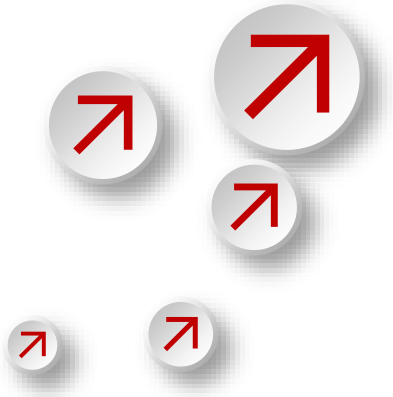
最简向导



https://medium.com/@darutk/the-simplest-guide-to-oauth-2-0-8c71bd9a15bb

Takahiko Kawasaki（川崎高彦）

首先

有用户的数据

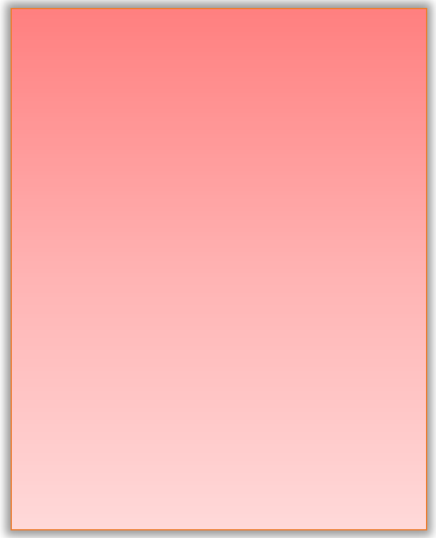
用户数据

有个**资源服务器**

负责管理用户数据

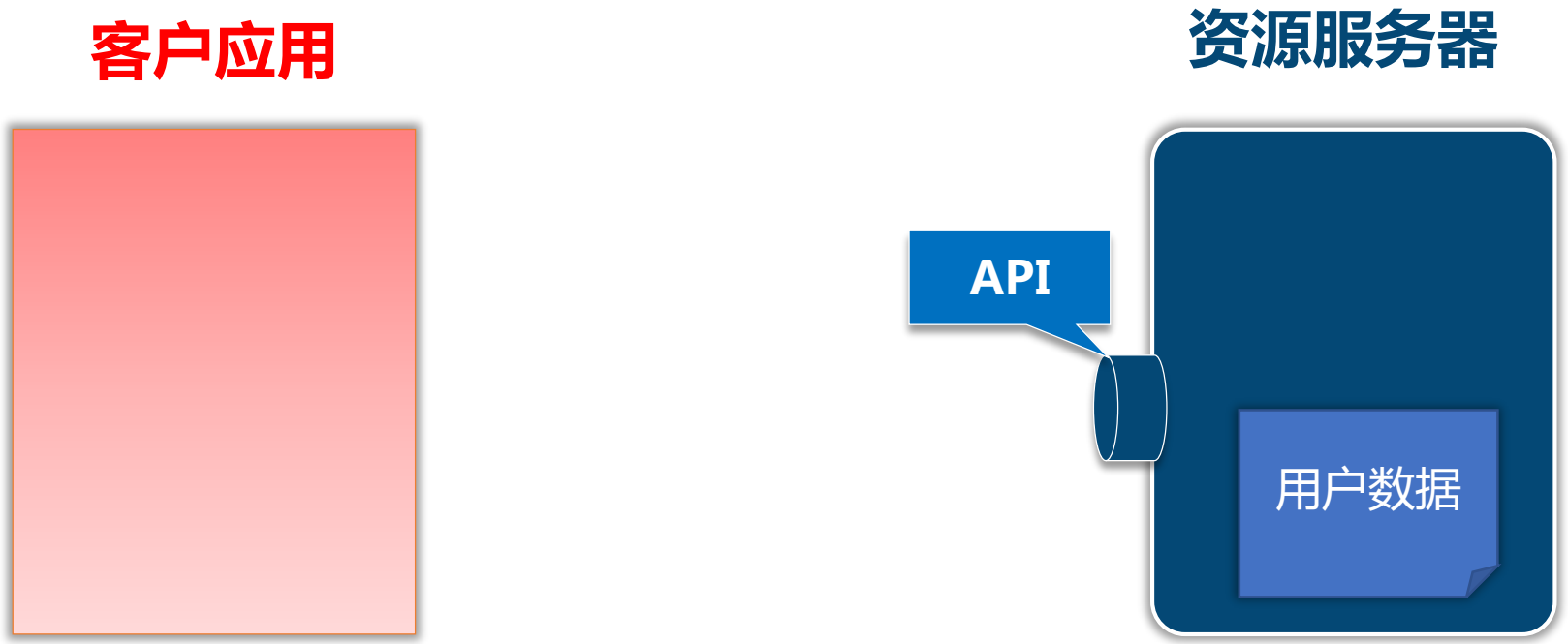
有个客户应用需要访问用户的数据

客户应用

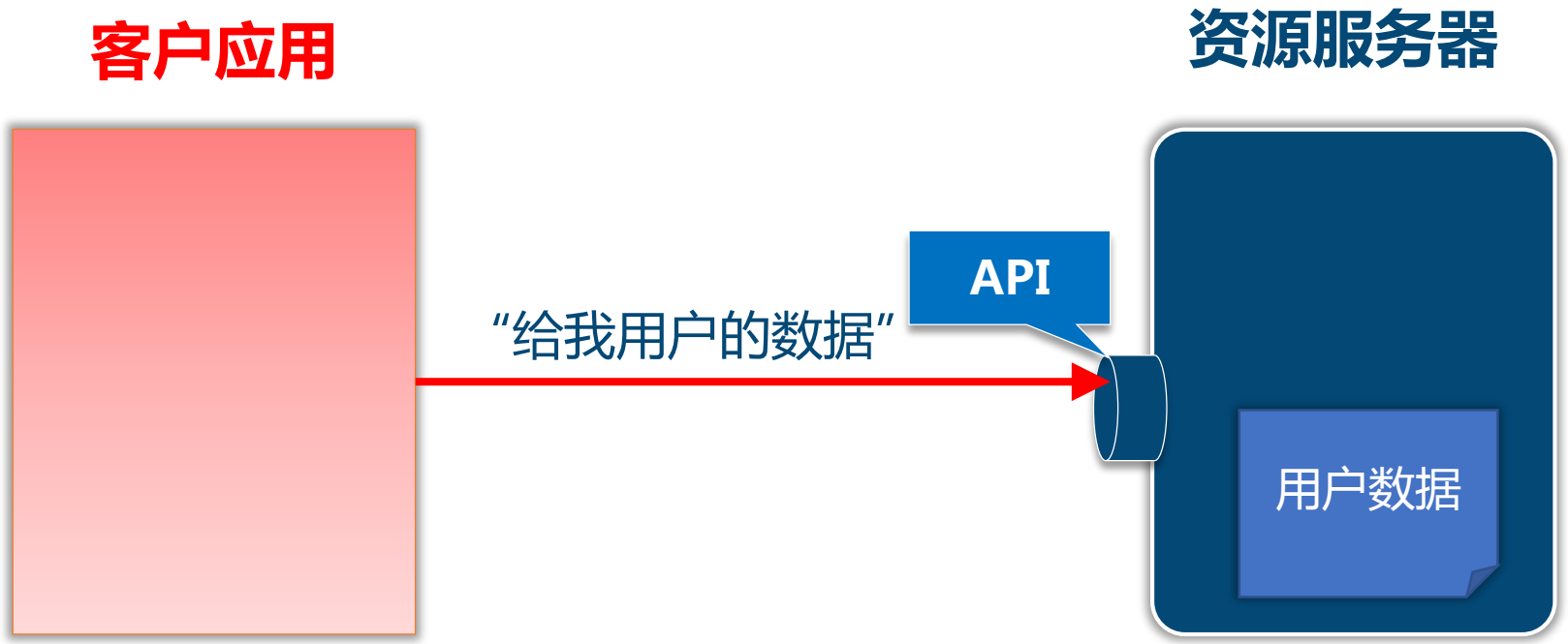


资源服务器

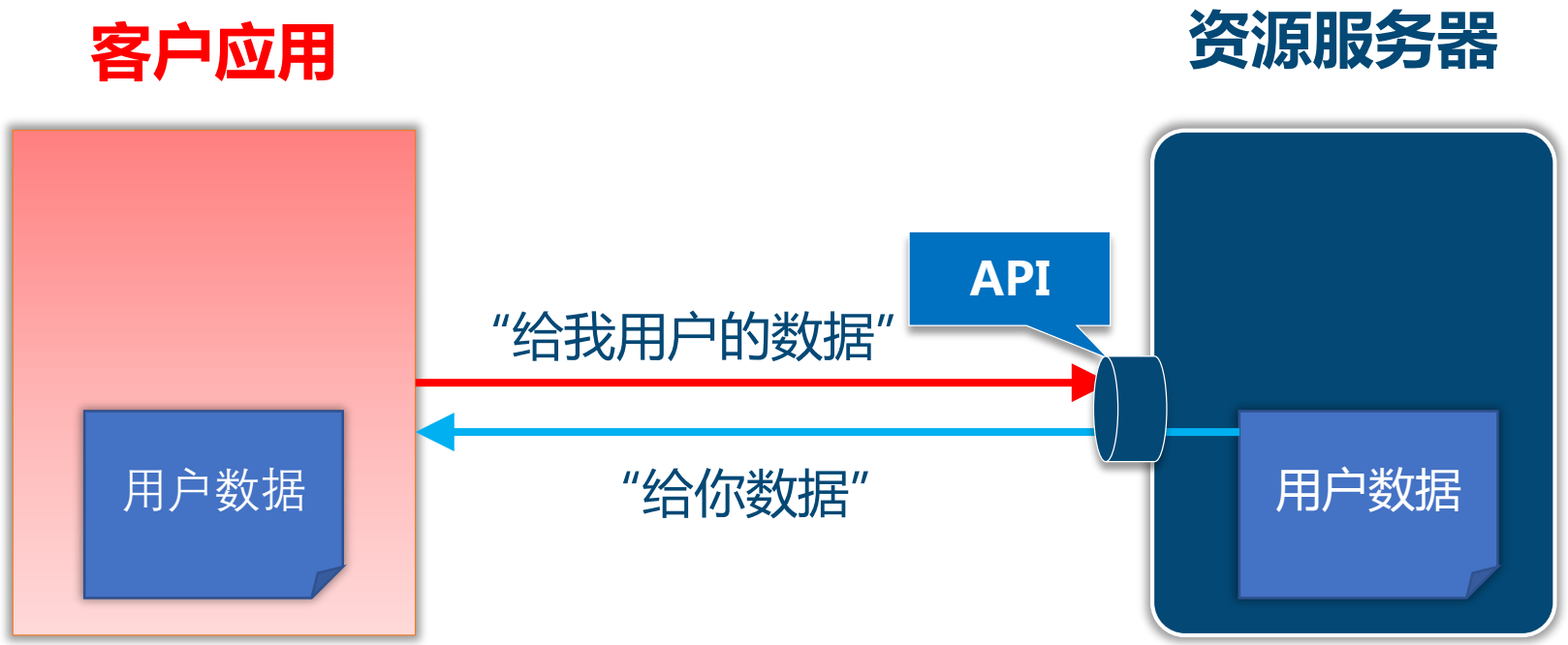
给资源服务器按个门暴露用户数据称为API



客户应用可以通过API访问用户数据

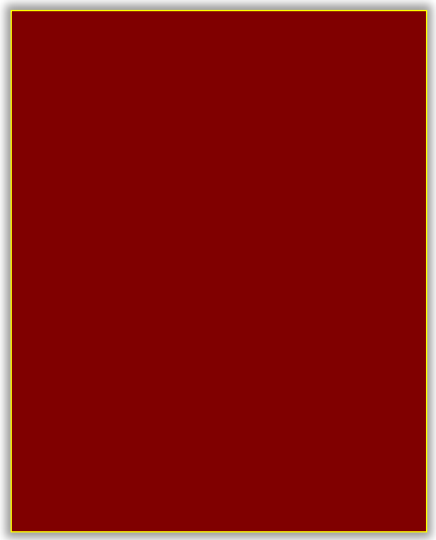


