

今天看到一篇 4 个月前的专栏，标题十分吓人

[99%的人都理解错了HTTP中GET与POST的区别](#)

文章前半部分讲了一些大家都知道的 GET vs POST 的区别，重点在后面。

这位BOSS有多神秘？当你试图在网上找“GET和POST的区别”的时候，那些你会看到的搜索结果里，从没有提到他。他究竟是什么呢。。。

GET和POST还有一个重大区别，简单的说：

GET产生一个TCP数据包；POST产生两个TCP数据包。

长的说：

对于GET方式的请求，浏览器会把http header和data一并发送出去，服务器响应200（返回数据）；

而对于POST，浏览器先发送header，服务器响应100 continue，浏览器再发送data，服务器响应200 ok（返回数据）。

然而我此前从来没听说过这回事，于是搜索了一下。结论是，文章作者关于『GET 发一个包，POST 发两个包』的知识 99% 是从下面这篇文章中得来的

XMLHttpRequest (XHR) Uses Multiple Packets for HTTP POST?

为了验证结论是否仍然成立，我做了测试，结果如下：

- Chrome 55.0.2883.95, two packets: **YES**
- Safari 10.0.2, two packets: **YES**
- Firefox 49.0, two packets: **NO**

看来的确有这回事，参见 RFC 中的描述：[Hypertext Transfer Protocol \(HTTP/1.1\): Semantics and Content](#)。

首先，感谢专栏作者让大家对 POST 多了一些了解。但如果仅仅是这样我显然也不会专门写一篇文章了。让我感到不爽的地方有三点：

一. 作者没有验证过自己写的东西

他在专栏中说：

而对于POST，浏览器先发送header，服务器响应100 continue，浏览器再发送data，服务器响应200 ok（返回数据）。

这个结论是错误的。

首先，在那篇英文文章中并没有提及这一点。其次，我在测试时也没有看到服务器响应 100 continue。RFC 里讲的很明白：100 continue 只有在请求里带了

```
Expect: 100-continue
```

header 的时候才有意义。

When the request contains an Expect header field that includes a 100-continue expectation, the 100 response indicates that the server wishes to receive the request payload body, as described in Section 5.1.1. The client ought to continue sending the request and discard the 100 response.

If the request did not contain an Expect header field containing the 100-continue expectation, the client can simply discard this interim response.

而实际上，不论哪一种浏览器，在发送 POST 的时候都没有带 Expect 头，server 也自然不会发 100 continue。通过抓包发现，尽管会分两次，body 就是紧随在 header 后面发送的，根本不存在『等待服务器响应』这一说。

二. 作者并不理解自己写的东西

如果我们去看那篇英文原文，或者只看标题：『[XMLHttpRequest \(XHR\) Uses Multiple Packets for HTTP POST?](#)』，就会发现，它讨论的一直是 XMLHttpRequest，也就是 AJAX 请求，而专栏中却把 XMLHttpRequest 写成了 POST。

我想但凡对 Web 有基本了解的人都不会认为两者是一个东西吧。

三. 强行标题党

写文章起个吸引眼球的标题，这没什么，但问题是，你这个标题和文章说的不是一件事啊！

我们通常在讨论 GET vs POST 的时候，实际上讨论的是 specification，而不是 implementation。什么是 specification？说白了就是相关的 RFC。implementation 则是所有实现了 specification 中描述的代码/库/产品，比如 curl，Python 的 requests 库，或者 Chrome。

POST 请求怎么发送，根本就不是这段 RFC 在讨论的事情。RFC 中只说明了 100 continue 和 Expect header 的联系，比如你想在 GET 请求里带 body，一样可以发送 Expect: 100-continue 并等待 100 continue，这是符合标准的。

也就是说，『XHR 发送两个 TCP packets』是关于 implementation 的知识，而不是关于 specification 的知识。你不能说『Chrome 在 AJAX POST 的时候会发两个 TCP packets，GET 只会发一个』是 GET 和 POST 的区别，正如你不能因为北京 PM 2.5 经常爆表就说国家关于工业废气排放的标准有问题。

从另一个角度说，TCP 是传输层协议。别人问你应用层协议里的 GET 和 POST 有啥区别，你回答说这俩在传输层上发送数据的时候不一样，确定面试官不抽你？

所以，不是 99% 的人理解错了 GET 和 POST，而是有 99% 的可能作者不知道自己在写什么。

PS：写完之后发现专栏文章是从微信转载的.....那么姑且认为我指的都是微信文章的作者吧。但是你帮他转发一遍，你等于你也有责任吧？