册客户端。用于在认证/授权服务器201中注册客户端203的方法在与普通终端用户(未示出)的对话之后使用HTML注册表格。本实施例假设所需客户端是预先注册的。

[0081] 根据本实施例的图6详细示出了在认证/授权服务器201验证资源所有者并注册授权代码之后执行的流程,并且客户端203根据OAuth2(RFC6749)的授权代码流程获得授权代码(未示出)。

[0082] 在步骤S601中，资源服务器202向认证/授权服务器201发出公钥获取请求，并获取由认证/授权服务器201保持的所有公钥。在本实施例中，在用于重置资源服务器202的处理期间执行一次该处理。

[0083] 在步骤S602中，资源服务器202保存从认证/授权服务器201获得公钥的日期/时间。获取的日期/时间也在下文将描述的根据第二实施例和第四实施例的JWS访问令牌验证流程中使用。这样，在本实施例、第三实施例等中不一定要保存获得的日期/时间。

[0084] 在步骤S603中，认证/授权服务器201将新的私钥/公钥对添加到私钥/公钥对管理单元403。当需要新的密钥对时执行该处理，例如当私钥已经从认证/授权服务器201的私钥/公钥对管理单元403泄漏时、正在使用的加密算法已被泄露时等。

[0085] 在步骤S604中，客户端203根据OAuth协议规范请求认证/授权服务器201发布指示授权代码和各种类型的凭证的访问令牌。

[0086] 在步骤S605中，认证/授权服务器201响应于该请求而发布JWS访问令牌。

[0087] 在步骤S606中，认证/授权服务器201的JWS访问令牌发布单元404将步骤S605中发布的JWS访问令牌返回给客户端203。

[0088] 在步骤S607中，客户端203使用先前获得的JWS访问令牌来请求资源服务器202提供服务。具体地，客户端203在HTTP认证头中设置JWS访问令牌，并在调用资源请求REST API等时使用JWS访问令牌。

[0089] 在步骤S608中，资源服务器202验证从客户端203接收的JWS访问令牌。稍后将利用图7中的流程图描述根据本实施例的JWS访问令牌验证。

[0090] 如果步骤S608中的验证结果指示JWS访问令牌有效，则在步骤S609中，资源服务器202执行用于提供服务的处理。但是，如果JWS访问令牌无效，则不会执行提供服务的处理。

[0091] 如果已经执行了步骤S609的处理，则在步骤S610中，资源服务器202将处理结果返回给客户端203。然而，如果在步骤S608中验证失败，则资源服务器202将错误返回给客户端203。然后，与图6所示的认证/授权系统中的签名访问令牌的发布和验证有关的序列结束。

[0092] 资源服务器访问令牌验证过程(No.1)

[0093] 图7是示出根据本发明的实施例由资源服务器202的JWS访问令牌验证单元504执行的JWS访问令牌验证处理的流程图。图7示出了资源服务器202在图6的步骤S608中的处理。

[0094] 在步骤S701中，资源服务器202的JWS访问令牌验证单元504从接收自客户端203的服务请求中获取JWS访问令牌。包括在所获取的JWS访问令牌的头参数中的密钥ID(KeyID)被获取，并被用于验证中。

[0095] 在步骤S702中，JWS访问令牌验证单元504向公钥管理单元502查询是否存在与步骤S701中获得的KeyID匹配的密钥。如果不存在与获得的KeyID匹配的密钥，则处理进入步骤S703，而如果存在匹配密钥，则处理进入步骤S705。

[0096] 在步骤S703中，公钥管理单元502向认证/授权服务器201发出用于获取未被资源服务器202保持的公钥的请求。公钥获取请求包括当前被资源服务器202保持的公钥的信息，且可以仅获得与认证/授权服务器201保持的密钥的差异，或者可以获取和更新认证/授权服务器201保持的所有密钥。

[0097] 在步骤S704中，JWS访问令牌验证单元504再次向公钥管理单元502查询是否存在

与步骤S701中获得的KeyID匹配的密钥。如果存在与获得的KeyID匹配的密钥，则处理进入步骤S705，而如果不存在匹配密钥，则处理进入步骤S707。

[0098] 在步骤S705中，JWS访问令牌验证单元504基于所获得的JWS访问令牌中的签名来验证访问令牌是否有效。这里，通过在步骤S704中使用与KeyID匹配的公钥确认签名，可以确认访问令牌已经由有效认证/授权服务器发布。

[0099] 在步骤S706中，确定JWS访问令牌是否是有效访问令牌。如果确定访问令牌是有效访问令牌，则流程结束，之后处理返回到图6中所示的序列，并执行步骤S609的处理。如果确定访问令牌无效，则处理进入步骤S707。

[0100] 在步骤S707中，资源服务器202确定访问令牌无效，并将错误返回给客户端203。然后，与图6所示的认证/授权系统中的签名访问令牌的发布和验证有关的序列以及如图7所示的访问令牌验证流程结束。换句话说，步骤S707对应于验证失败时图6中的步骤S610。

[0101] 这样，通过已经接收到资源请求API调用的资源服务器202的JWS访问令牌管理单元503和JWS访问令牌验证单元504执行的JWS访问令牌验证处理和资源服务器202的验证处理结束。

[0102] 根据如上所述的本发明的JWS访问令牌验证提供了以下效果。添加到签名令牌(JWS访问令牌)的信息包括唯一指示用于签名的密钥的ID(KeyID)。根据本发明的方法，用此来验证资源服务器中的签名并向授权服务器查询公钥，使得资源服务器甚至可以针对由未被资源服务器保持的密钥签署的JWS访问令牌验证签名。

[0103] 第二实施例

[0104] 在上述第一实施例中，在图7中所示的JWS访问令牌验证流程中，每当接收到用资源服务器202未保持的KeyID签名的JWS访问令牌时，就对认证/授权服务器201进行查询。第二实施例将描述基于资源服务器202查询认证/授权服务器201的时间来减少对认证/授权服务器201作出的查询的数量。

[0105] 表4是用于在步骤S602中管理资源服务器202已向认证/授权服务器201查询公钥的时间的公钥查询时间管理表。

[0106] 表4

[0107]

	日期	Key IDs
1	2018/6/15 9:07:34.325	60730fd46abf4d51908057f7ef059050, f2bcf072e23142b0909163cadf0c2347, 3e4aed8ee5914d9e82ca0ae762fc84ef

[0108] 表4:公钥查询时间管理表

[0109] 表4中的公钥查询时间管理表由指示查询认证/授权服务器201的时间(日期/时间)的日期项以及指示在查询时获得的公钥列表的Key ID项构成。使用仅与最新查询有关的信息来管理查询时间就已足够，因此当已经进行新的查询时可以丢弃旧的查询信息。根据系统配置，还可以仅管理查询时间。

[0110] 在第一实施例中，资源服务器202使用图7中所示的序列在步骤S608中验证从客户

端203接收的JWS访问令牌。接下来第二实施例将利用图8中的流程图描述JWS访问令牌验证。

[0111] 图8是示出根据本发明第二实施例的资源服务器202的JWS访问令牌验证单元504执行的JWS访问令牌验证处理的流程图。

[0112] 在步骤S801中，资源服务器202的JWS访问令牌验证单元504从接受自客户端203的服务请求中获取JWS访问令牌。包括在获得的JWS访问令牌中的头参数中的KeyID被获取并用于验证。