资源服务器返回用户数据

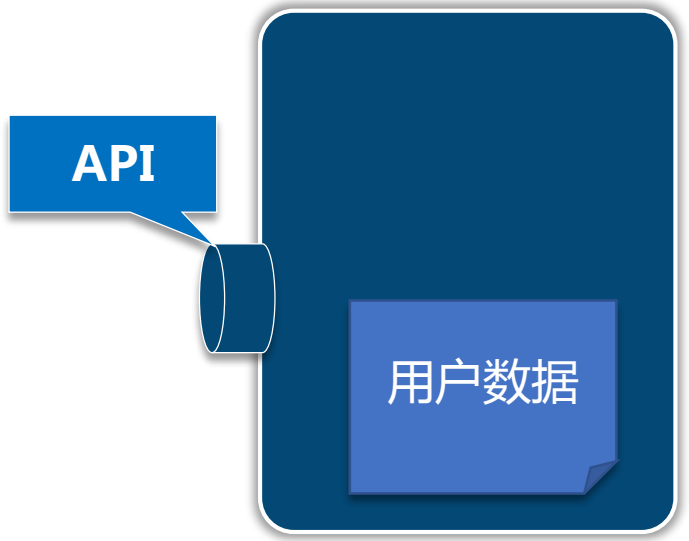


如果来了个恶意客户应用怎么办

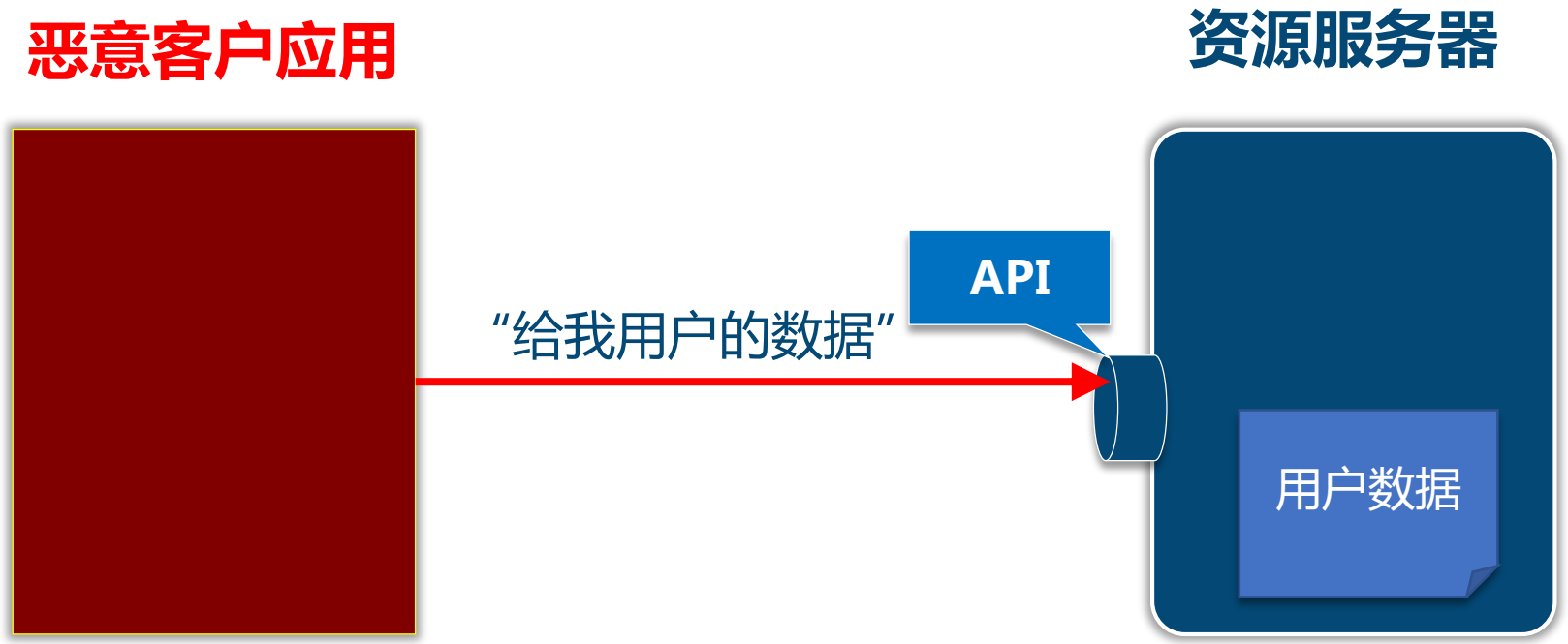
恶意客户应用



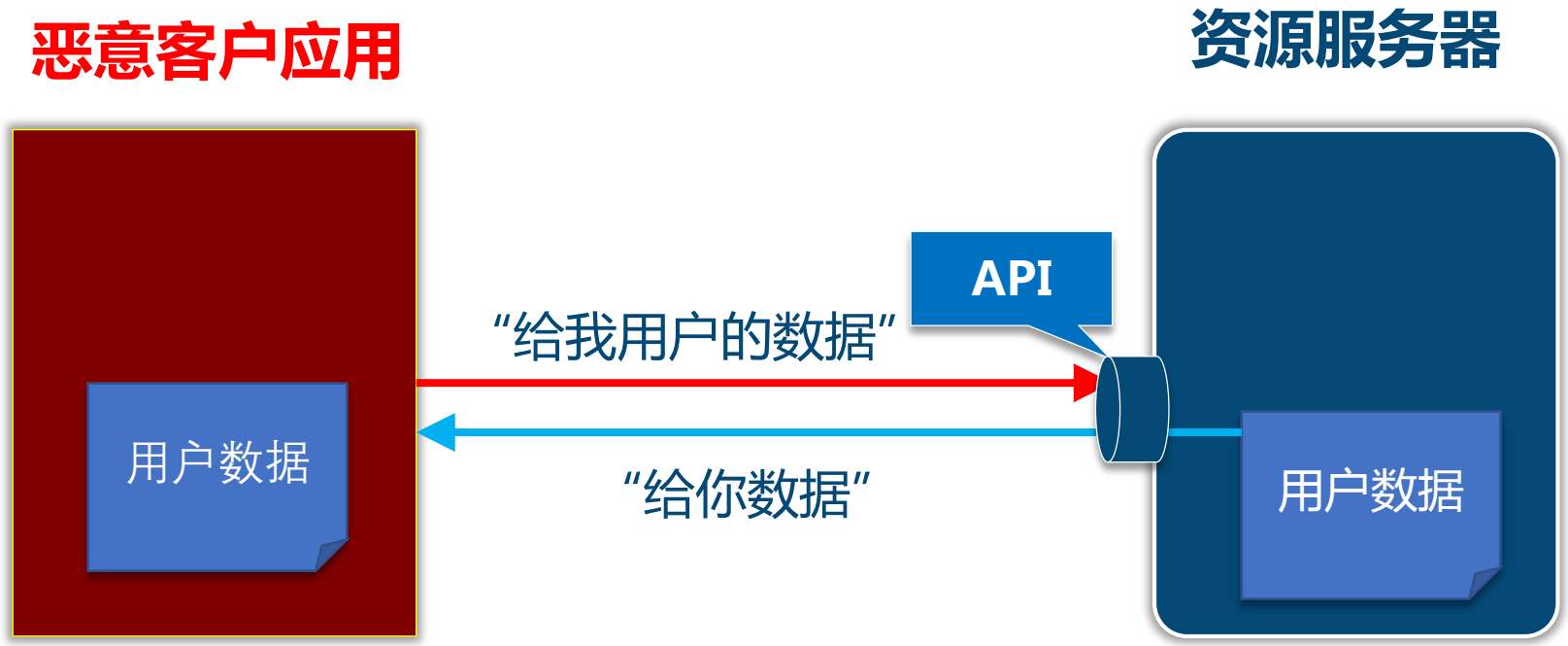
资源服务器



即使恶意客户应用要求访问用户数据

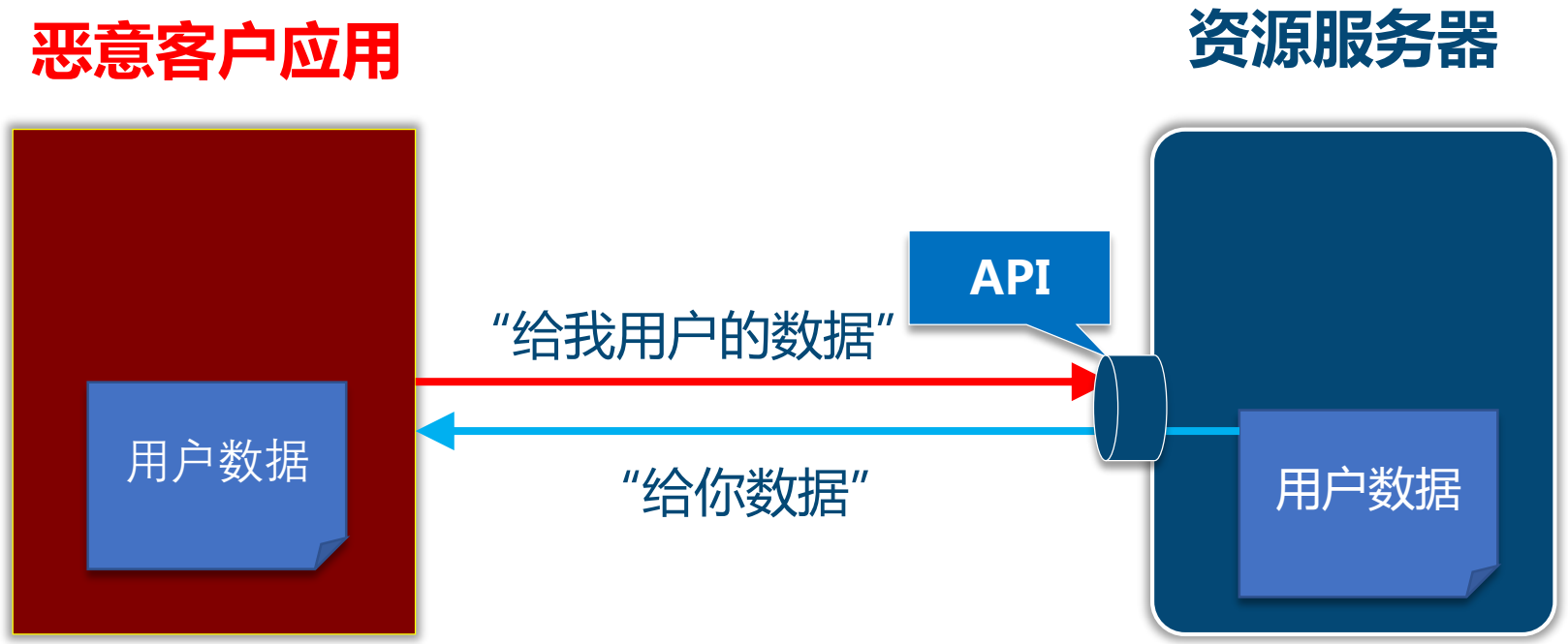


资源服务器还是返回用户数据



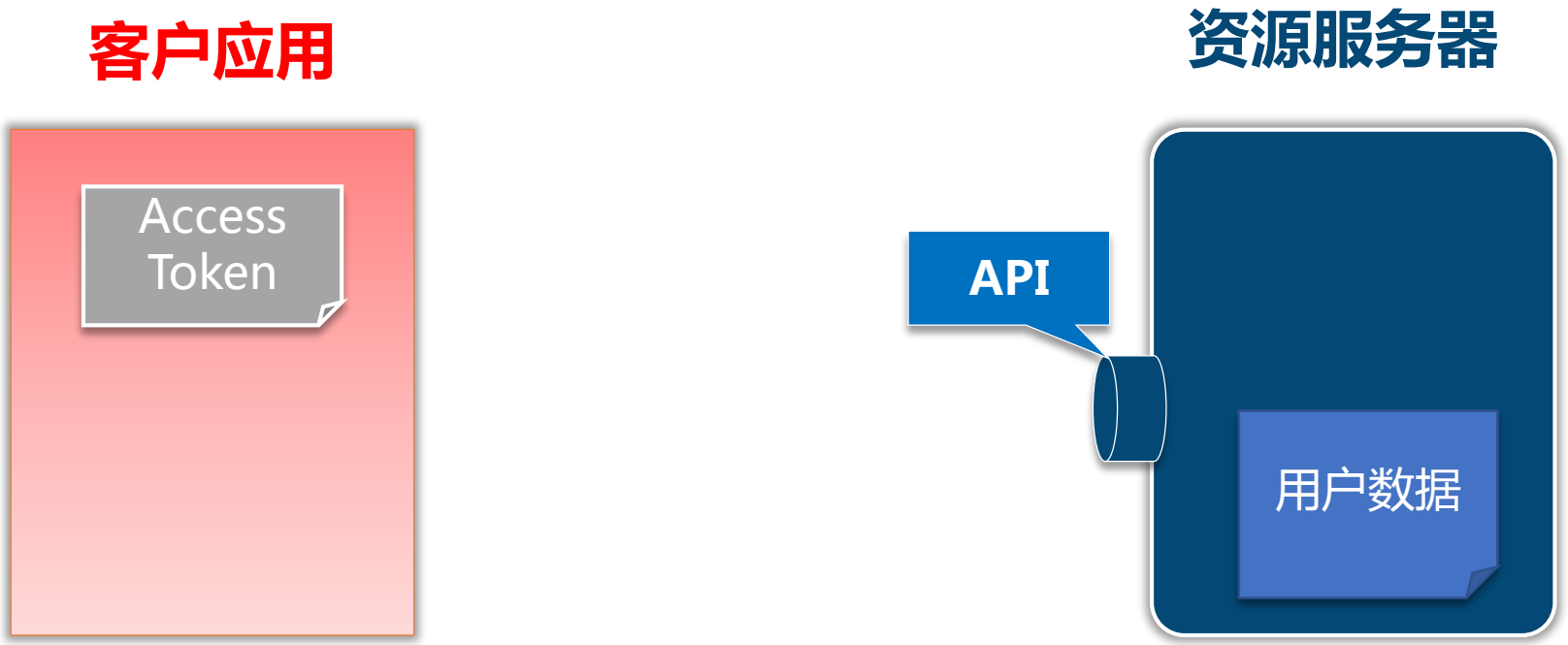
恶意应用也能访问用户数据

需要一种机制保护用户数据

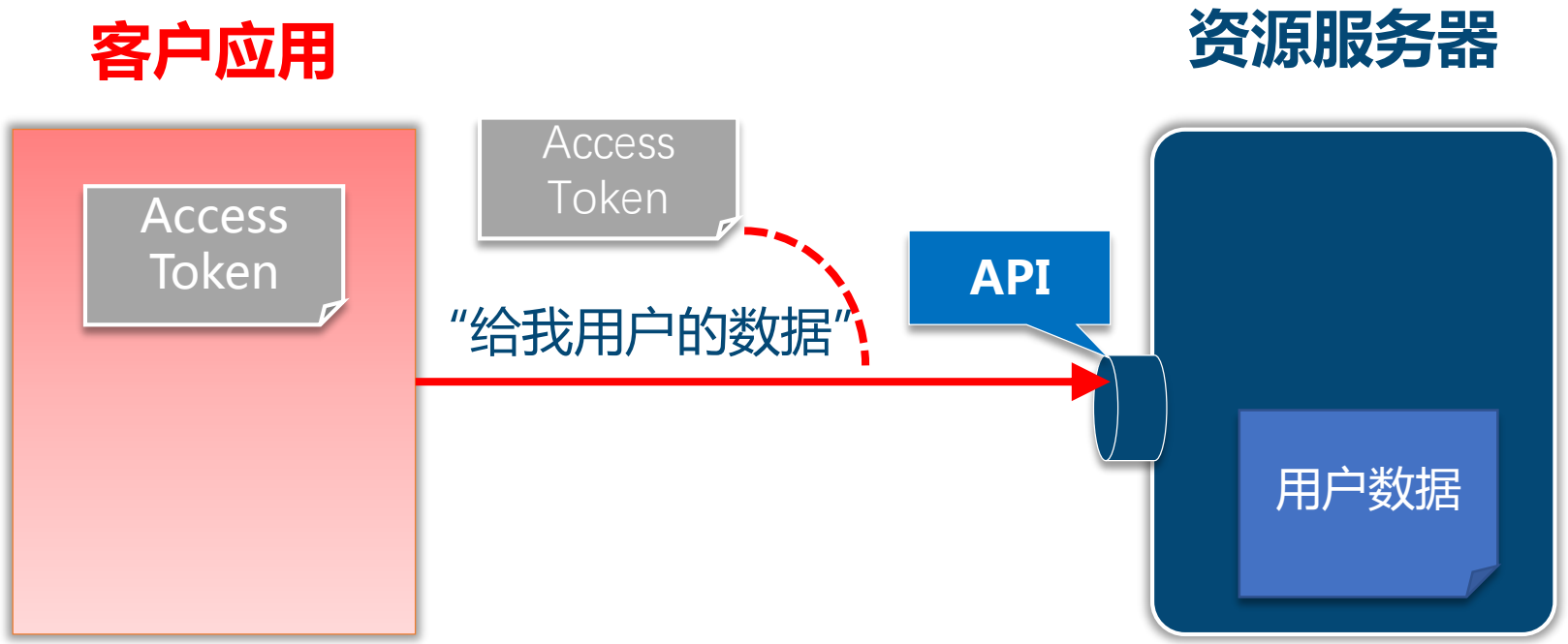


恶意应用也能访问用户数据

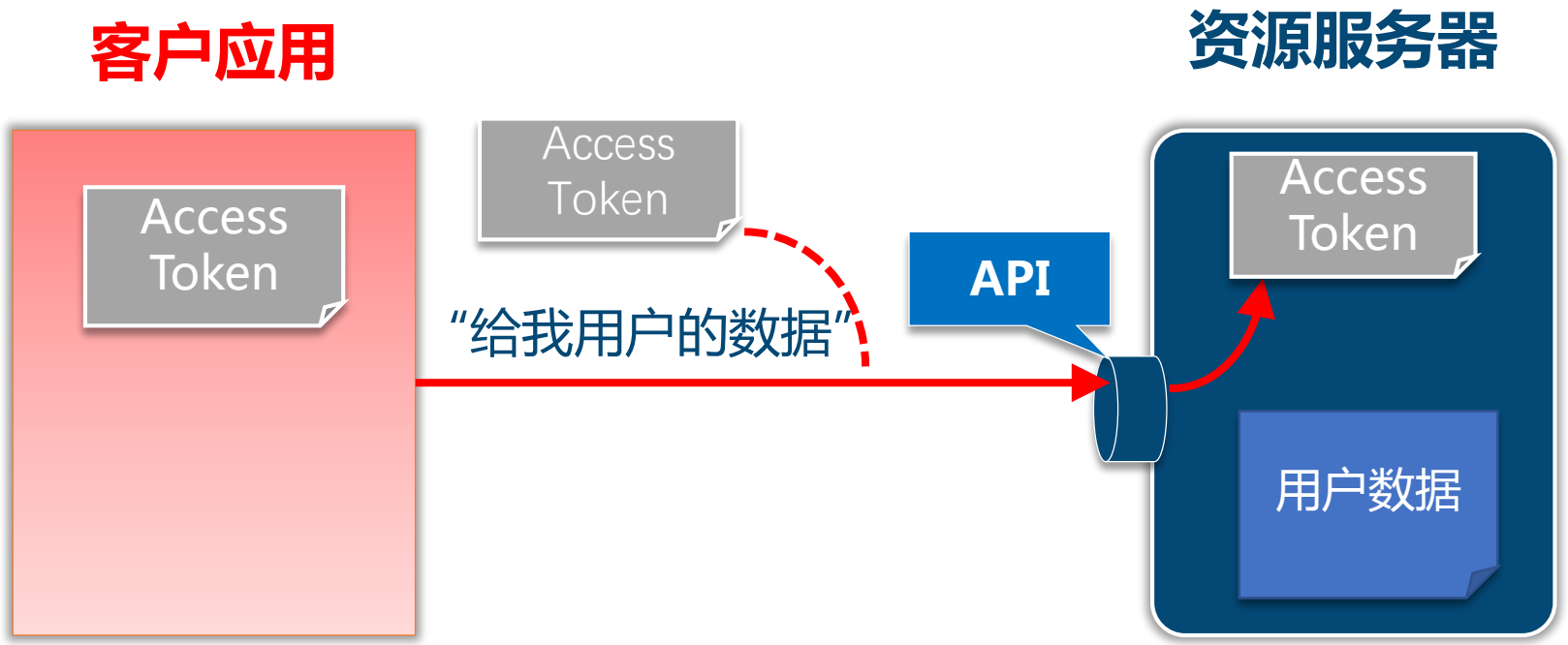
业界实践是提前给客户应用颁发一个Access Token，它表示客户应用被授权可以访问用户数据



