<https://www.cnblogs.com/XiongMaoMengNan/p/6785155.html>

[返回主页](https://www.cnblogs.com/XiongMaoMengNan/)

# [熊猫猛男](https://www.cnblogs.com/XiongMaoMengNan/)

## "与善人居，如入兰芷之室，久而不闻其香，则与之化矣。与恶人居，如入鲍鱼之肆，久而不闻其臭，亦与之化矣。"愚非君子，但求兰芷而室之，得鲍鱼而食之矣。

* [**博客园**](https://www.cnblogs.com/)
* [**首页**](https://www.cnblogs.com/XiongMaoMengNan/)
* [**新随笔**](https://i.cnblogs.com/EditPosts.aspx?opt=1)
* [**联系**](https://msg.cnblogs.com/send/%E7%86%8A%E7%8C%AB%E7%8C%9B%E7%94%B7)
* [**订阅**](https://www.cnblogs.com/XiongMaoMengNan/rss)
* [**管理**](https://i.cnblogs.com/)

随笔 - 43  文章 - 0  评论 - 18

# [OAuth 2.0: Bearer Token Usage](https://www.cnblogs.com/XiongMaoMengNan/p/6785155.html)

　　Bearer Token ([RFC 6750](http://www.rfcreader.com/#rfc6750)) 用于OAuth 2.0授权访问资源，任何Bearer持有者都可以无差别地用它来访问相关的资源，而无需证明持有加密key。一个Bearer代表授权范围、有效期，以及其他授权事项；一个Bearer在存储和传输过程中应当防止泄露，需实现Transport Layer Security (TLS)；一个Bearer有效期不能过长，过期后可用Refresh Token申请更新。

**一. 资源请求**

Bearer实现资源请求有三种方式：Authorization Header、Form-Encoded Body Parameter、URI Query Parameter，这三种方式优先级依次递减

* **Authorization Header**：该头部定义与Basic方案类似
* GET /resource HTTP/1.1
* Host: server.example.com

Authorization: Bearer mF\_9.B5f-4.1JqM

* **Form-Encoded Body Parameter**： 下面是用法实例
* POST /resource HTTP/1.1
* Host: server.example.com
* Content-Type: application/x-www-form-urlencoded

access\_token=mF\_9.B5f-4.1JqM

使用该方法发送Bearer须满足如下条件：

1.头部必须包含"Content-Type: application/x-www-form-urlencoded"

2.entity-body必须遵循application/x-www-form-urlencoded编码([RFC 6749](http://www.rfcreader.com/#rfc6749_line3235))

3.如果entity-body除了access\_token之外，还包含其他参数，须以"&"分隔开

4.entity-body只包含ASCII字符

5.要使用request-body已经定义的请求方法，不能使用GET

如果客户端无法使用Authorization请求头，才应该使用该方法发送Bearer

* **URI Query Parameter**：
* GET /resource?access\_token=mF\_9.B5f-4.1JqM HTTP/1.1

Host: server.example.com  
Cache-Control: no-store

服务端应在响应中使用 Cache-Control: private

**二. WWW-Authenticate头**

　　在客户端未发送有效Bearer的情况下，即错误发生时，资源服务器须发送WWW-Authenticate头，下为示例：

HTTP/1.1 401 Unauthorized

WWW-Authenticate: Bearer realm="example", error="invalid\_token", error\_description="The access token expired"

下面将就WWW-Authenticate字段的用法进行详细描述(下列这些属性/指令不应重复使用)：

* **Bearer**：Beare作为一种认证类型(基于OAuth 2.0)，使用"Bearer"关键词进行定义
* **realm**：与Basic、Digest一样，Bearer也使用相同含义的域定义reaml
* **scope**：授权范围，可选的，大小写敏感的，空格分隔的列表(%x21 / %x23-5B / %x5D-7E)，可以是授权服务器定义的任何值，不应展示给终端用户。OAuth 2.0还规定客户端发送scope请求参数以指定授权访问范围，而在实际授权范围与客户端请求授权范围不一致时，授权服务器可发送scope响应参数以告知客户端下发的token实际的授权范围。下为两个scope用法实例：
* scope="openid profile email"

scope="urn:example:channel=HBO&urn:example:rating=G,PG-13"