[0113] 在步骤S802中，JWS访问令牌验证单元504向公钥管理单元502查询是否存在与在步骤S801中获得的KeyID匹配的密钥。如果不存在与获得的KeyID匹配的密钥，则处理进入步骤S803，如果存在匹配密钥，则处理进入步骤S807。

[0114] 在步骤S803中，JWS访问令牌验证单元504从所接收的JWS访问令牌中获得发布访问令牌的时间，并将该时间与由表4中的公钥查询时间管理表管理的查询时间进行比较。保持在公钥查询时间管理表中的查询时间具有在步骤S602中保持的公钥获得日期/时间作为初始值。如果发布JWS访问令牌的时间晚于公钥查询时间管理表中保持的查询时间，则处理进入步骤S804，而如果早于则进入步骤S809。如果发出访问令牌的时间早于对认证/授权服务器作出的密钥的最后查询时间，则可以确定即使进行另一个查询也无法获取能验证访问令牌的公钥。在这种情况下，处理进入步骤S809，返回错误。如果两个时间相同，则考虑到传输延迟可以确定发布访问令牌的时间靠后。换句话说，在这种情况下，即使两个时间相同，也可以确定发布访问令牌的时间是在最后查询时间之后。

[0115] 在步骤S804中，公钥管理单元502向认证/授权服务器201请求获取未由资源服务器202保持的公钥。公钥获取请求包括当前由资源服务器202保持的公钥的信息，可以仅获得与认证/授权服务器201保持的密钥的差异，或者可以获取和更新认证/授权服务器201保持的所有密钥。

[0116] 在步骤S805中，资源服务器202将对认证/授权服务器201进行的公钥的查询时间与密钥的ID(KeyID)相关联地保持在表4中的公钥查询时间管理表中。在该点之前保持的查询时间可以被丢弃，仅保持最新的查询时间。这里保持的最新查询时间用于步骤S803中进行的比较。

[0117] 在步骤S806中，JWS访问令牌验证单元504再次向公钥管理单元502查询是否存在与步骤S801中获得的KeyID匹配的密钥。如果存在与获得的KeyID匹配的密钥，则处理进入步骤S807，如果不存在匹配密钥，则处理进入步骤S809。

[0118] 在步骤S807中，JWS访问令牌验证单元504基于所获得的JWS访问令牌中的签名来验证访问令牌是否有效。通过使用步骤S806中与KeyID匹配的公钥确认签名，可以确认访问令牌已由有效认证/授权服务器发布。

[0119] 在步骤S808中，确定JWS访问令牌是否是有效的访问令牌。如果确定访问令牌是有效访问令牌，则流程结束，之后处理返回到图6所示的序列，然后执行步骤S609的处理。如果确定访问令牌无效，则处理进入步骤S809。

[0120] 在步骤S809中，资源服务器202确定访问令牌无效，并向客户端203返回错误。然后，与图6所示的认证/授权系统中的签名访问令牌的发布和验证有关的序列和如图8所示的访问令牌验证流程结束。换句话说，步骤S809对应于验证失败时图6中的步骤S610。

[0121] 如上所述的根据本实施例的JWS访问令牌验证提供以下效果。添加到签名令牌(JWS访问令牌)的信息包括认证/授权服务器发布JWS访问令牌的时间。根据本发明的方法，通过将资源服务器查询认证/授权服务器以获取公钥的时间与发布时间进行比较，可以抑制向认证/授权服务器进行不太可能产生能够验证访问令牌的密钥的查询。这样，即使已从客户端接收到由未保持在资源服务器中的密钥签署的JWS访问令牌时，仍可以减少从资源服务器向认证/授权服务器进行的查询数量，并提高服务器的性能。即使恶意第三方不断向资源服务器发送无效令牌，也可以抑制对授权服务器的不必要查询，从而减轻服务器上的负荷。

[0122] 在步骤S805中，可以仅存储查询时间而不保持KeyID。如果通过查询获得的KeyID已经与查询时间相关联地被保持，则即使已接收到利用具有相同KeyID的密钥签名的JWS访问令牌，也可以避免在上一次查询之后的设定时间段内进行查询。这使得可以减少对认证/授权服务器的查询次数。

[0123] 第三实施例

[0124] 在上文所述的第一实施例中，在图7所示的JWS访问令牌验证流程中，每当接收到利用资源服务器202未保持的KeyID签署的JWS访问令牌时，就对认证/授权服务器201进行查询。第三实施例将描述通过在认证/授权服务器生成JWS访问令牌时向令牌的一部分添加签名(例如向指示用来签署访问令牌的KeyID(“kid”JWS头声明)添加签名)来减少对授权服务器进行的查询的数量。

[0125] 第一实施例利用表1描述了包括在JWS访问令牌中的声明。在第三实施例中，添加了以下新的声明。

[0126] 表5

[0127]

	值	声明名称	声明细节
1	JWS 头	“kid_signature”	对“kid” (KeyID)这一字符串签名

[0128] 表5:添加到JWS访问令牌的声明

[0129] 表5中的声明是在本实施例中唯一设置的JWS头私有声明。JWS访问令牌发布单元404对JWS头“kid”声明设置在步骤S605中发布JWS访问令牌时用于对JWS访问令牌签名的密钥的KeyID。此时，JWS访问令牌发布单元404对设置为“kid”声明的字符串生成签名，并设置“kid_signature”声明。用于生成“kid”声明的签名的密钥被存储在私钥/公钥对管理单元403中，并且被保持以便与用于生成添加到JWS访问令牌的签名的密钥区分开。用于“kid”声明签名的密钥对的公钥与响应于来自步骤S601的资源服务器202的公钥获取请求而提供的公钥一起被从认证/授权服务器201传递。

[0130] 在本实施例中，定义JWS私有头(private header)以便在JWS访问令牌中包括KeyID的签名。但是，可以使用句点(.)将签名连接到指定KeyID的字符串，并对“kid”声明设置。

[0131] 在第一实施例中，在步骤S608中，资源服务器202通过图7所示的序列验证从客户

端203接收的JWS访问令牌。下面将使用图9中的流程图描述第三实施例的JWS访问令牌验证。

[0132] 图9是示出根据本发明第三实施例的资源服务器202的JWS访问令牌验证单元504执行的JWS访问令牌验证过程的流程图。

[0133] 在步骤S901中，资源服务器202的JWS访问令牌验证单元504从接收自客户端203的服务请求中获取JWS访问令牌。包括在获得的JWS访问令牌的头参数中的KeyID被获取，并用于验证。

[0134] 在步骤S902中，JWS访问令牌验证单元504向公钥管理单元502查询是否存在与步骤S801中获得的KeyID匹配的密钥。如果没有与所获得的KeyID匹配的密钥，则处理进入步骤S903，而如果存在匹配密钥，则处理进入步骤S907。

[0135] 在步骤S903中，JWS访问令牌验证单元504验证所获得的KeyID的签名，并验证KeyID已由有效授权服务器设置。具体地，JWS访问令牌验证单元504获得包括在JWS访问令牌的JWS头部中的“kid_signature”声明，并使用由公钥管理单元502保持的用以签署KeyID的公钥来验证签名。

[0136] 在步骤S904中，资源服务器202确定步骤S903中执行的签名验证是否指示KeyID是由有效授权服务器发出的。如果确定KeyID已由有效授权服务器发出，则处理进入步骤S905，而如果确定KeyID已由无效授权服务器发出，则处理进入步骤S909。