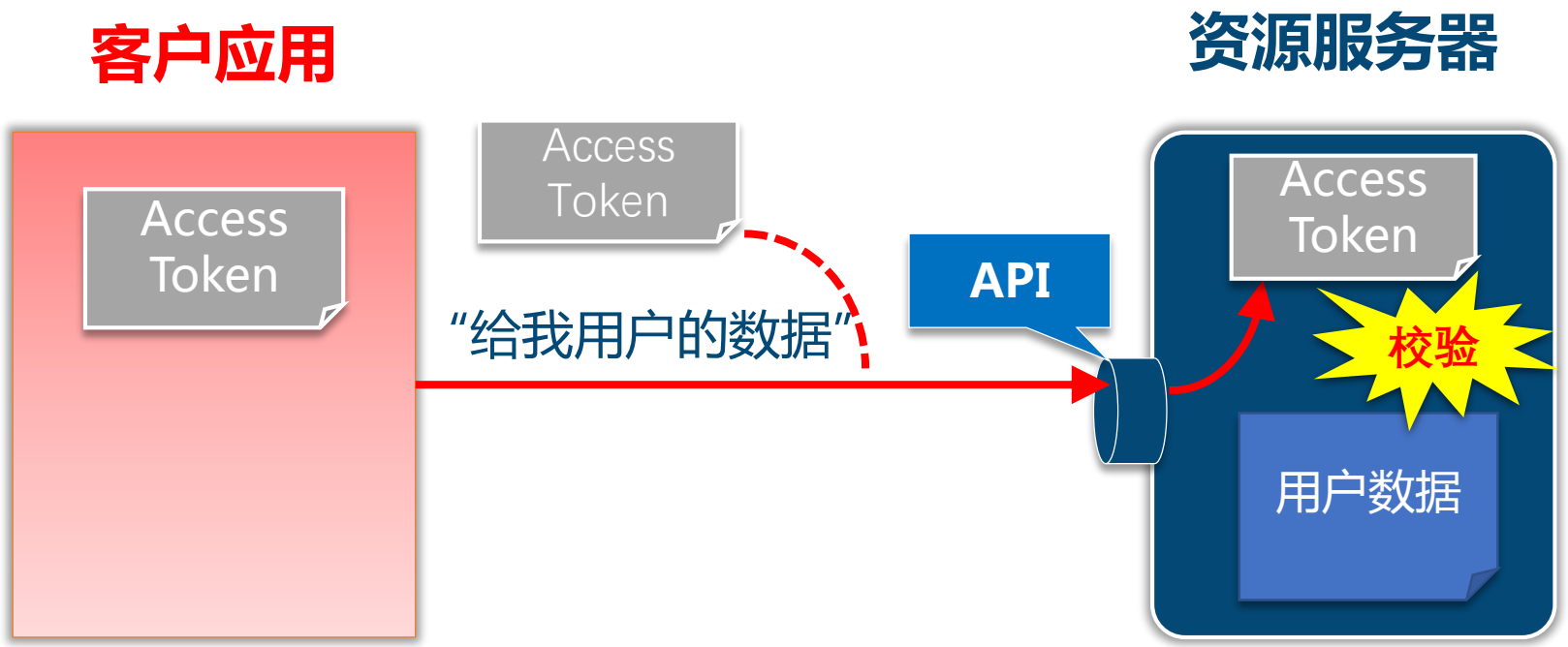
访问用户数据时，给出Access Token



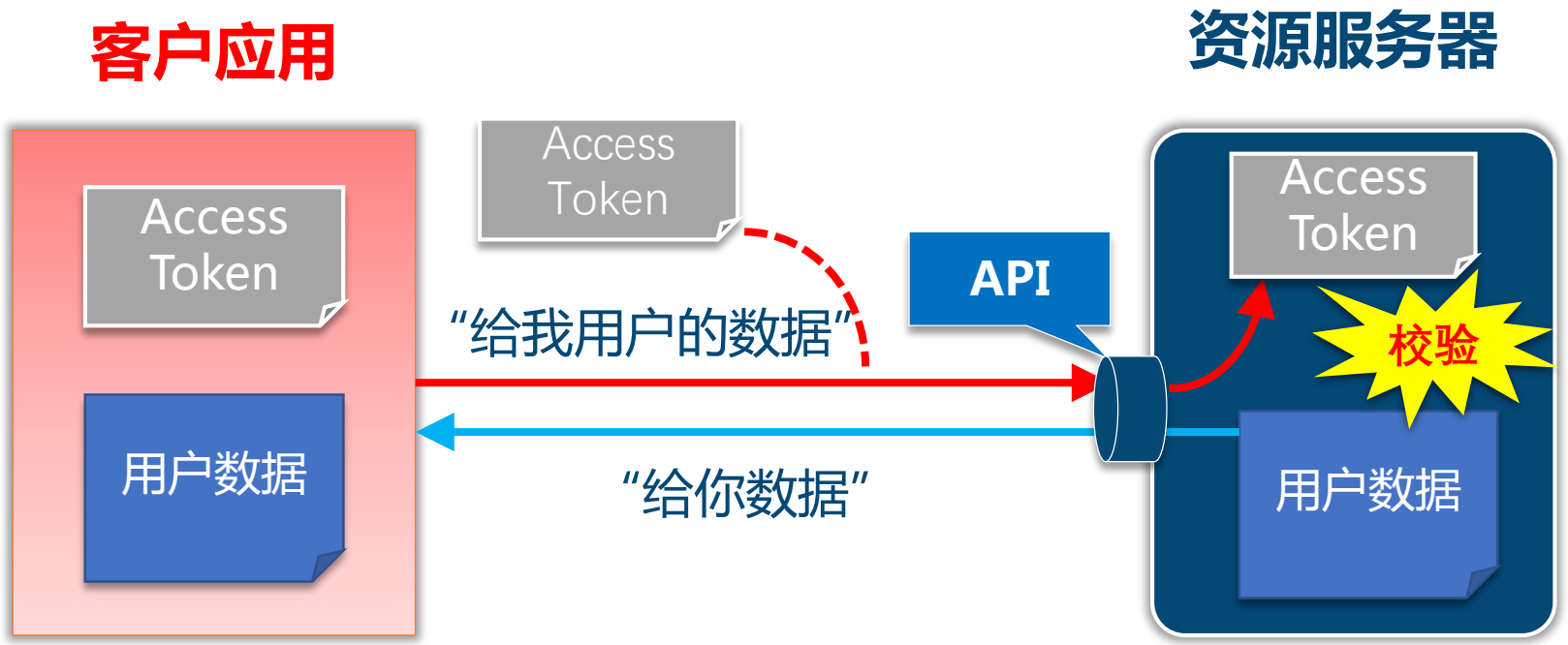
资源服务器取出请求中的Access Token



并校验Access Token确认客户应用有访问用户数据的权限

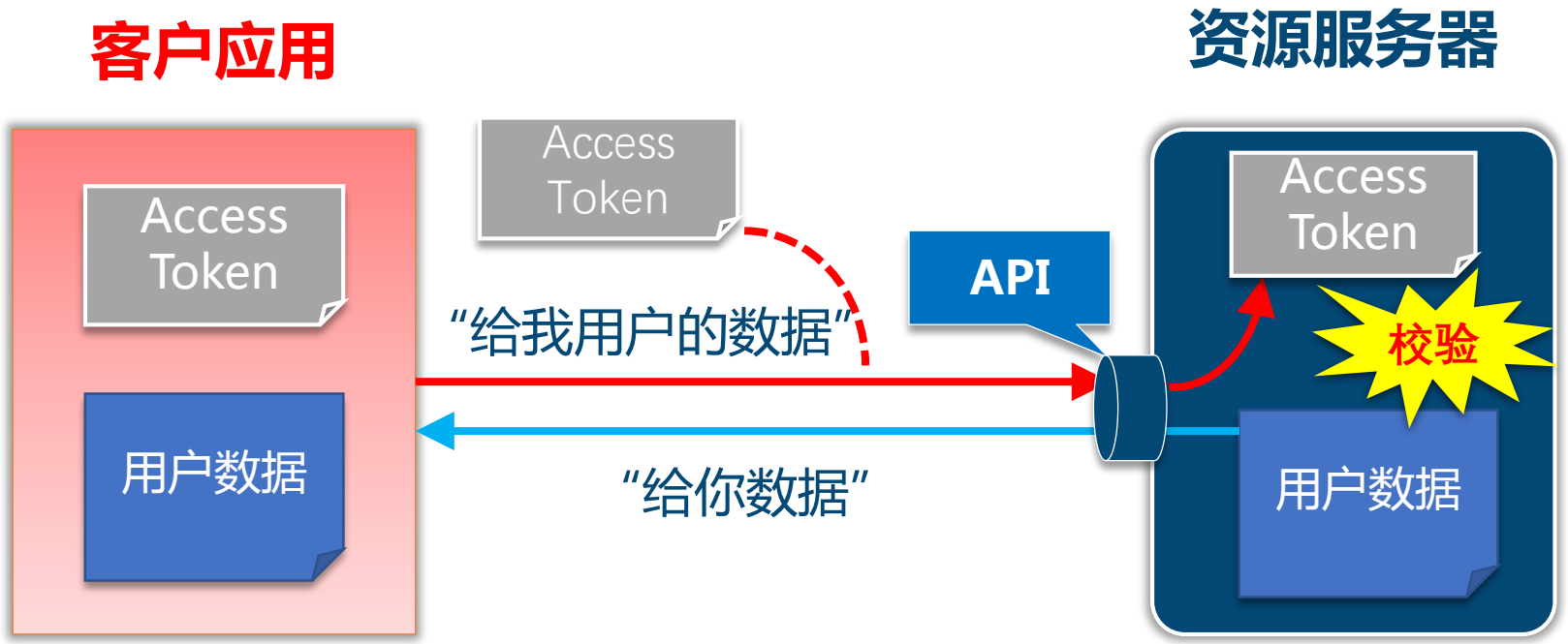


校验通过后，资源服务器返回用户数据

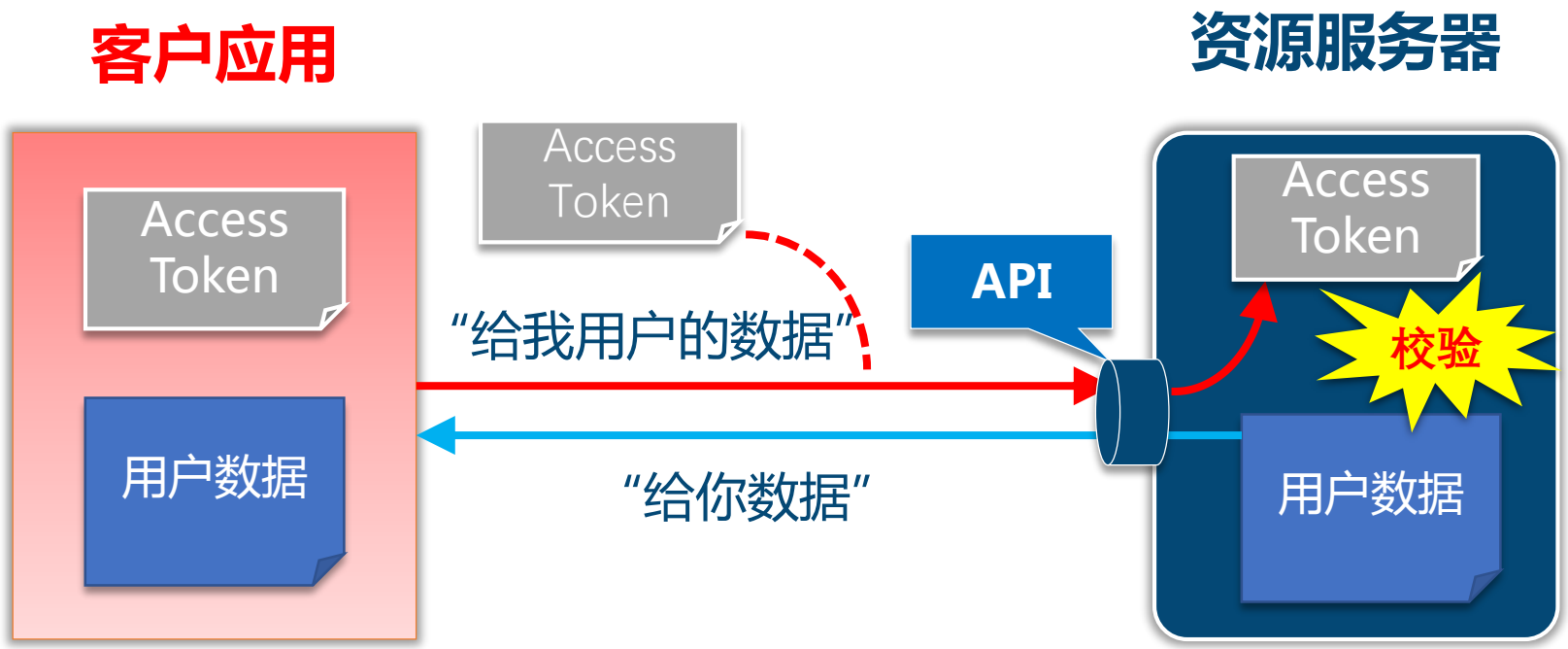


该机制可以工作的前提是