* **error**：描述访问请求被拒绝的原因，字符%x20-21 / %x23-5B / %x5D-7E之内
* **error\_description**：向开发者提供一个可读的解释，字符%x20-21 / %x23-5B / %x5D-7E之内
* **error\_uri**：absolute URI，标识人工可读解释错误的页面，字符%x21 / %x23-5B / %x5D-7E之内

　　当错误发生时，资源服务器将发送的HTTP Status Code(通常是400, 401, 403, 或405)及Error Code如下：

* **invalid\_request**：请求丢失参数，或包含无效参数、值，参数重复，多种方法发送access token，畸形等。资源服务器将发送HTTP 400 (Bad Request)
* **invalid\_token**：access token过期、废除、畸形，或存在其他无效理由的情况。资源服务器将发送HTTP 401 (Unauthorized)，而客户端则需要申请一个新的access token，然后才能重新发送该资源请求
* **insufficient\_scope**：客户端提供的access token的享有的权限太低。资源服务器将发送HTTP 403 (Forbidden)，同时WWW-Authenticate头包含scope属性，以指明需要的权限范围

如果客户端发送的资源请求缺乏任何认证信息(如缺少access token，或者使用 [RFC 6750](http://www.rfcreader.com/#rfc6750) 所规定的三种资源请求方式之外的任何method)，资源服务器不应该在响应中包含错误码或者其他错误信息，如下即可：

HTTP/1.1 401 Unauthorized

WWW-Authenticate: Bearer realm="example"

**三. Bearer Token Response**

下为示例：

[复制代码](javascript:void(0);)

HTTP/1.1 200 OK

Content-Type: application/json;charset=UTF-8

Cache-Control: no-store

Pragma: no-cache

{

"access\_token":"mF\_9.B5f-4.1JqM",

"token\_type":"Bearer",

"expires\_in":3600,

"refresh\_token":"tGzv3JOkF0XG5Qx2TlKWIA"

}

[复制代码](javascript:void(0);)

**四. 安全威胁**

* **Token 伪造/修改(Token manufacture/modification)**：攻击者伪造或修改已有的token，导致资源服务器授权通过非法访问的客户端。因此需要对token使用数字签名或消息认证码来保证其完整性
* **Token 泄露(Token disclosure)**：Token本身可能包含认证、有效期等敏感信息。因此实现TLS并验证证书是必选项，加密token可用于防止客户端观察token的内容，加密token还可防止token在前端服务器和后端服务器(如果他们没有启用TLS)之间发生泄露
* **Token 改寄(Token redirect)**：攻击者用一个访问A资源服务器的token去请求B资源服务器的资源。因此通常token中可以包含代表资源服务器的标识来防止这种情况的发生
* **Token 重放(Token replay)**：攻击者企图使用曾经使用过的token来请求资源。因此token需包含有效期(比如少于1小时)

 　　另外cookie不能包含token，关于cookie的安全弱点，[RFC 6265](http://www.rfcreader.com/#rfc6265_line1310) 中有如下描述：

[复制代码](javascript:void(0);)

　　A server that uses cookies to authenticate users can suffer security

vulnerabilities because some user agents let remote parties issue

HTTP requests from the user agent (e.g., via HTTP redirects or HTML

forms). When issuing those requests, user agents attach cookies even

if the remote party does not know the contents of the cookies,

potentially letting the remote party exercise authority at an unwary

server.