[0137] 在步骤S905中，公钥管理单元502向认证/授权服务器201请求获取未由资源服务器202保持的公钥。公钥获取请求包括当前由资源服务器202保持的公钥的信息，并且可以仅获得与认证/授权服务器201保持的密钥的差异，或者可以获得认证/授权服务器201保持的所有密钥，并将其更新为最新状态。

[0138] 在步骤S906中，JWS访问令牌验证单元504再次向公钥管理单元502查询是否存在与步骤S801中获得的KeyID匹配的密钥。如果存在与获得的KeyID匹配的密钥，则处理进入步骤S907，而如果不存在匹配密钥，则处理进入步骤S909。

[0139] 在步骤S907中，JWS访问令牌验证单元504基于所获得的JWS访问令牌中的签名来验证访问令牌是否有效。通过在步骤S906中使用与KeyID匹配的公钥确认签名，可以确认访问令牌已由有效认证/授权服务器发布。

[0140] 在步骤S908中，确定JWS访问令牌是否是有效的访问令牌。如果确定访问令牌是有效访问令牌，则流程结束，之后处理返回到图6所示的序列，执行步骤S609的处理。如果确定访问令牌无效，则处理进入步骤S909。

[0141] 在步骤S909中，资源服务器202确定访问令牌无效，并向客户端203返回错误。然后，与图6所示的认证/授权系统中的签名访问令牌的发布和验证有关的序列以及如图9所示的访问令牌验证流程结束。换句话说，步骤S909对应于验证失败时的图6中的步骤S610。

[0142] 如上所述，根据本实施例的JWS访问令牌验证提供以下效果。除了JWS访问令牌的签名之外，添加到签名令牌(JWS访问令牌)的信息还包括作为JWS头声明的“kid”(KeyID)的签名。根据本实施例的方法，当获得用资源服务器未保持的KeyID签名的JWS访问令牌时，通过验证KeyID的签名，可以减少对认证/授权服务器的查询次数。这样，即使已从客户端接收到由未保持在资源服务器中的密钥签署的JWS访问令牌时，仍可以减少从资源服务器向认证/授权服务器进行的查询次数，并提高服务器的性能。即使恶意第三方不断向资源服务器

发送无效令牌，也可以抑制对授权服务器的不必要查询，从而减轻服务器上的负荷。

[0143] 第四实施例

[0144] 第二实施例和第三实施例各自基于第一实施例描述了用于验证JWS访问令牌的流程，也可以组合这些方法。第四实施例将描述通过将根据第二实施例的使用发出令牌的时间的验证流程与根据第三实施例的使用签名的KeyID的验证流程相结合，从而减少对授权服务器的查询的次数。接下来将利用图10中所示的流程图给出具体描述。

[0145] 图10是示出由资源服务器202的JWS访问令牌验证单元504执行的、组合了根据本发明第二实施例和第三实施例的JWS访问令牌验证处理的流程图。因此，附加到先前附图的步骤的数字将按原样使用。图10中的验证流程进入步骤S901。处理以与图9所示的验证流程相同的方式从步骤S901进行到S904。如果步骤S904中执行的KeyID验证指示KeyID是由有效授权服务器发出的，则处理移至图8所示的流程中的步骤S803。从步骤S803开始的流程遵循图8中所示的验证流程。在步骤S902中，如果在JWS访问令牌中设置的KeyID与资源服务器所保持的公钥的KeyID匹配，则处理移至步骤S807，并验证JWS访问令牌。

[0146] 以这种方式组合第二实施例和第三实施例中描述的验证方法使得可以实现这些实施例的效果的组合。换句话说，即使对于具有由授权服务器发布的KeyID的JWS访问令牌，也可以根据发布令牌的时间来处理服务器查询。这使得可以更有效地减少对授权服务器进行的查询的数量。注意，步骤S803可以移动到步骤S902和S903之间。换句话说，根据本实施例，如果签名令牌早于获得最后一个密钥的时间，并且如果KeyID(即密钥识别信息)的验证是成功的，则无论进行这些确定的顺序如何，访问令牌都从授权服务器获得。

[0147] 其它实施例

[0148] 本发明的实施例还可以通过如下的方法来实现，即，通过网络或者各种存储介质将执行上述实施例的功能的软件(程序)提供给系统或装置，该系统或装置的计算机或是中央处理单元(CPU)、微处理单元(MPU)读出并执行程序的方法。

[0149] 虽然参照示例性实施例描述了本发明，但是应当理解，本发明并不限于所公开的示例性实施例。应当对下列声明的范围赋予最宽的解释，以使其涵盖所有这些变型例以及等同的结构及功能。