必须提前给客户应用颁发Access Token



需要颁发Access Token的角色

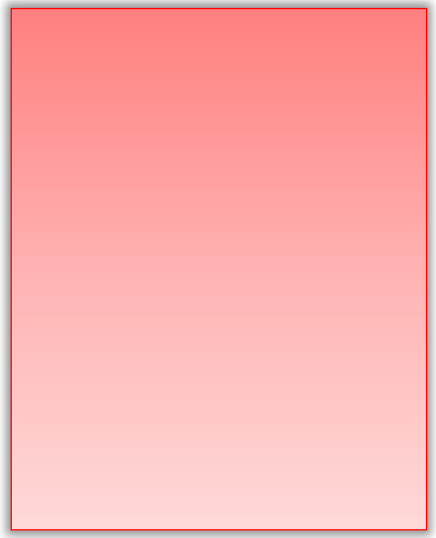


谁颁发Access Token呢？

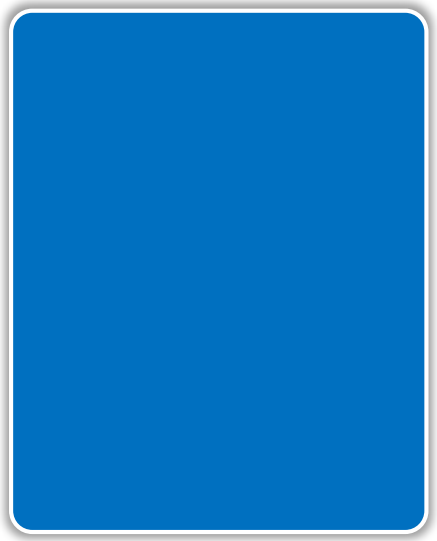
授权服务器

授权服务器和客户应用的关系如下

客户应用

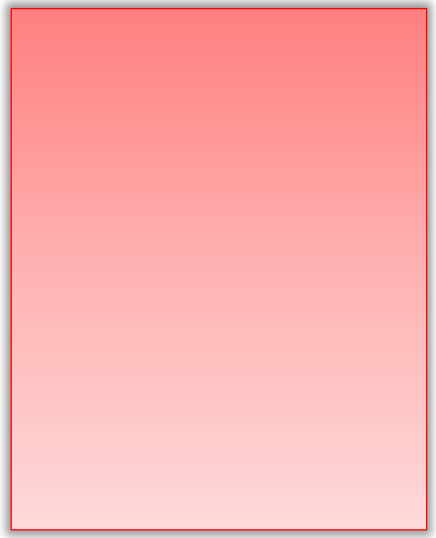


授权服务器



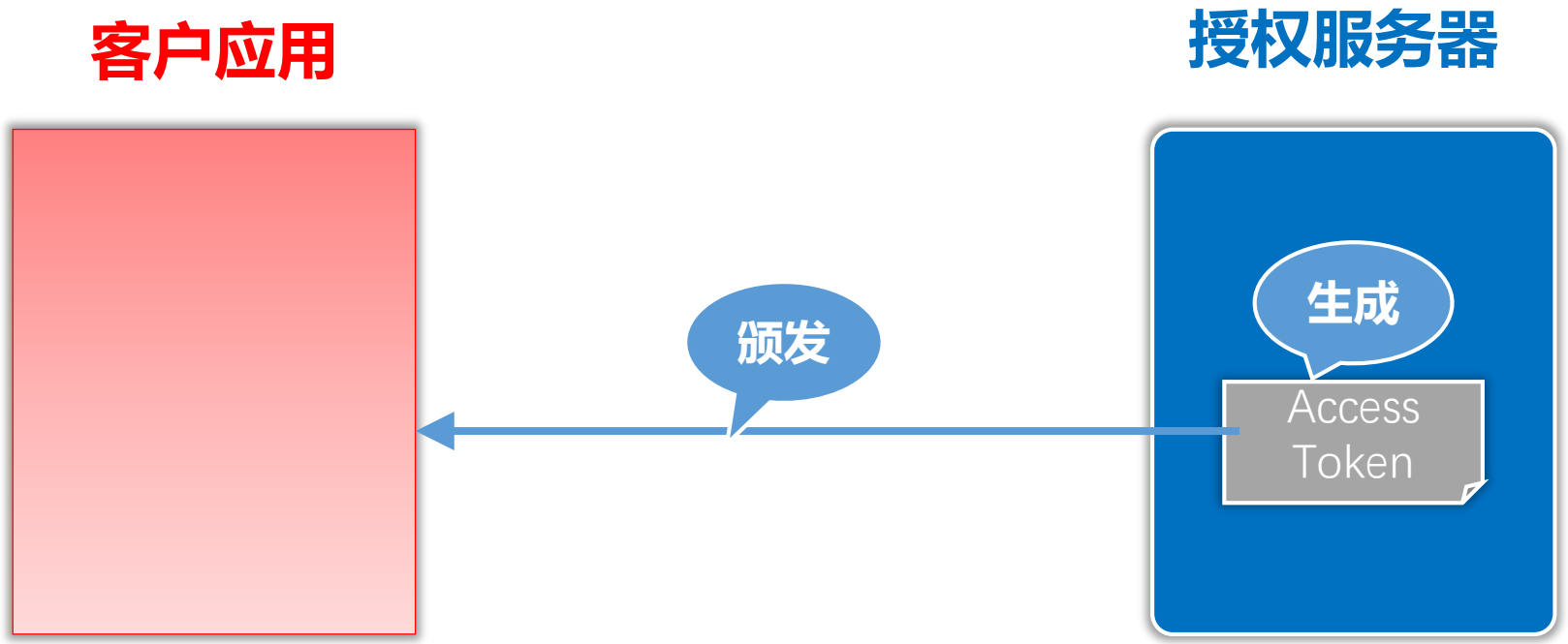
授权服务器负责生成Access Token

客户应用

