自：https://blog.csdn.net/chunqiuwei/article/details/23461995

**jsessionid的简单说明**

2014年04月11日 17:41:50 [chunqiuwei](https://me.csdn.net/chunqiuwei) 阅读数：41296更多

个人分类： [SSH](https://blog.csdn.net/chunqiuwei/article/category/1138882)

版权声明：本文为博主原创文章，未经博主允许不得转载。 https://blog.csdn.net/chunqiuwei/article/details/23461995

1）第一次访问服务器的时候，会在响应头里面看到Set-Cookie信息（只有在首次访问服务器的时候才会在响应头中出现该信息）。



上面的图JSESSIONID=ghco9xdnaco31gmafukxchph;Path=/acr，

浏览器会根据响应头的set-cookie信息设置浏览器的cookie并保存之

注意此cookie由于没有设置cookie有效日期，所以在关闭浏览器的情况下会丢失掉这个cookie。

2）当再次请求的时候（非首次请求），浏览器会在请求头里将cookie发送给服务器(每次请求都是这样)  


（JSESSIONID=ghco9xdnaco31gmafukxchph）

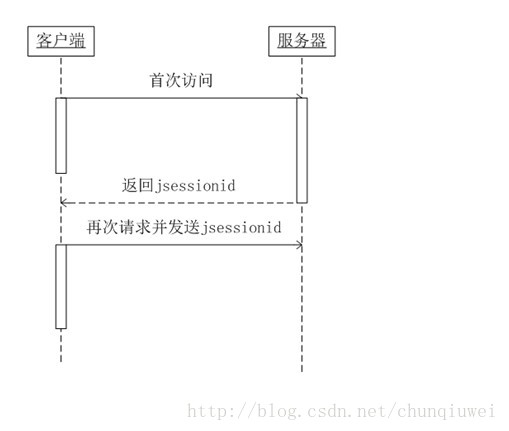
不难发现这个的jsessionid和上面的jsessionid是一样的

3）为什么除了首次请求之外每次请求都会发送这个cookie呢（在这里确切地说是发送这个jsessionid）？

事实上当用户访问服务器的时候会为每一个用户开启一个session，浏览器是怎么判断这个session到底是属于哪个用户呢？jsessionid的作用就体现出来了：jsessionid就是用来判断当前用户对应于哪个session。换句话说服务器识别session的方法是通过jsessionid来告诉服务器该客户端的session在内存的什么地方。

事实上jsessionid ==request.getSession().getId()

4)总结，jsessionid的工作流程可以简单用下面的图表示：



4.补充说明一下，http请求是无状态的，每一次请求都是独立的。

（1）.如果本次请求在服务器端没有任何session相关代码执行到，服务器端返回的response参数中是不会返回Set-cookie属性的，原因很简单，本次请求服务器根本就不关心session，也就是说与session无关的请求，服务器为了性能，智能忽略对该参数的响应。

（2）.如果本次请求服务器端有执行到session相关的代码。则区分是否为第一次执行到，本质上是看根据request传过来的cookie中jseesionId值是否拿到了对应的session对象（如果第一次访问，甚至都是没有cookie的，即没有jsessionId，那么当然也是取不到对应的session对象的）。

如果没拿到，则说明此jsessionId值已经过期（比如是session过期造成的jsessionId过期）或者是伪造的，那么会创建一个新的jsessionId值放入set-cookie中返回给浏览器，这就是上文中所谓的“第一次”会返回Set-cookie属性。

如果拿到了，则说明此jsessionId值对应的session对象在服务器端是存在的，还是有效的，处于服务器端所谓的会话中，就不会再返回新的Set-cookie属性，再返回也是浪费，浏览器自然在后续请求中还会继续沿用这个jsessionId值。

直到cookie过期，或者session过期，或者手动清除cookie，服务器服务关闭，这些情况下，浏览器发送请求到服务器端，一旦服务器端执行到了读取session相关的代码，就会在本次请求返回set-cookie值，这就又回到了刚才说的过程。

System.*out*.println("session:"+request.getSession());