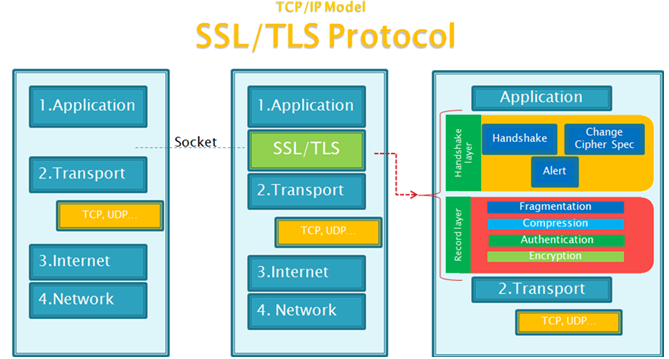
[复制代码](javascript:void(0);)

可见即使服务端实现了TLS，攻击者依旧可以利用cookie来获取机密信息，如果cookie中包含机密信息的话；对此，不得已可将机密信息包含于URLs(前提是已实现了TLS)，但尽量使用更安全的办法，因为浏览器历史记录、服务器日志等可能泄露URLs机密信息。

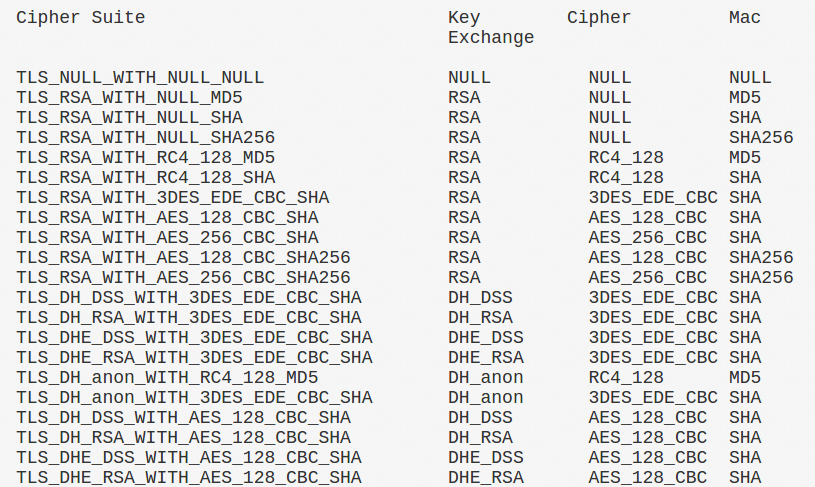
**五. Transport Layer Security (TLS)**

TLS (SSL / TLS)源于NetScape设计的SSL(Secure Sockets Layer，1994 / SSL 1.0、1995 / SSL 2.0、1996 / SSL 3.0)；1999年，IETF接替NetScape，发布了SSL的升级版TLS 1.0，最新为TLS 1.3(draft)，TLS 用于在两个通信应用程序之间提供保密性和数据完整性。目前，应用最广泛的是TLS 1.0，接下来是SSL 3.0，主流浏览器都已经实现了TLS 1.2的支持。TLS 1.0通常被标示为SSL 3.1，TLS 1.1为SSL 3.2，TLS 1.2为SSL 3.3。获取更多信息可参考 [TLS 1.2 / RFC 5246](http://www.rfcreader.com/#rfc5246) 。

　　TLS是一种位于传输层(TCP)和应用层之间的协议层，由记录协议(Record Layer)和握手协议(Handshake Layer)两层构成：



　　TLS算法套件由三个部分组成：了解更多可参考 <http://www.rfcreader.com/#rfc5246_line3649>



1. 密钥交换算法：  RSA或Diffie-Hellman算法的各种变种
2. 加密算法：  AES, DES, Triple-DES, RC4, RC2, IDEA 或 none
3. 摘要算法：  MD5, SHA

**六. application/x-www-form-urlencoded Media Type**

首先**application/x-www-form-urlencoded**这种编码类型未考虑非US ASCII字符的情况，因此待编码的内容(包括名称、值)可先经UTF-8编码，然后再按字节序列进行字符转义操作；而接收这种数据类型则需进行逆向处理。通常各种web编程语言已经提供原生URL编码/URL解码接口(可能不同语言版本的URL编码/解码会有差异)，使用起来也极为方便，因此这里不做详细介绍。