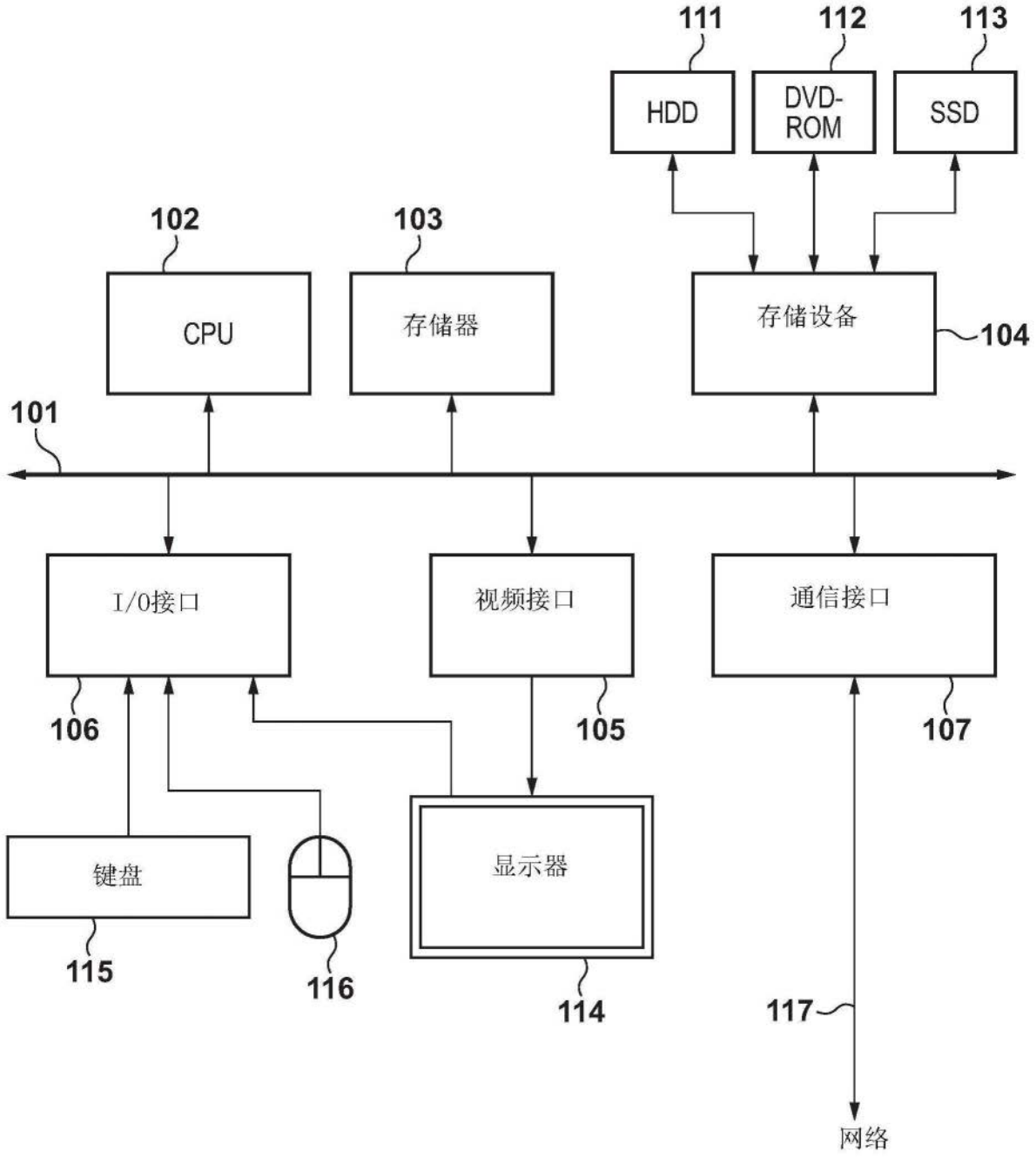

111
112
113
HDD
DVD-ROM
SSD
102
103
CPU
存储器
存储设备
104
101
I/O接口
视频接口
通信接口
106
105
107
键盘
显示器
115
116
114
117
网络