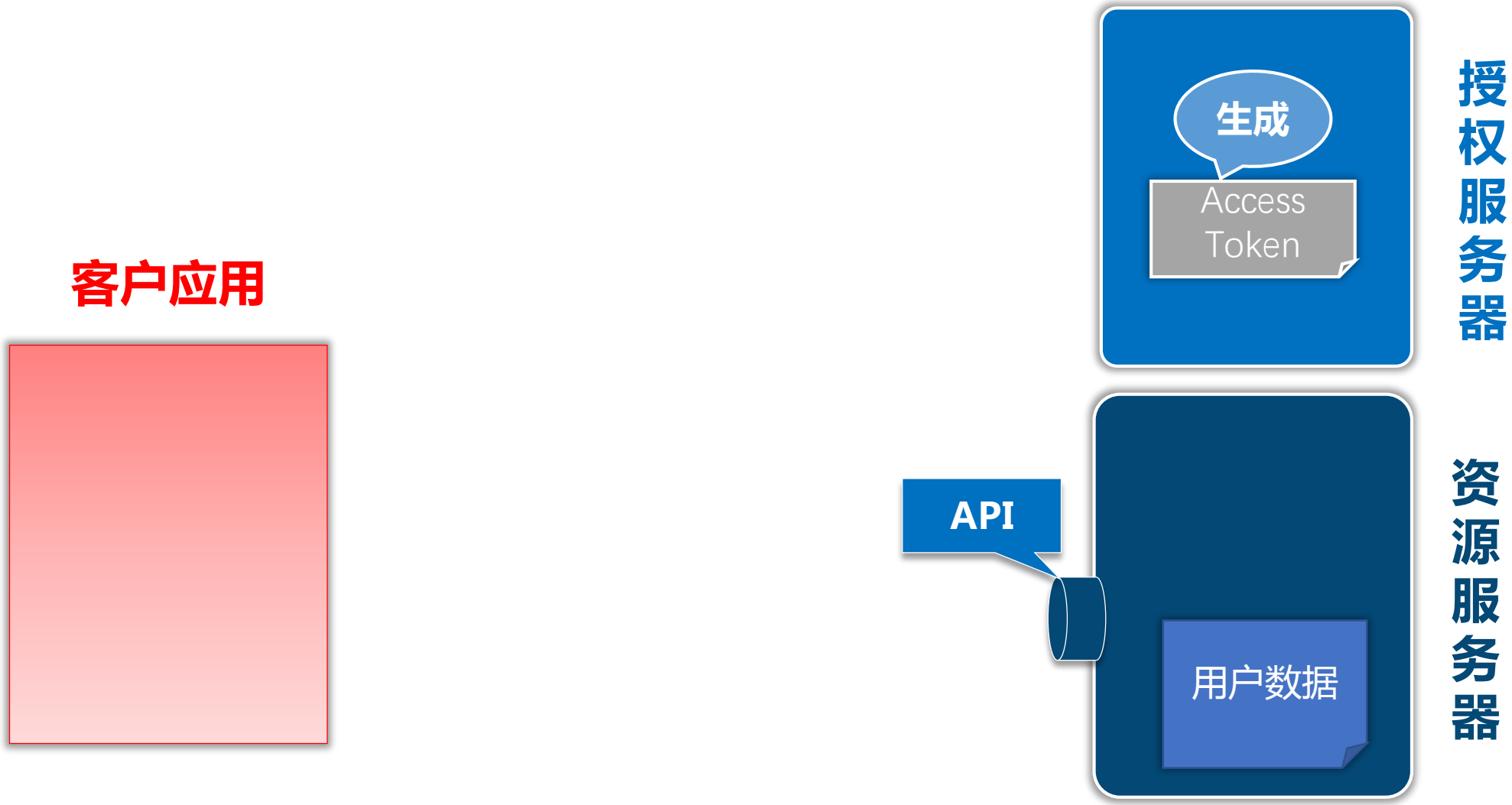


授权服务器

并给客户应用颁发Access Token

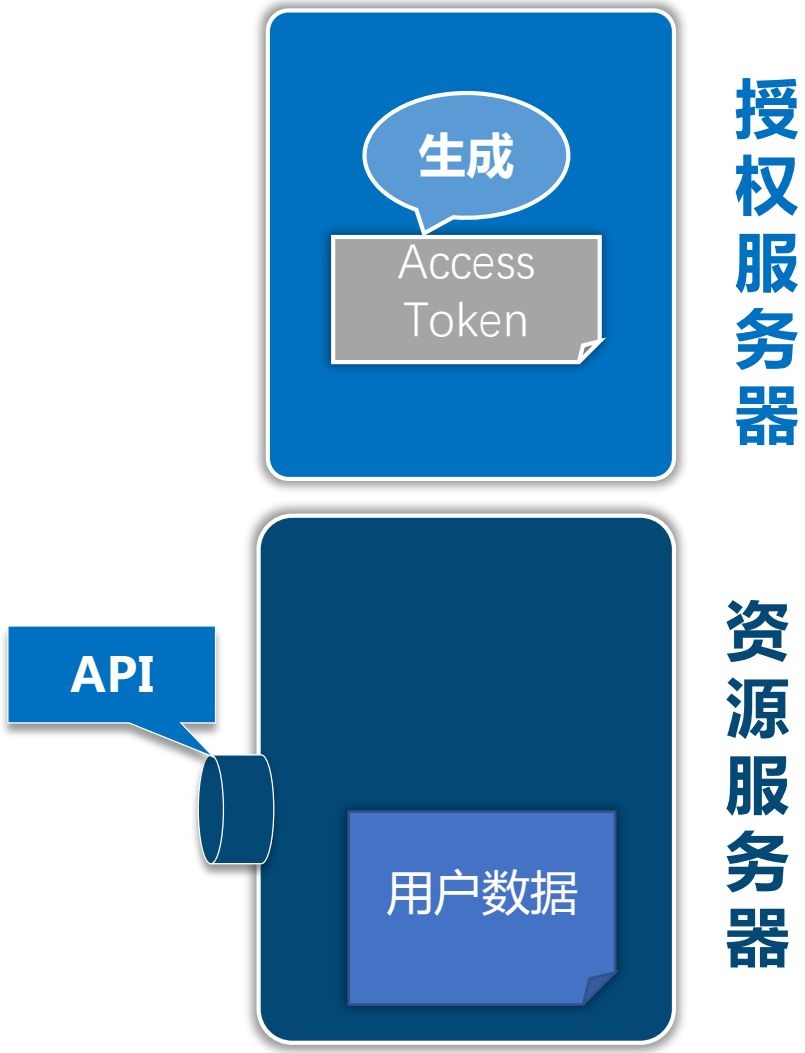


角色回顾：一个授权服务器，一个客户应用，一个资源服务器

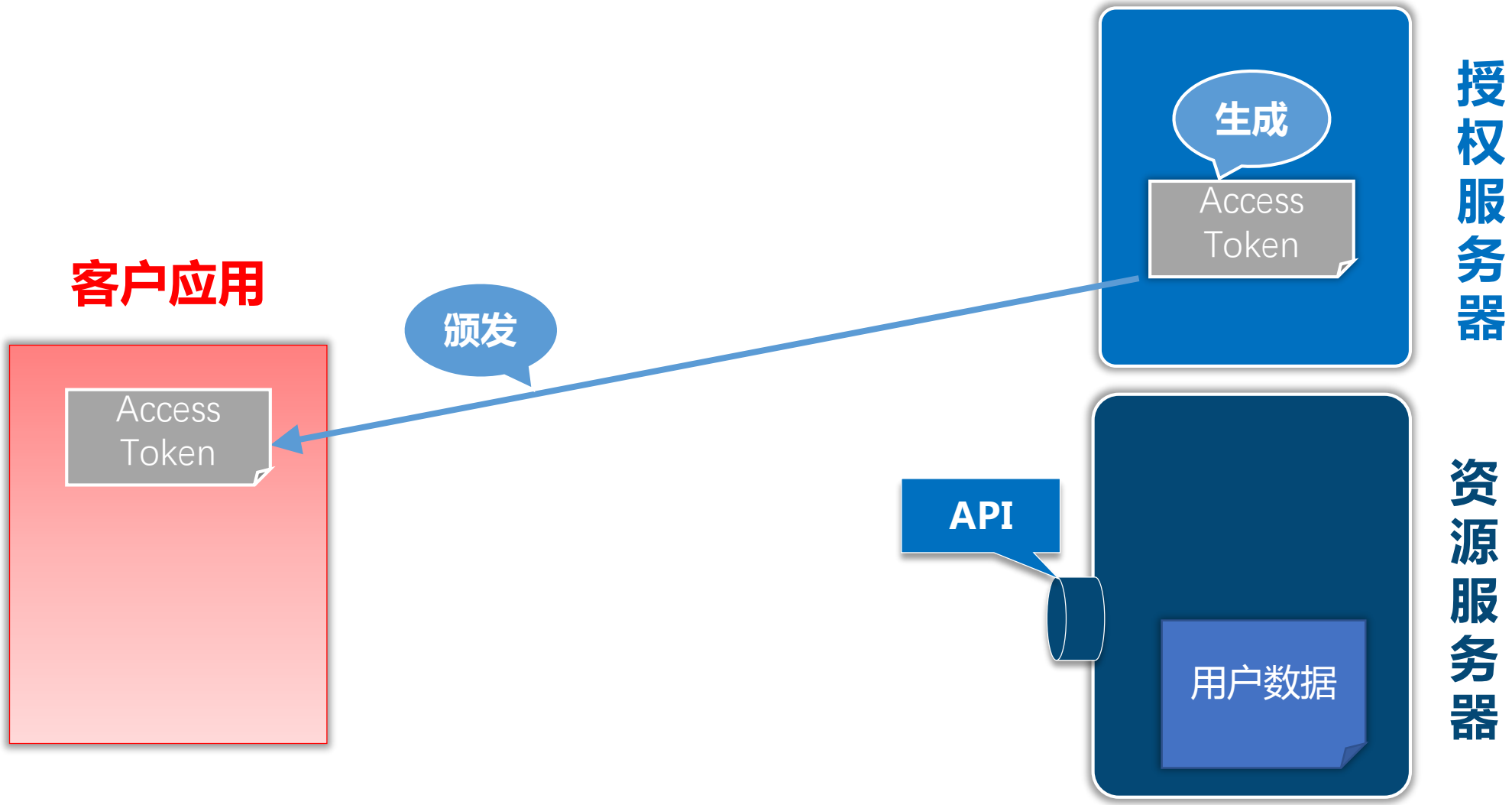


授权服务器负责生成Access Token

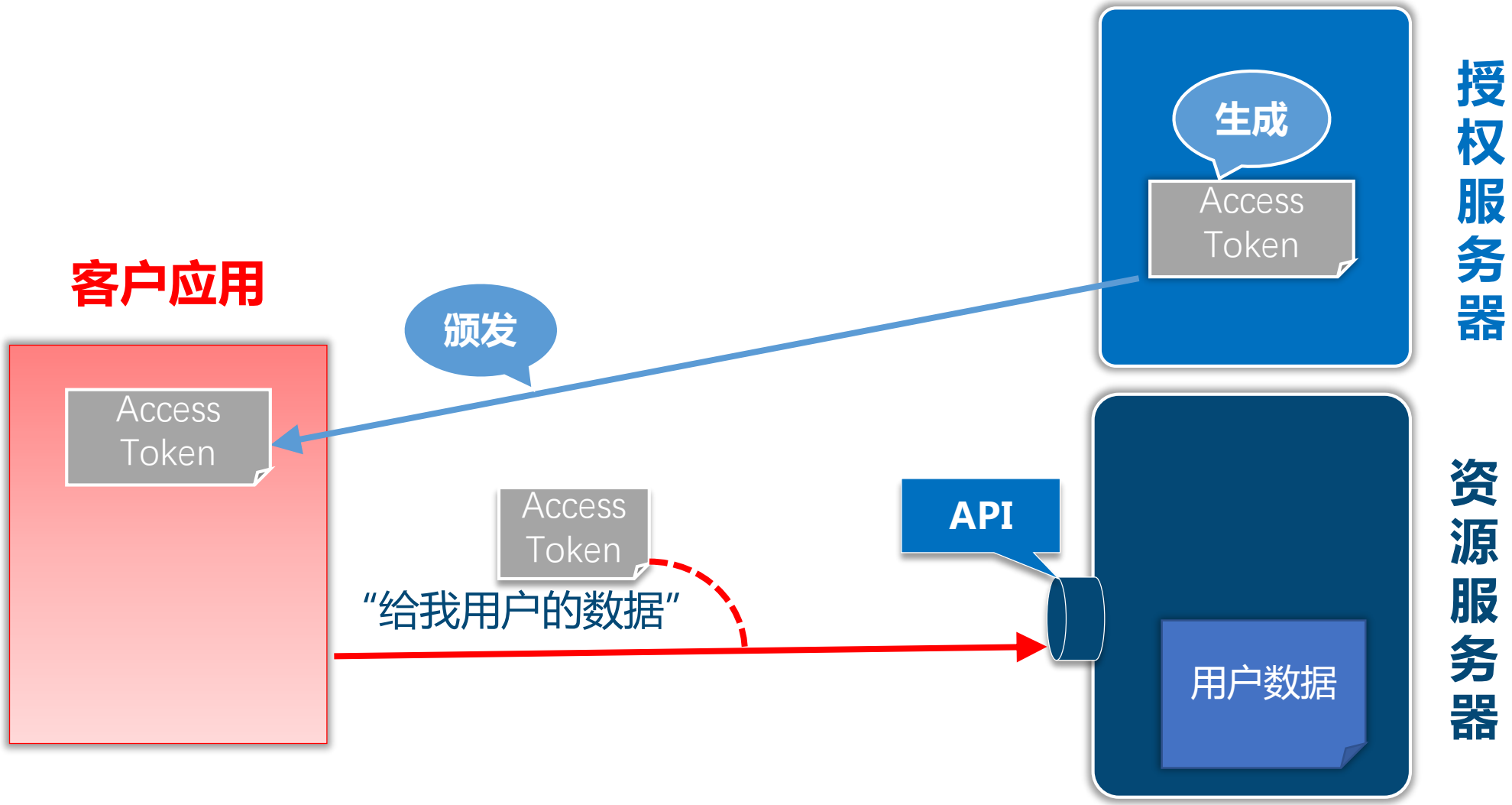
客户应用



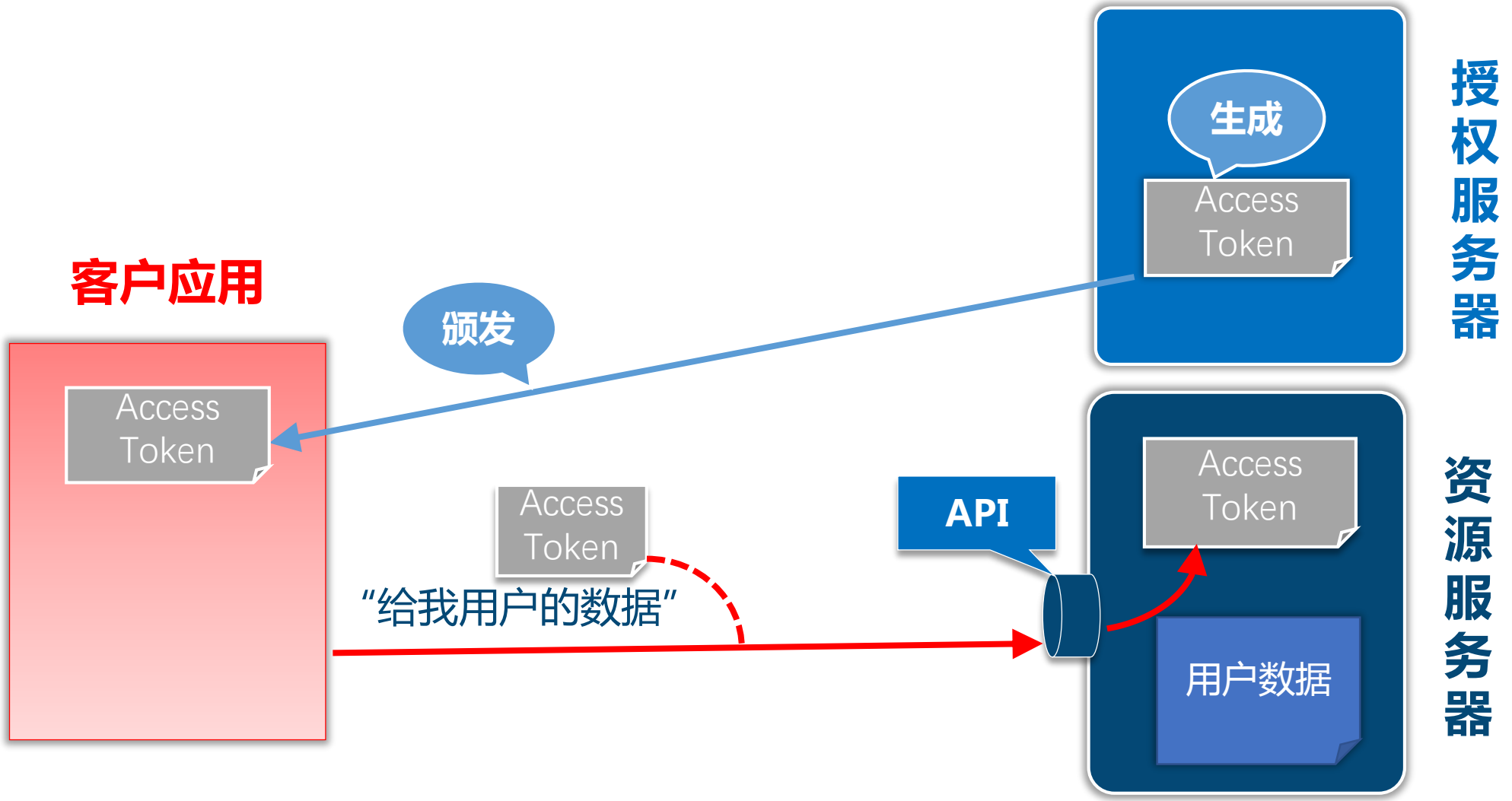