**七. MAC Token**

MAC Token与Bearer Token一样，可作为OAuth 2.0的一种Access Token类型，但Bearer Token才是RFC建议的标准；MAC Token是在[MAC Access Authentication](https://tools.ietf.org/html/draft-hammer-oauth-v2-mac-token-05)中被定义的，采用Message Authentication Code(MAC)算法来提供完整性校验。

[MAC Access Authentication](https://tools.ietf.org/html/draft-hammer-oauth-v2-mac-token-05) 是一种HTTP认证方案，具体内容可参考： [HTTP Authentication: MAC Access Authentication draft-hammer-oauth-v2-mac-token-05](https://tools.ietf.org/html/draft-hammer-oauth-v2-mac-token-05) 。

分类: [开发基础](https://www.cnblogs.com/XiongMaoMengNan/category/933986.html),[协议/规范](https://www.cnblogs.com/XiongMaoMengNan/category/981936.html)

标签: [Bearer Token](https://www.cnblogs.com/XiongMaoMengNan/tag/Bearer%20Token/), [TLS](https://www.cnblogs.com/XiongMaoMengNan/tag/TLS/)

[**好文要顶**](javascript:void(0);) [**关注我**](javascript:void(0);) [**收藏该文**](javascript:void(0);) **[https://common.cnblogs.com/images/icon_weibo_24.png](javascript:void(0);)** **[https://common.cnblogs.com/images/wechat.png](javascript:void(0);)**

[https://pic.cnblogs.com/face/sample_face.gif](http://home.cnblogs.com/u/XiongMaoMengNan/)

[熊猫猛男](http://home.cnblogs.com/u/XiongMaoMengNan/)  
[关注 - 0](http://home.cnblogs.com/u/XiongMaoMengNan/followees)  
[粉丝 - 23](http://home.cnblogs.com/u/XiongMaoMengNan/followers)

[+加关注](javascript:void(0);)

6

0

[«](https://www.cnblogs.com/XiongMaoMengNan/p/6671206.html)上一篇：[HTTP认证模式：Basic and Digest Access Authentication](https://www.cnblogs.com/XiongMaoMengNan/p/6671206.html)  
[»](https://www.cnblogs.com/XiongMaoMengNan/p/6768535.html)下一篇：[OAuth 2.0 / RCF6749 协议解读](https://www.cnblogs.com/XiongMaoMengNan/p/6768535.html)

posted @ 2017-05-07 15:52 [熊猫猛男](https://www.cnblogs.com/XiongMaoMengNan/) 阅读(17975) 评论(4) [编辑](https://i.cnblogs.com/EditPosts.aspx?postid=6785155) [收藏](https://www.cnblogs.com/XiongMaoMengNan/p/6785155.html)

**评论列表**

[#1楼](https://www.cnblogs.com/XiongMaoMengNan/p/6785155.html#3687163) 2017-05-08 23:08 [心中呈和](https://www.cnblogs.com/atsats/) 

赞！  
为您的无私奉献点赞！  
  
(看到博客中大量的文章被阅读成百上千次，但竟然无人评论！)  
  
呼吁大家践行 "开放、平等、协作、快速、分享"的互联网精神！

[支持(1)反对(0)](javascript:void(0);)

[#2楼](https://www.cnblogs.com/XiongMaoMengNan/p/6785155.html#3806641) 2017-10-10 19:25 [匠心十年](https://www.cnblogs.com/gaochundong/) 

nice

[支持(0)反对(0)](javascript:void(0);)

[#3楼](https://www.cnblogs.com/XiongMaoMengNan/p/6785155.html#3954714) 2018-04-18 14:44 [yyyyx方](https://www.cnblogs.com/yuxin5156/) 

学习了

[支持(0)反对(0)](javascript:void(0);)

[#4楼](https://www.cnblogs.com/XiongMaoMengNan/p/6785155.html#4006648) 2018-06-27 10:23 [Ryan.Miao](https://www.cnblogs.com/woshimrf/) 

给猛男点赞