图1

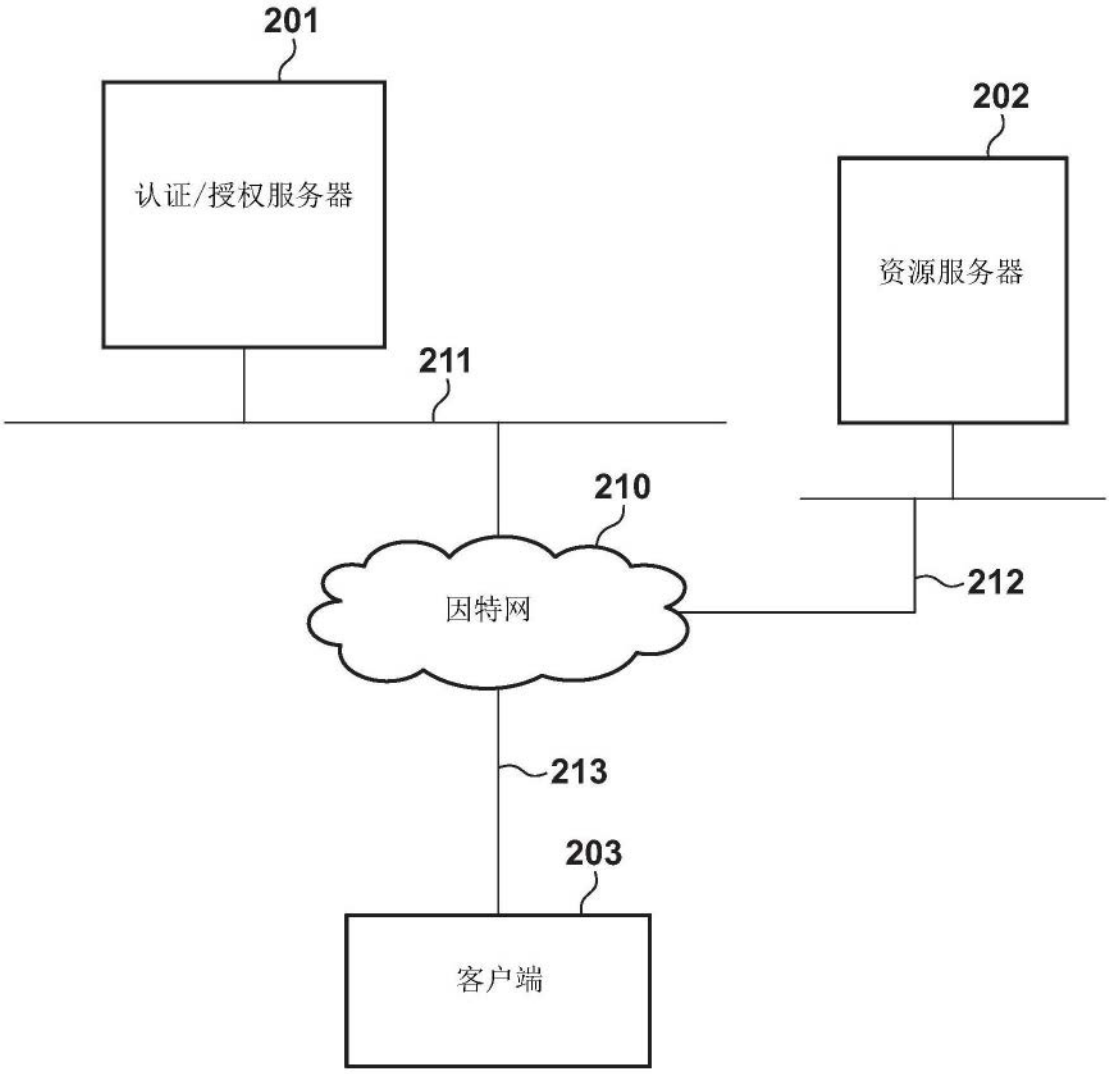


图2

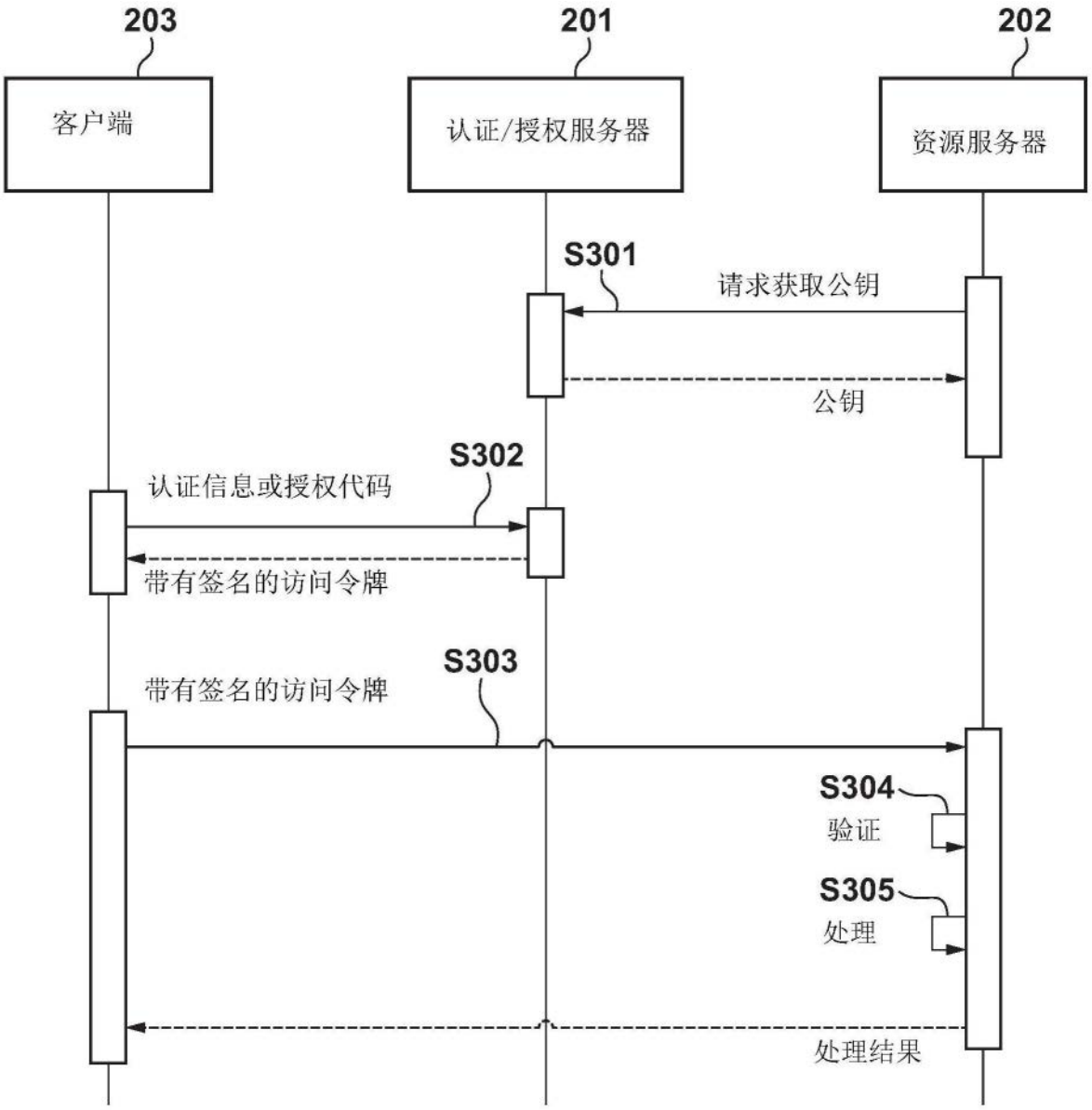

203
201
202
客户端
认证/授权服务器
资源服务器
S301
请求获取公钥
公钥
S302
认证信息或授权代码
带有签名的访问令牌
S303
带有签名的访问令牌
S304
验证
S305
处理
处理结果

图3

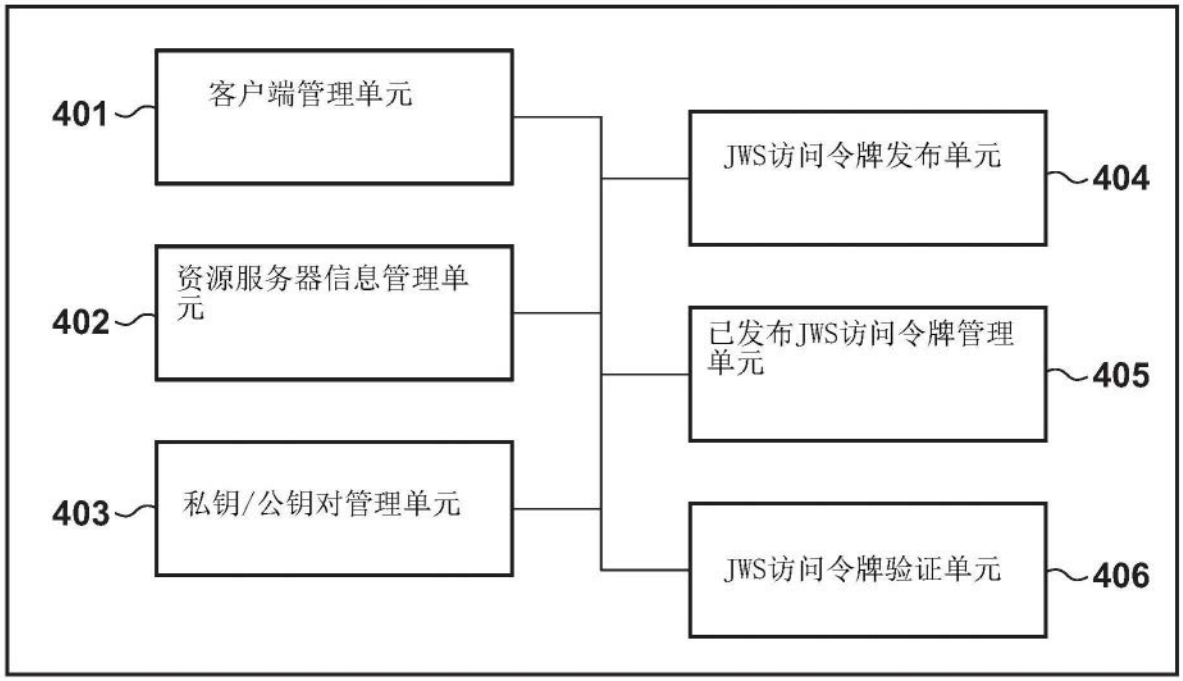

401
客户端管理单元
402
资源服务器信息管理单元
403
私钥/公钥对管理单元
JWS访问令牌发布单元
404
已发布JWS访问令牌管理单元
405
JWS访问令牌验证单元
406