并将Access Token颁发给客户应用



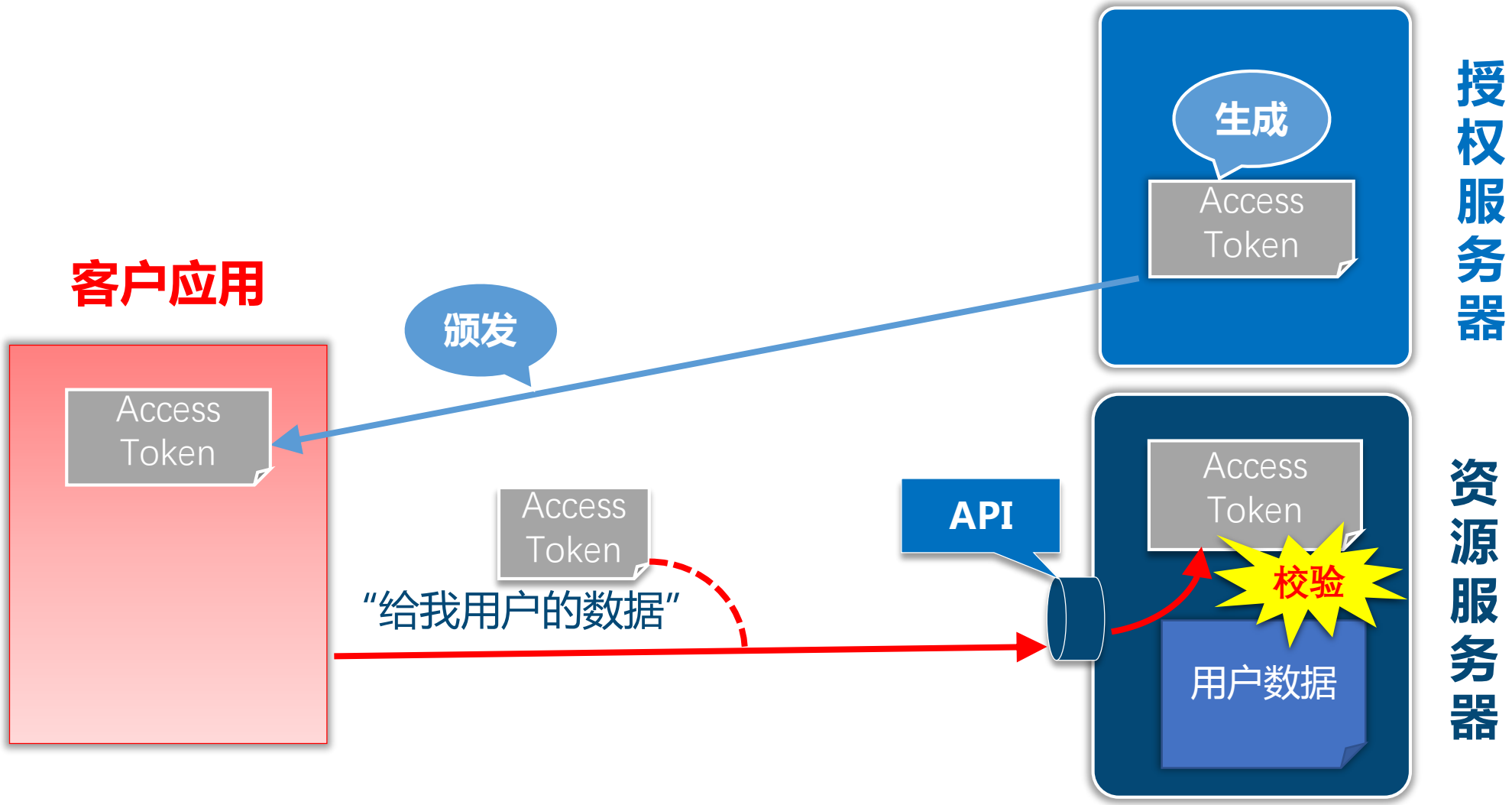
客户应用带上Access Token访问用户数据



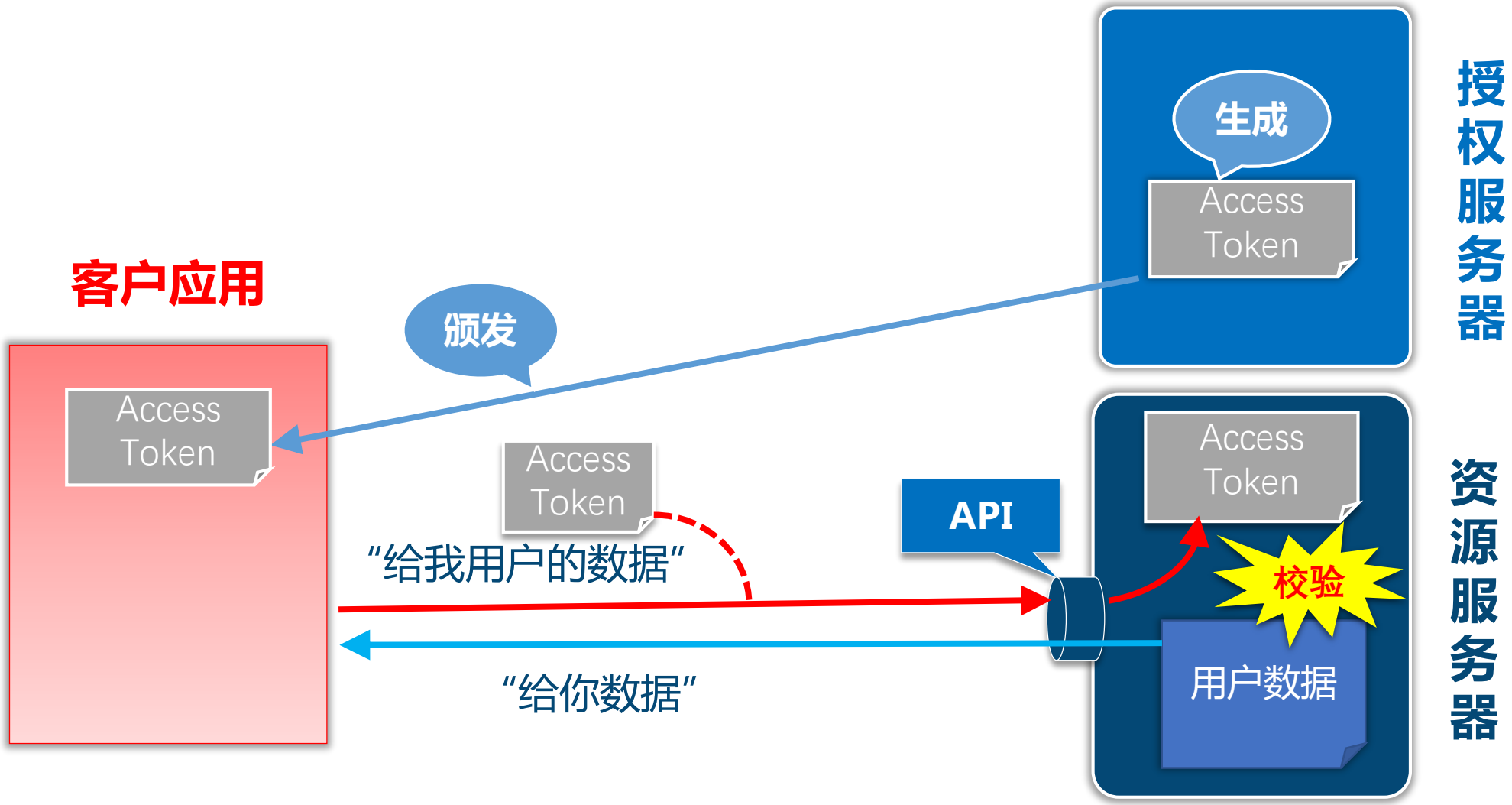
资源服务器从请求中取出Access Token



校验Access Token具有访问用户数据的权限

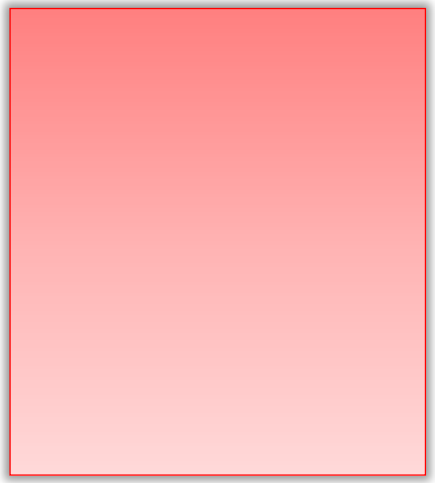


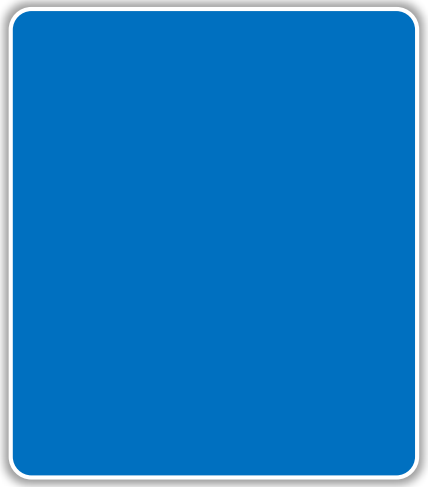
校验Access Token具有访问用户数据的权限



