随笔-45 文章-0 评论-0

cookie、session与token

一、详述概念

1、Cookie机制

cookie机制是采用在客户端保持状态的方案（cookie的作用就是为了解决HTTP协议无状态的缺陷所作的努力）。cookie的使用是由浏览器按照一定的原则在后台自动发送给服务器的。浏览器检查所有存储的cookie，如果某个cookie所声明的作用范围大于等于将要请求的资源所在的位置，则把该cookie附在请求资源的HTTP请求头上发送给服务器。

cookie的内容主要包括：名字、值、过期时间、路径和域。

路径与域一起构成cookie的作用范围。

若不设置过期时间，则表示这个cookie的生命期为浏览器会话期间，关闭浏览器窗口，cookie就消失。这种生命期为浏览器会话期的cookie被称为会话cookie.会话cookie一般不存储在硬盘上而是保存在内存里，当然这种行为 并不是规范规定的。

若设置了过期时间，浏览器就会把cookie保存在硬盘上，关闭后再次打开浏览器，这些cookie仍然有效直到超过设定的过期时间。存储在硬盘上的cookie可以在不同的浏览器进程间共享，比如两个IE窗口。而对于保存在内存里cookie，不同的浏览器有不同的处理方式。

2、Session机制

session机制是一种服务器端的机制，服务器使用一种类似于散列表的结构（也可能就是使用散列

随笔档案
2017年3月 (4)
2017年2月 (21)
2017年1月 (9)
2016年12月 (8)
2016年11月 (3)

阅读排行榜
1. 使用nodeJs安装Vue-cli(653)
2. springboot整合freemarker(461)
3. cookie、session与token(382)
4. windows环境，idea的Terminal每次输入git命令都要提示输入用户名，密码(339)
5. 创建一个springboot项目(282)

推荐排行榜
1. 使用nodeJs安装Vue-cli(1)

表）来保存信息。

当程序需要为某个客户端的请求创建一个session时，服务器首先检查这个客户端的请求里是否已包含了一个session标识（称为session id），如果已包含则说明以前已经为此客户端创建过session,服务器就按照session id把这个session检索出来使用（检索不到，会新建一个），如果客户端请求不包含session id，则为此客户端创建一个session并且生成一个与此session相关联的session id，session id的值应该是一个既不会重复，又不容易被找到规律以伪造的字符串，这个session id将被 在本次响应中返回给客户端保存。保存这个session id的方式可以采用cookie，这样在交互过程中浏览器 可以自动的按照规则把这个标识发送给服务器。一般这个cookie的名字就是类似于SESSIONID。

由于cookie可以被人为的禁止，必须有其他机制以便在cookie被禁止时仍然能够把session id传递回服务器。

两种方式：

第一种：URL重写（常用），就是把session id直接附加在URL路径的后面。

第二种：表单隐藏字段（现已很少使用）。就是服务器会自动修改表单，添加一个隐藏字段，以便在表单提交时能够把session id传递回服务器。

3、token

token的意思是“令牌”，是用户身份的验证方式，最简单的token组成:uid(用户唯一的身份标识)、time(当前时间的时间戳)、sign(签名，由token的前几位+盐以哈希算法压缩成一定长的十六进制字符串，可以防止恶意第三方拼接token请求服务器)。还可以把不变的参数也放进token，避免多次查库

二、cookie与session的区别

1、cookie数据存放在客户端上，session数据放在服务器上。

2、cookie不是很安全，别人可以分析存放在本地的COOKIE并进行COOKIE欺骗

考虑到安全应当使用session。

3、session会在一定时间内保存在服务器上。当访问增多，会比较占用你服务器的性能

考虑到减轻服务器性能方面，应当使用COOKIE。

4、单个cookie保存的数据不能超过4K，很多浏览器都限制一个站点最多保存20个cookie。

5、所以个人建议：

将登陆信息等重要信息存放为SESSION

其他信息如果需要保留，可以放在COOKIE中

三、session与token的区别

session 和 oauth token并不矛盾，作为身份认证 token安全性比session好，因为每个请求都有签名还能防止监听以及重放攻击，而session就必须靠链路层来保障通讯安全了。如上所说，如果你需要实现有状态的会话，仍然可以增加session来在服务器端保存一些状态

App通常用restful api跟server打交道。Rest是stateless的，也就是app不需要像browser那样用cookie来保存session,因此用session token来标示自己就够了，session/state由api server的逻辑处理。 如果你的后端不是stateless的rest api, 那么你可能需要在app里保存session.可以在app里嵌入webkit, 用一个隐藏的browser来管理cookie session.

Session 是一种HTTP存储机制，目的是为无状态的HTTP提供的持久机制。所谓Session 认证只是简单的把User 信息存储到Session 里，因为SID 的不可预测性，暂且认为是安全的。这是一种认证手段。 而Token ，如果指的是OAuth Token 或类似的机制的话，提供的是 认证 和 授权 ，认证是针对用户，授权是针对App 。其目的是让 某App有权利访问 某用户 的信息。这里的Token是唯一的。不可以转移到其它 App上，也不可以转到其它 用户 上。 转过来说Session 。

Session只提供一种简单的认证，即有此 SID，即认为有此 User的全部权利。是需要严格保密的，这个数据应该只保存在站方，不应该共享给其它网站或者第三方App。 所以简单来说，如果你的用户数据可能需要和第三方共享，或者允许第三方调用 API 接口，用 Token 。如果永远只是自己的网站，自己的 App，用什么就无所谓了。

四、打破误解：

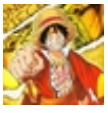
"只要关闭浏览器 ，session就消失了"?

不对。对session来说，除非程序通知服务器删除一个session，否则服务器会一直保留，程序一般都是在用户做log off的时候发个指令去删除session。

然而浏览器从来不会主动在关闭之前通知服务器它将要关闭，因此服务器根本不会有机会知道浏览器已经关闭，之所以会有这种错觉，是大部分session机制都使用会话cookie来保存session id，而关闭浏览器后这个session id就消失了，再次连接服务器时也就无法找到原来的session。如果服务器设置的cookie被保存在硬盘上，或者使用某种手段改写浏览器发出的HTTP请求头，把原来的session id发送给服务器，则再次打开浏览器仍然能够打开原来的session.

恰恰是由于关闭浏览器不会导致session被删除，迫使服务器为session设置了一个失效时间，当距离客户端上一次使用session的时间超过这个失效时间时，服务器就可以以为客户端已经停止了活动，才会把session删除以节省存储空间。

标签: session, token, cookie



张楠楠
关注 - 2
粉丝 - 0
+加关注

0 0

« 上一篇：基于 Token 的身份验证方法
» 下一篇：spring使用ehcache
posted @ 2017-01-06 19:48 张楠楠 阅读(382)
评论(0) 编辑 收藏

刷新评论 刷新页面 返回顶部

注册用户登录后才能发表评论，请 登录 或 注册，访问网站首页。