图4

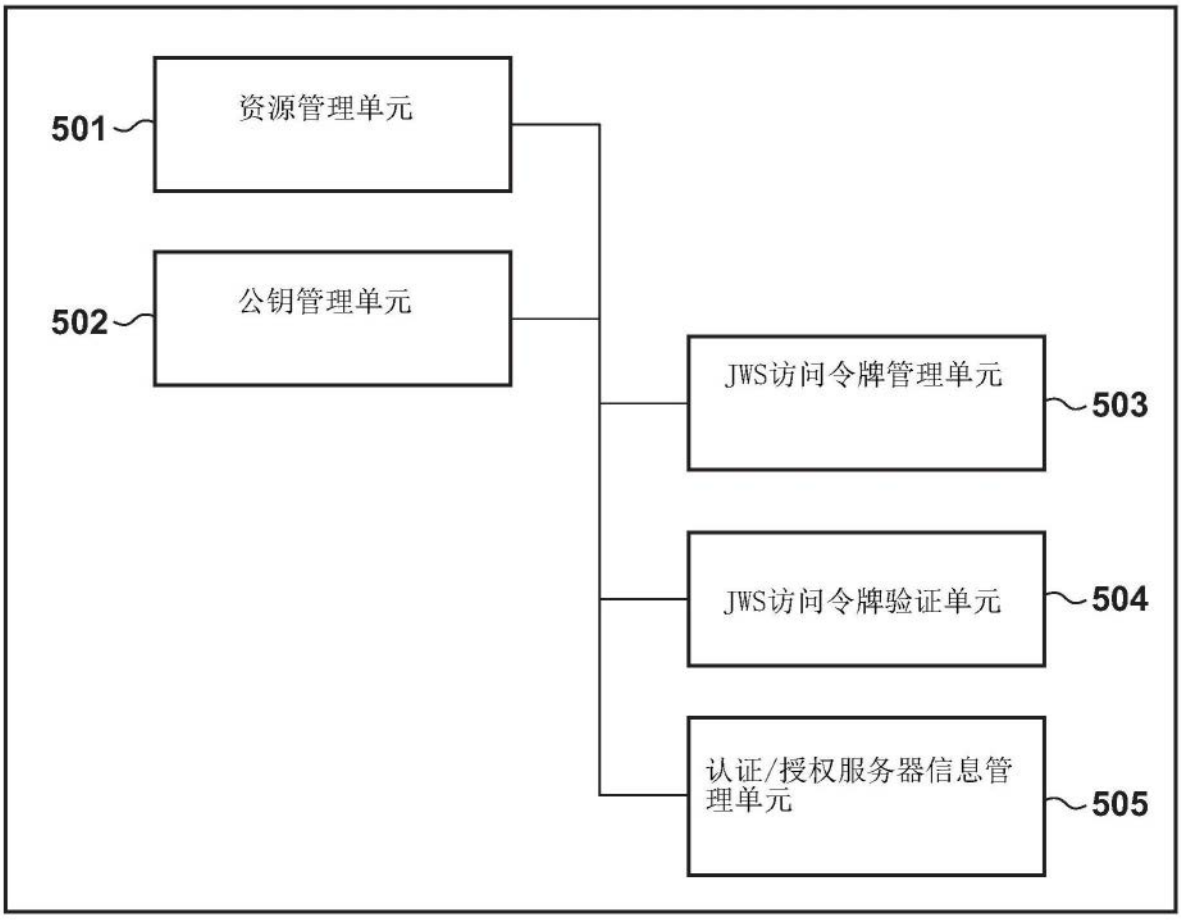

501
资源管理单元
502
公钥管理单元
JWS访问令牌管理单元
503
JWS访问令牌验证单元
504
认证/授权服务器信息管理单元
505

图5

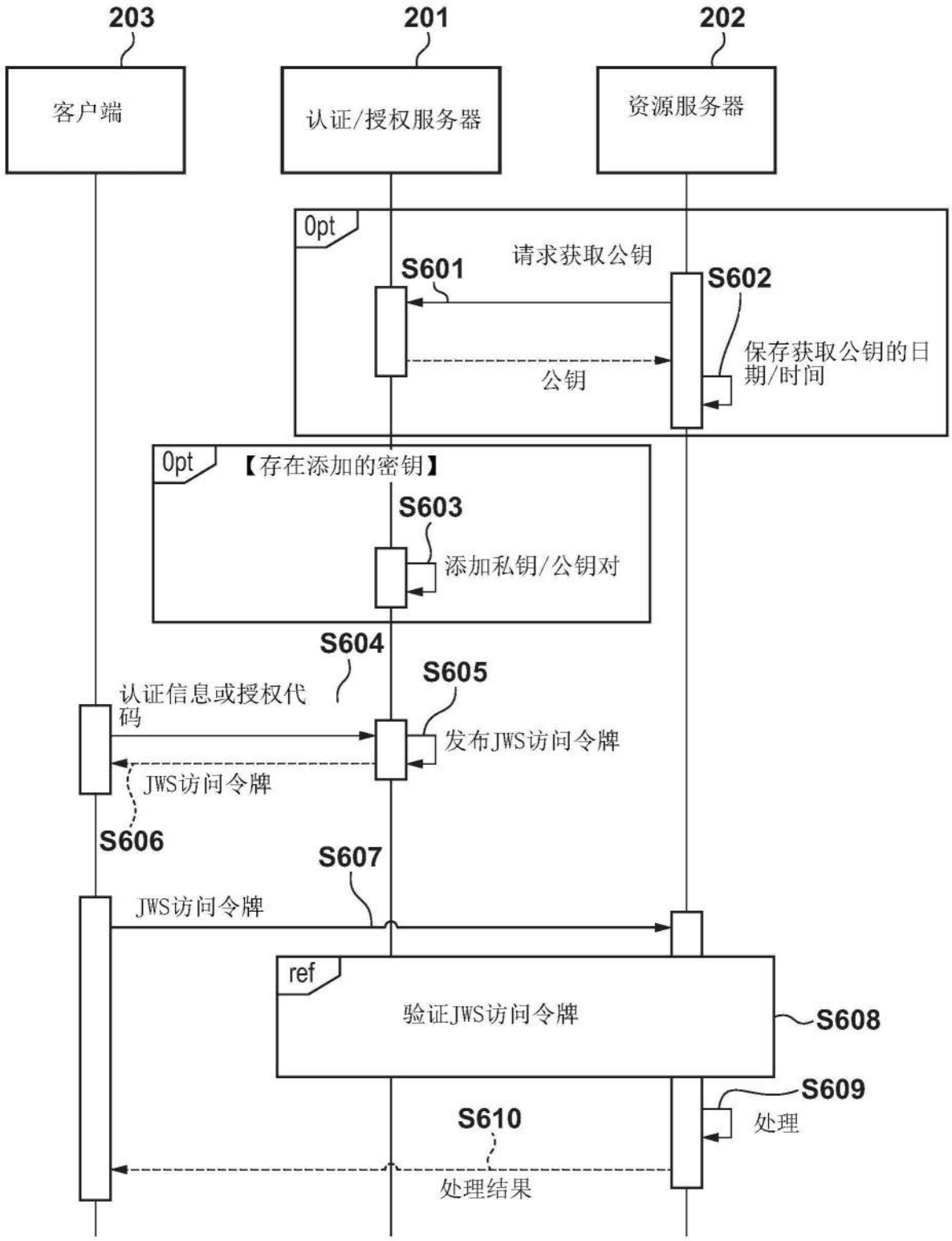

203
201
202
客户端
认证/授权服务器
资源服务器
Opt
请求获取公钥
S601
S602
保存获取公钥的日期/时间
公钥
Opt
【存在添加的密钥】
S603
添加私钥/公钥对
S604
S605
认证信息或授权代码
发布JWS访问令牌
JWS访问令牌
S606
S607
JWS访问令牌
ref
验证JWS访问令牌
S608
S609
处理
S610
处理结果