上面的流程中第一步是授权服务器生成Access Token，在真实流程中，在颁发Token前先要征询用户同意

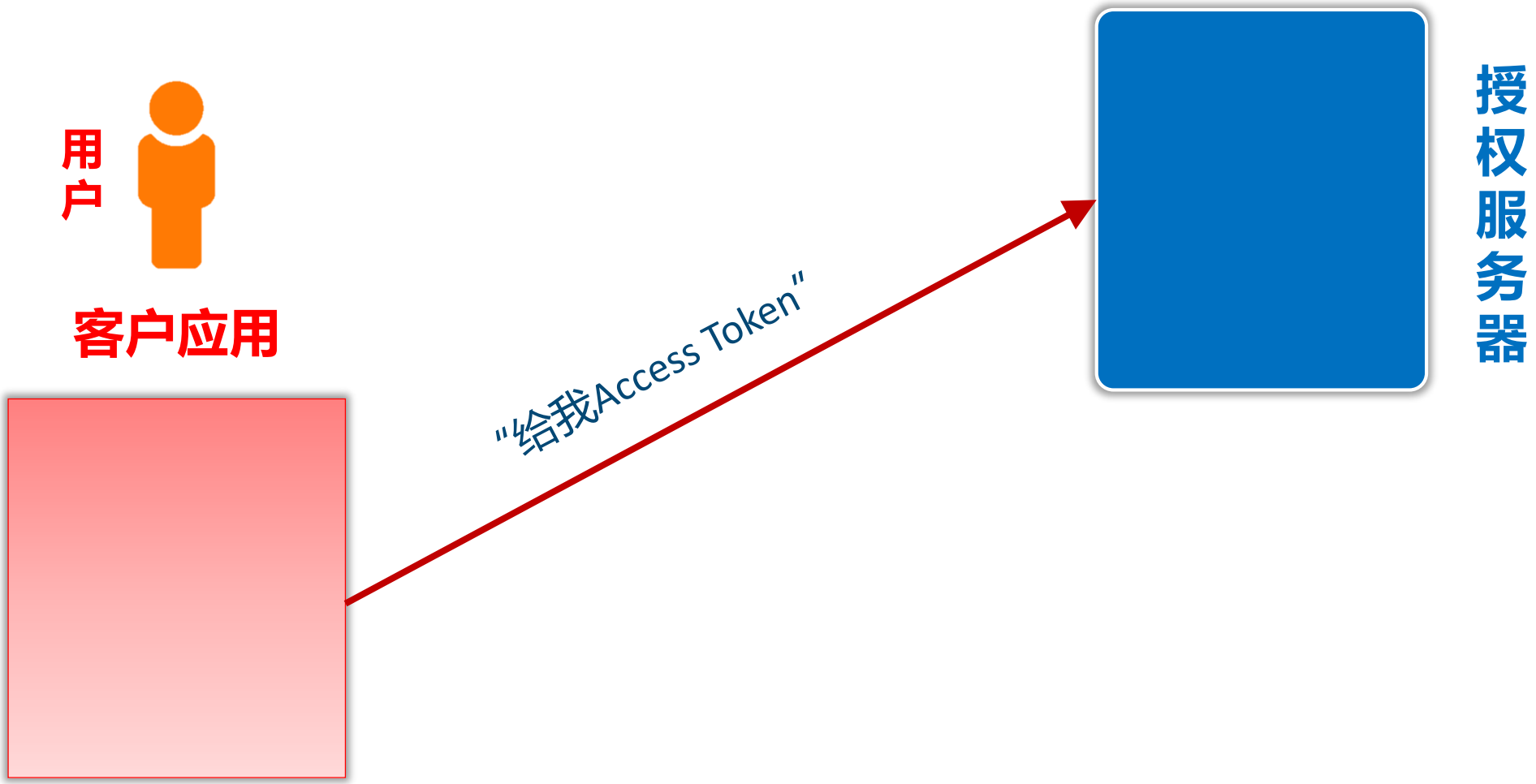
客户应用



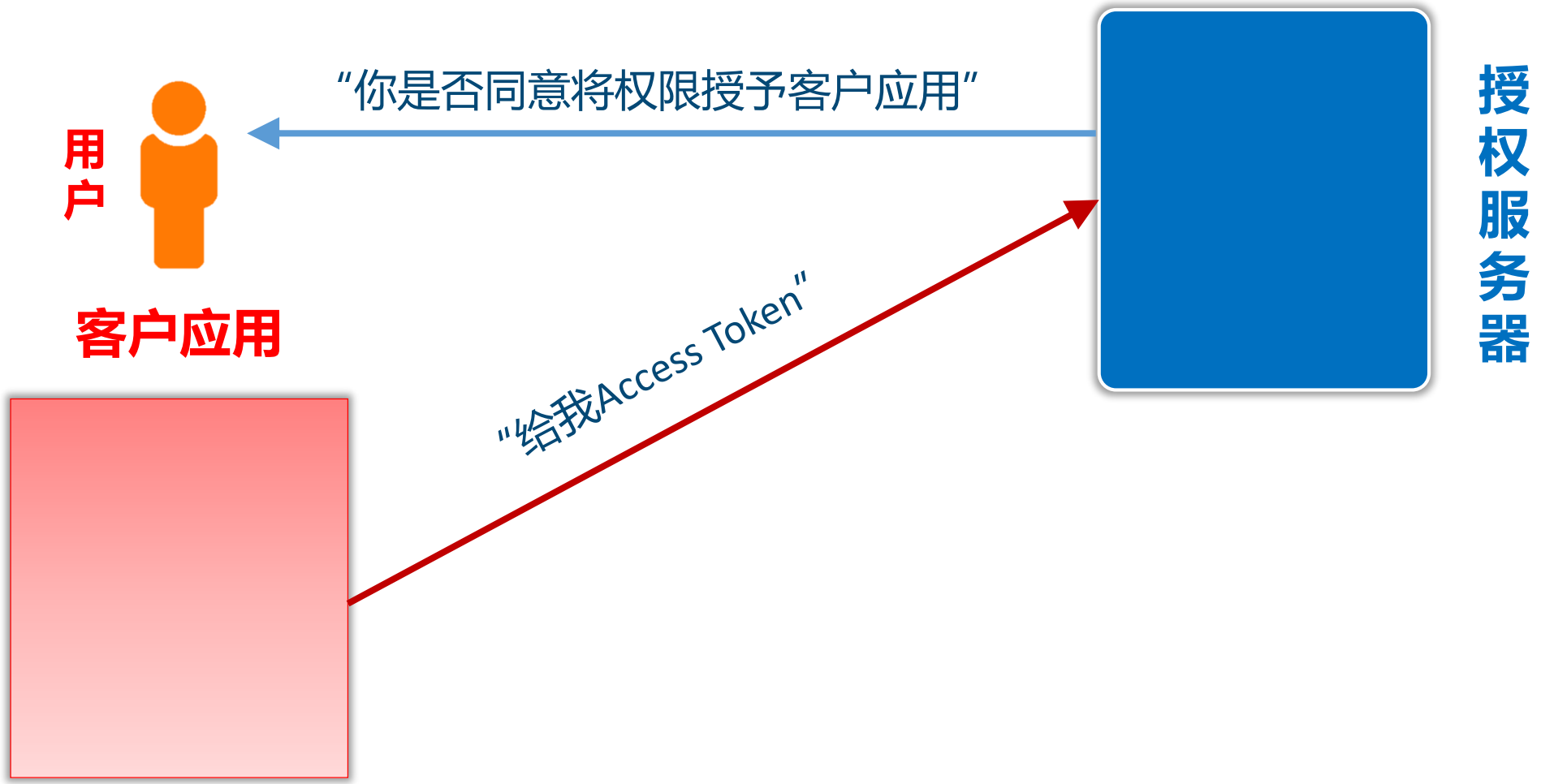


授权服务器

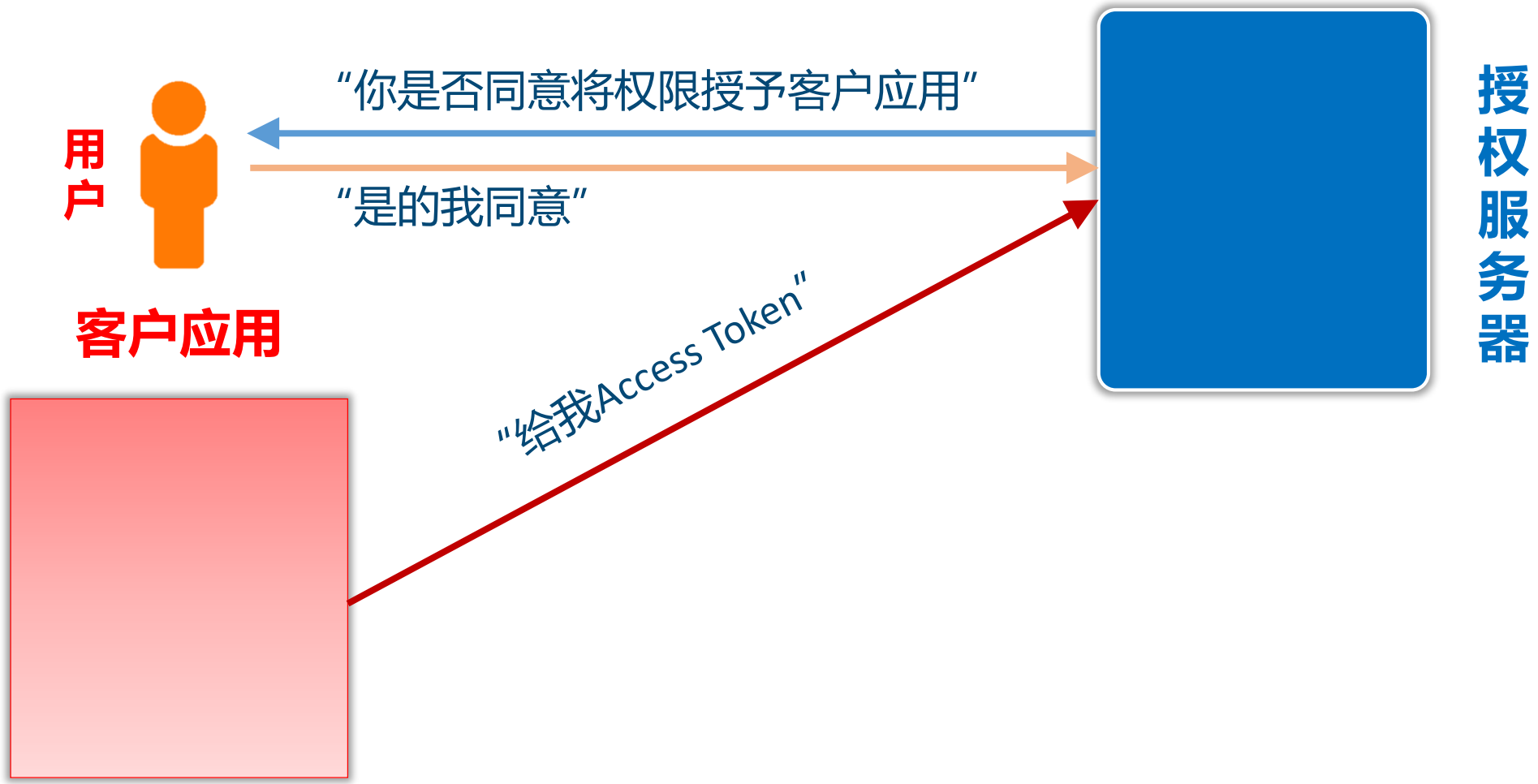
首先客户应用请求Access Token



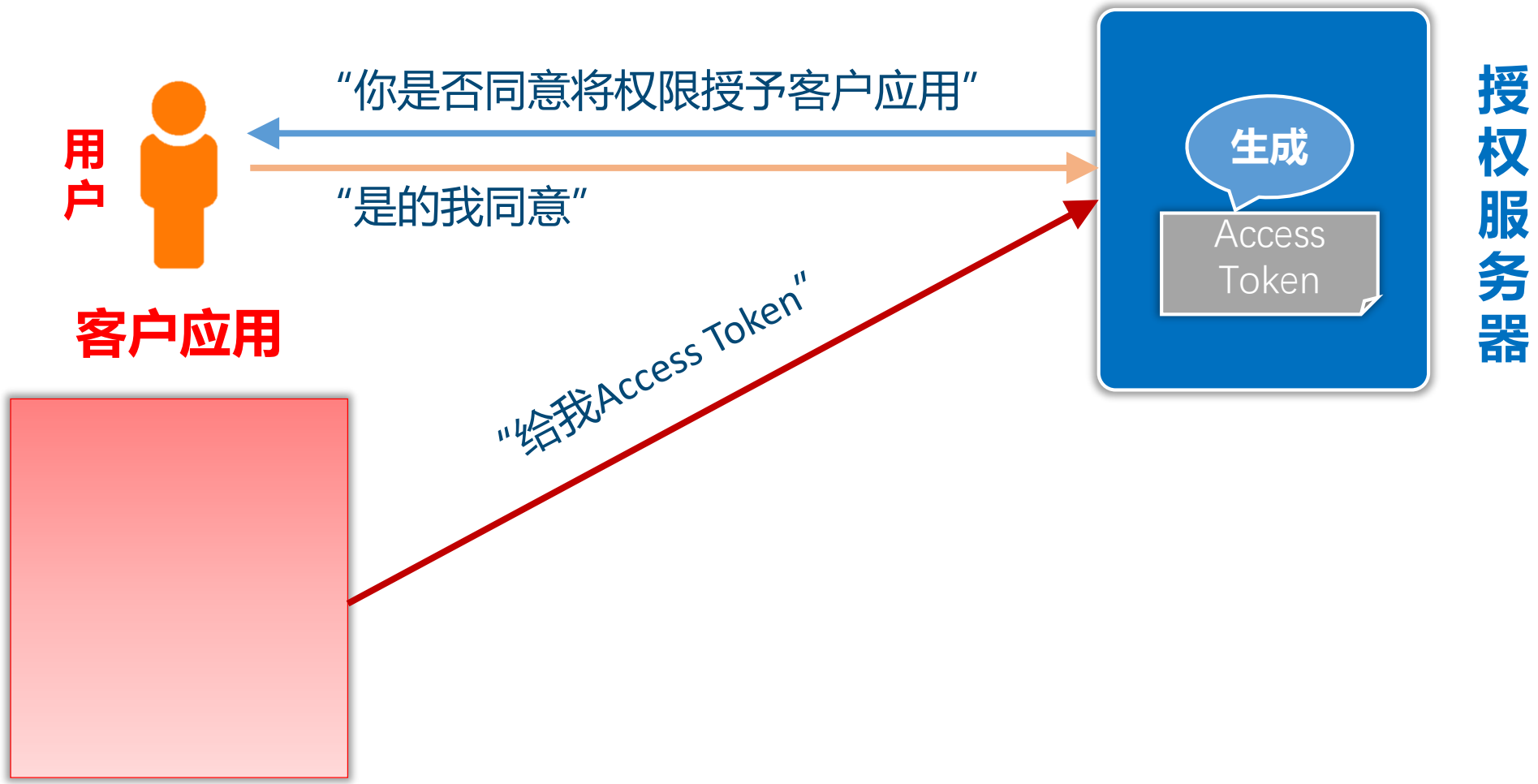
授权服务器征询用户意见，是否将权限授予客户应用