图6

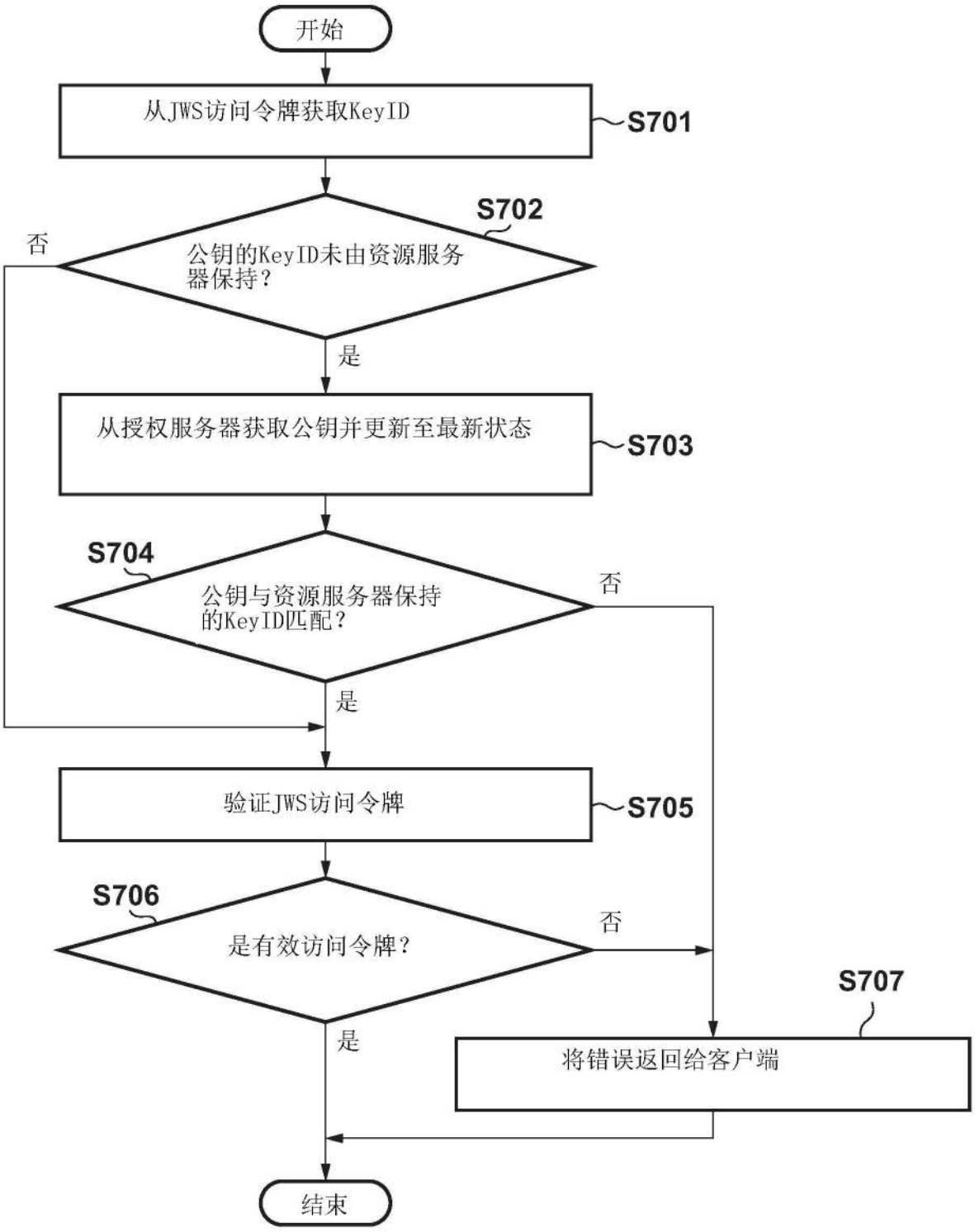

开始
从JWS访问令牌获取KeyID
S701
S702
公钥的KeyID未由资源服务器保持？
否
是
从授权服务器获取公钥并更新至最新状态
S703
S704
公钥与资源服务器保持的KeyID匹配？
否
是
验证JWS访问令牌
S705
S706
是有效访问令牌？
否
是
S707
将错误返回给客户端
结束

图7

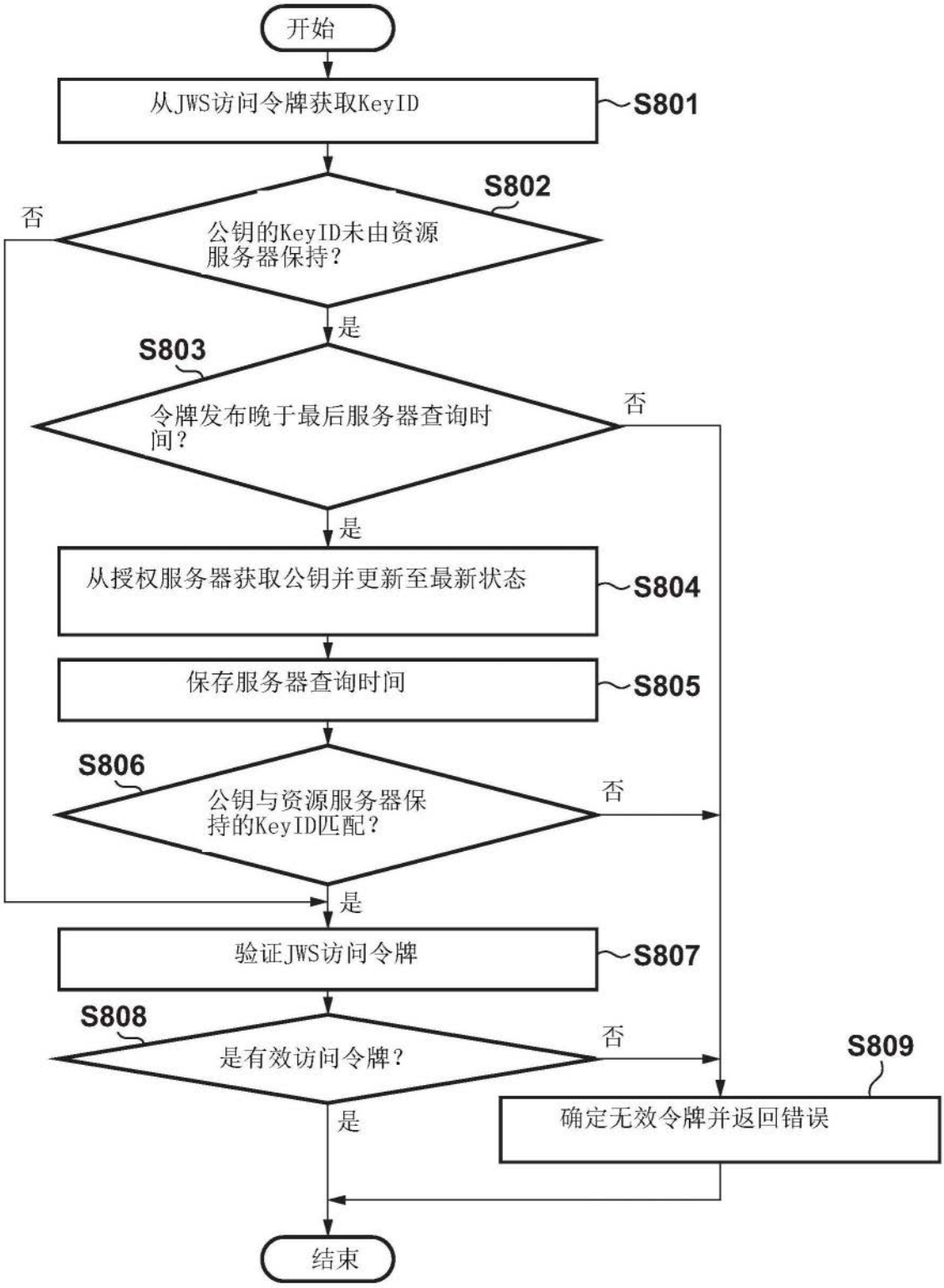

开始
从JWS访问令牌获取KeyID
S801
S802
公钥的KeyID未由资源服务器保持?
否
是
S803
令牌发布晚于最后服务器查询时间?
否
是
从授权服务器获取公钥并更新至最新状态
S804
保存服务器查询时间
S805
S806
公钥与资源服务器保持的KeyID匹配?
否
是
验证JWS访问令牌
S807
S808
是有效访问令牌?
否
是
S809
确定无效令牌并返回错误
结束