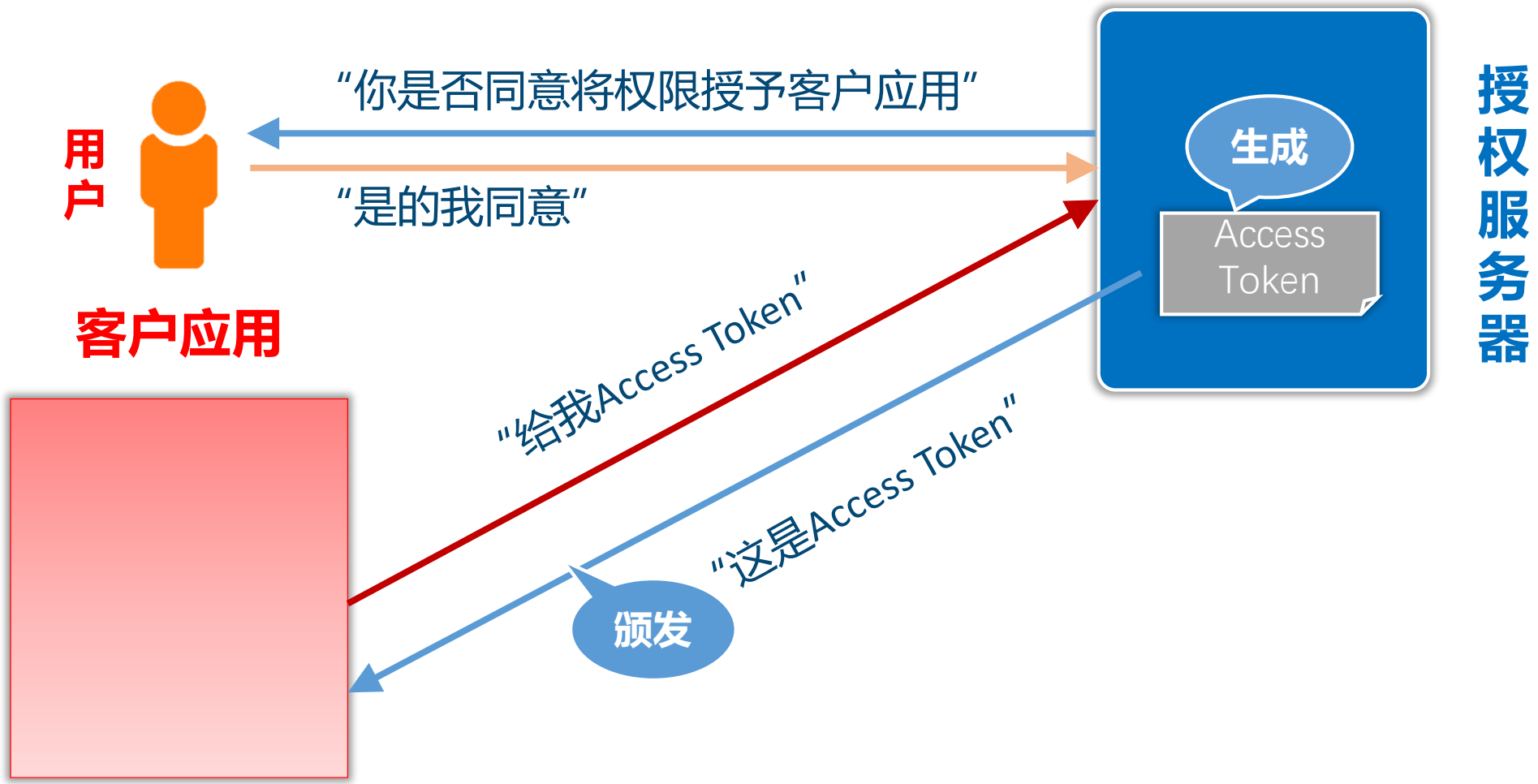
如果用户同意授权服务器颁发token



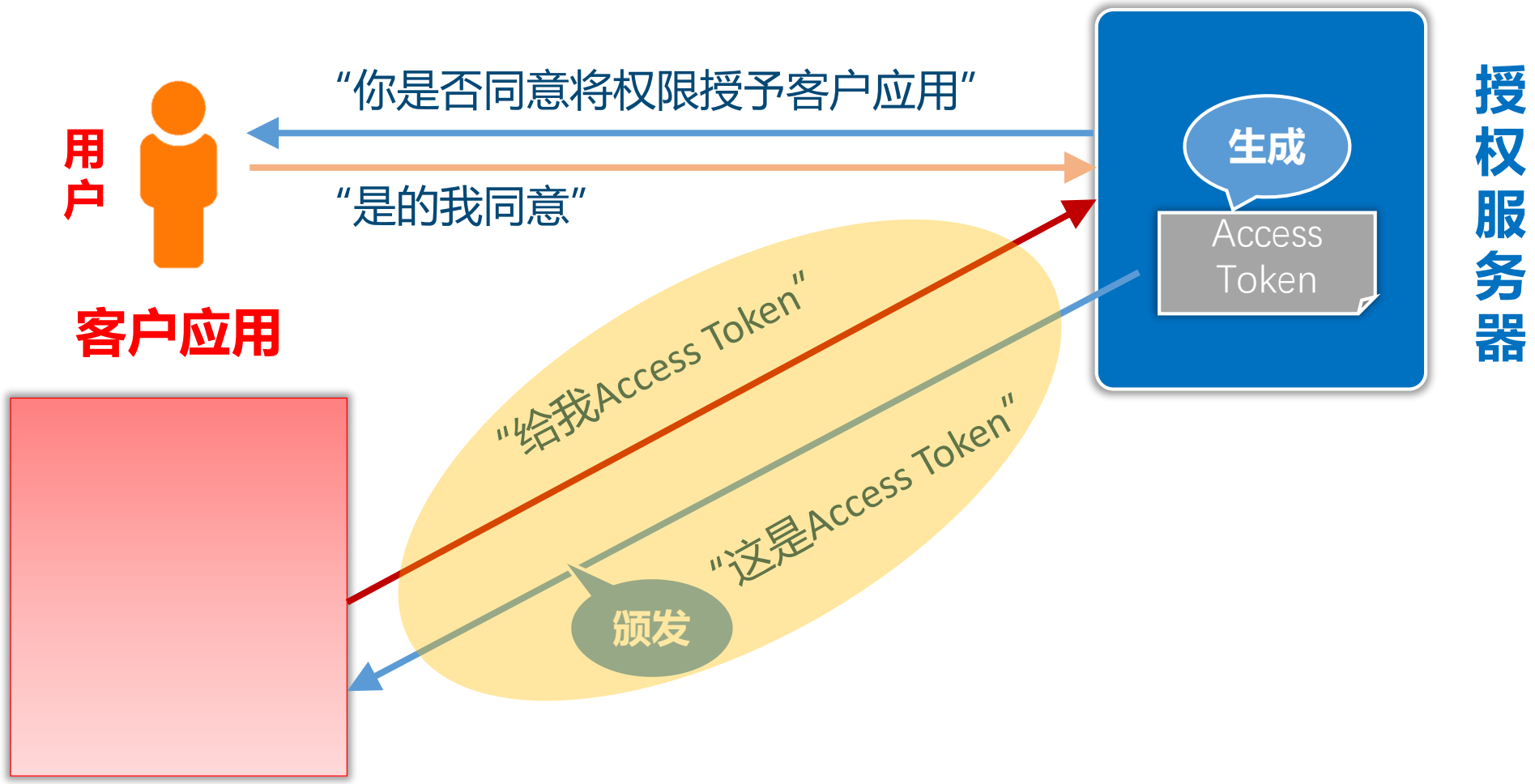
授权服务器生成一个Access Token



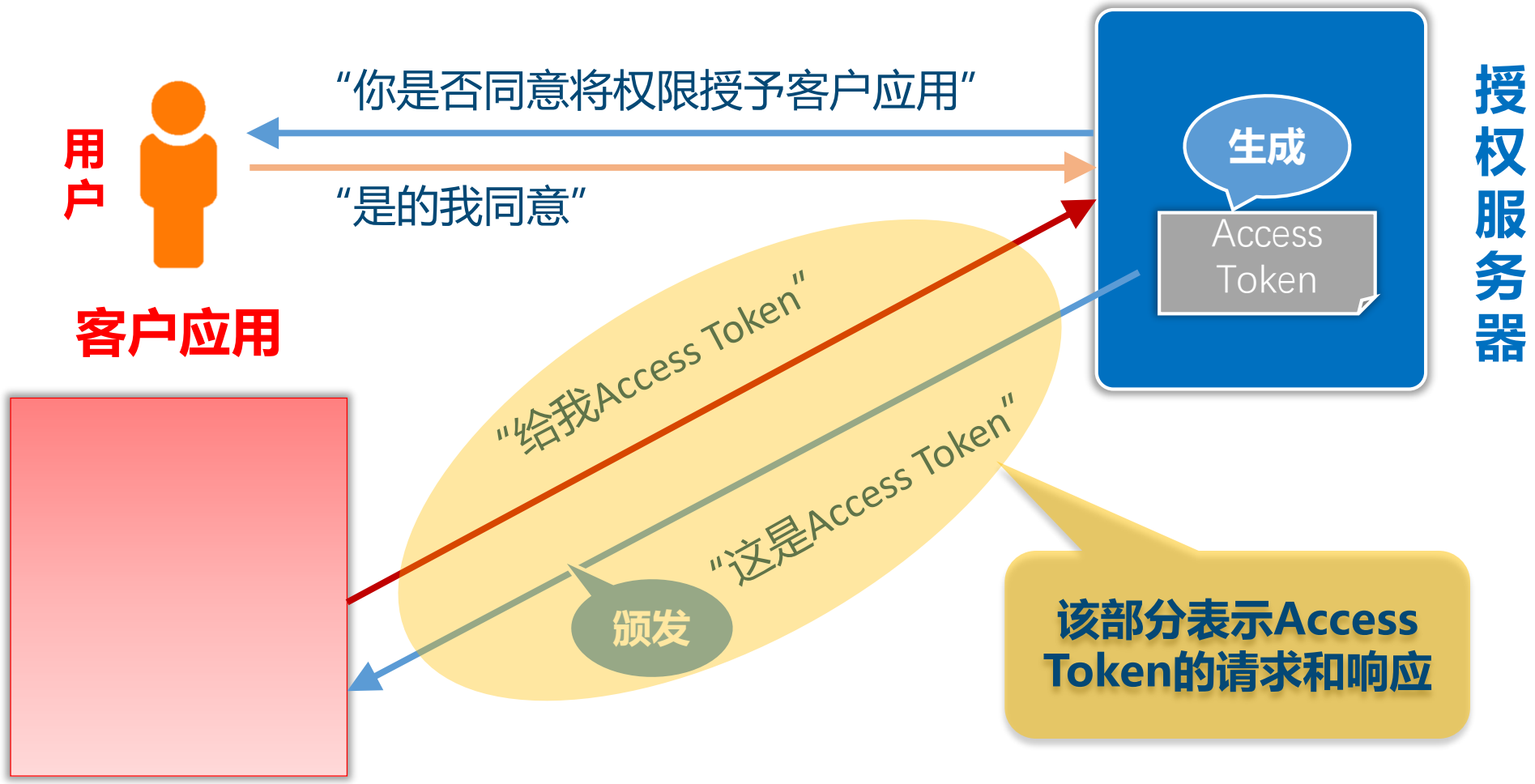
并将token颁发给客户应用



注意黄色椭圆圈起来的部分



注意黄色椭圆圈起来的部分



OAuth 2.0标准化了Access Token的请求和响应部分，OAuth2.0的细节在RFC 6749（OAuth 2.0授权框架）中描述