图8

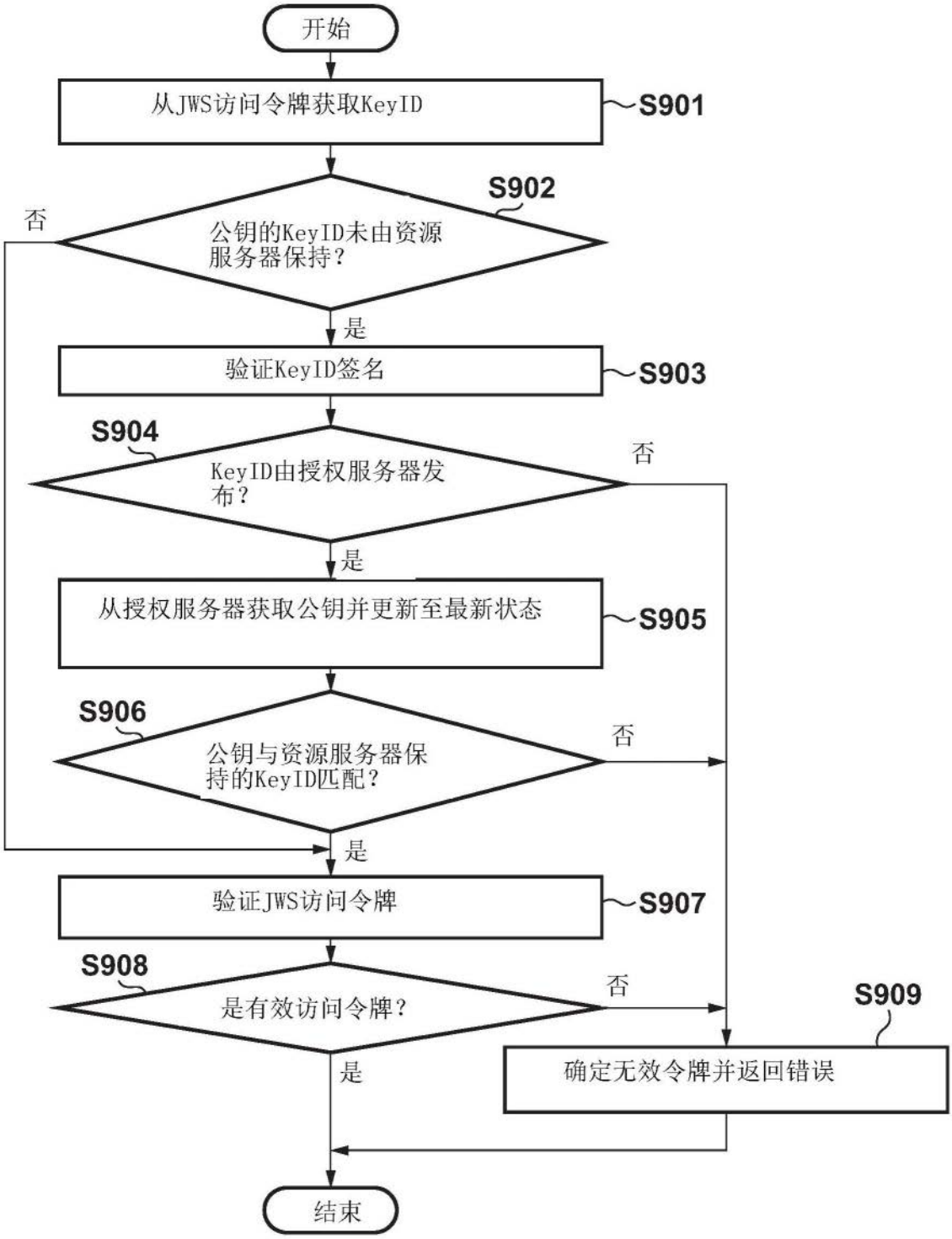

开始
从JWS访问令牌获取KeyID
S901
S902
公钥的KeyID未由资源服务器保持？
否
是
验证KeyID签名
S903
S904
KeyID由授权服务器发布？
否
是
从授权服务器获取公钥并更新至最新状态
S905
S906
公钥与资源服务器保持的KeyID匹配？
否
是
验证JWS访问令牌
S907
S908
是有效访问令牌？
否
是
S909
确定无效令牌并返回错误
结束

图9

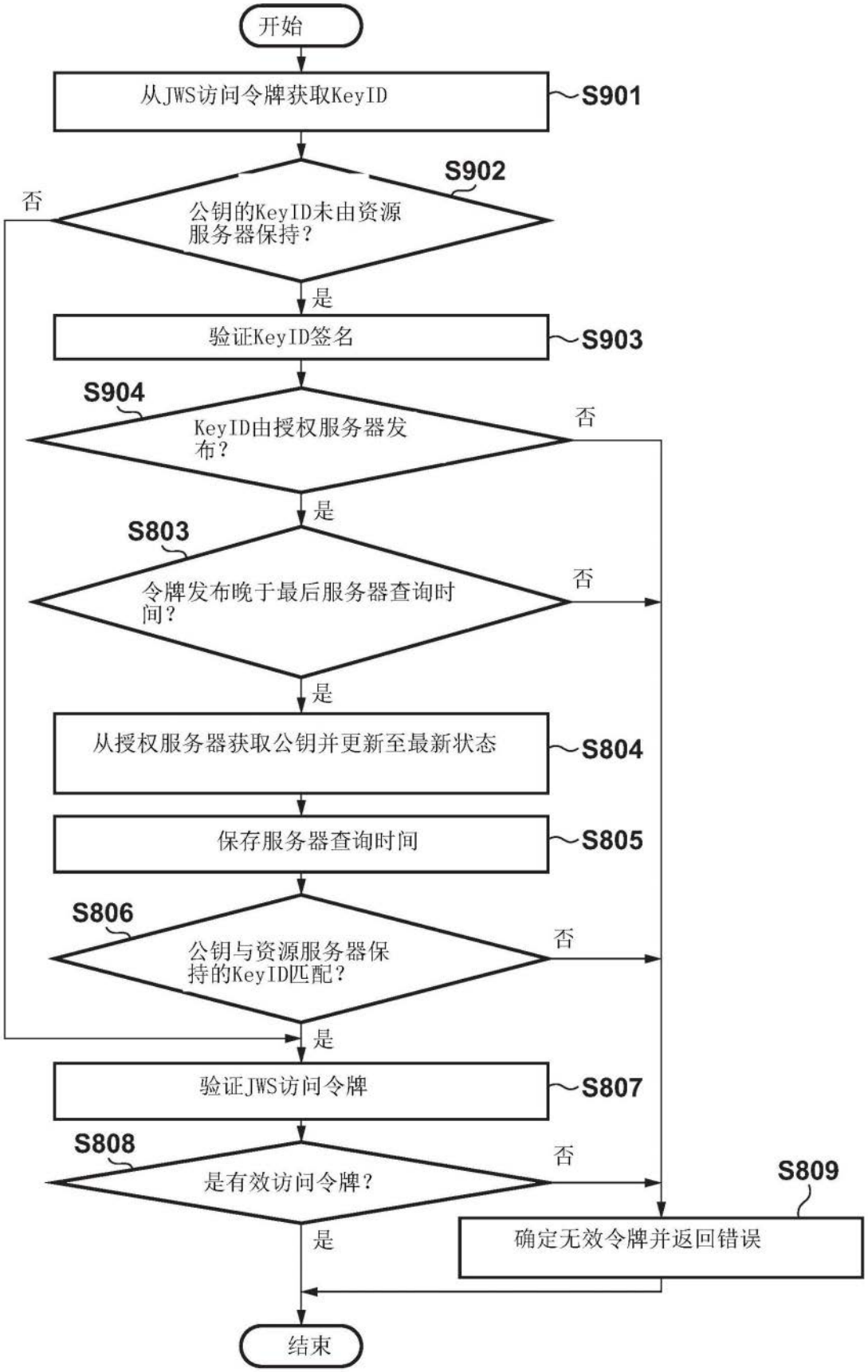

开始
从JWS访问令牌获取KeyID
S901
S902
公钥的KeyID未由资源服务器保持？
否
是
验证KeyID签名
S903
S904
KeyID由授权服务器发布？
否
是
S803
令牌发布晚于最后服务器查询时间？
否
是
从授权服务器获取公钥并更新至最新状态
S804
保存服务器查询时间
S805
S806
公钥与资源服务器保持的KeyID匹配？
否
是
验证JWS访问令牌
S807
S808
是有效访问令牌？
否
是
S809
确定无效令牌并返回错误
结束

图10