

解决百度统计无法获取https来源的referrer

我们都知道，当我们的页面请求一个js文件、一个cs文件或者点击到其他页面，浏览器一般都会给这些请求头加上表示来源的 Referrer 字段。Referrer 在分析用户的来源时非常有用，比如大家熟悉的百度统计里面就利用到 Referrer 信息了。但是遗憾的是，目前百度统计仅仅支持来源于http页面的referrer头信息；也就是说，如果你网站是http的，那么百度统计只能统计到从其他http页面进入到你网站的referrer，而如果是从https页面进入到你的网站，百度统计目前是无法统计到。

在介绍解决办法之前，我们先来了解Referrer Policies。

Referrer Policies

X	Headers	Preview	Response	Cookies	Timing
	AlexaToolbar-ALX_NS_PH: AlexaToolbar/alx-4.0.1				
	Connection: keep-alive				
	Cookie: BDPCIEXP=28; CNZZDATA1254868071=256406192-1477114602-%7C1482235866; bdshare_firsttime=1483104106; 45; bigfa_ding_1957=1957; bigfa_ding_1949=1949; bigfa_ding_1956=1956; bigfa_ding_1947=1947; bigfa_ding_1254=1254; bigfa_ding_1542=1542; bigfa_ding_886=886; bigfa_ding_1150=1150; bigfa_ding_1129=1129; bigfa_ding_1958=1958; wordpress_logged_in_ffb9bc9f338d4e1bae54930a7c0d4e46=w397090770%7C1485076825%7C0kWZvPRRCHuq b3a160074feb0e3fc7a23db9a0a00ccd; bigfa_ding_1887=1887; wp-settings-1=hidetb%3D1%26editor%3Dhtml%26libr 576%26unfold%3D1%26mfold%3Do%26dfw_width%3D822%26posts_list_mode%3Dlist%26post_dfw%3Don; wp-settings-ti 9mi228th4; bigfa_ding_1929=1929; _gat=1; _ga=GA1.2.418379059.1477116596; Hm_lvt_e9e1d61340ff3d9495a8cca 1d61340ff3d9495a8cca04ebdb49d=1484058848				
	Host: www.iteblog.com				
	If-Modified-Since: Sun, 01 Jan 2017 06:32:06 GMT				
	If-None-Match: "5868a266-0"				
	Referer: https://www.iteblog.com/archives/category/flink				
	User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/55.0.				

如果想及时了

解Spark、Hadoop或者Hbase相关的文章，欢迎关注微信公共帐号：iteblog_hadoop

目前存在九种Referrer Policies，包括：空字符串、no-referrer、no-referrer-when-downgrade、same-origin、origin、strict-origin、origin-when-cross-origin、strict-origin-when-cross-origin 以及 unsafe-url：

```
enum ReferrerPolicy {  
    "",  
    "no-referrer",  
    "no-referrer-when-downgrade",  
    "same-origin",  
    "origin",  
    "strict-origin",  
    "origin-when-cross-origin",  
    "strict-origin-when-cross-origin",  
}
```

```
"unsafe-url"  
};
```

我们这对每种Referrer Policies进行介绍。

空字符串

空字符串表示没有referrer策略。比如我们有一个HTML标签元素，而且并没有声明任何的Referrer Policies，这时候的Referrer Policies就是空字符串。默认的空字符串Referrer Policies就等同于no-referrer-when-downgrade。

no-referrer

最简单的Referrer Policies就是 no-referrer；这种Referrer Policies会导致所有的请求都不带referrer信息。比如如果 <https://www.iteblog.com/archives/1929> 页面将Referrer Policies设置成 no-referrer，那么从这个页面转到 <https://www.iteblog.com/>（或者其他任何页面）都不会发送Referer头信息。

no-referrer-when-downgrade

这种策略在从受TLS保护的URL跳转到任何潜在可信的URL以及从不受TLS保护的URL（比如HTTP）跳转到任何URL都会发送完整的URL Referrer 信息。

但是如果从受TLS保护的URL（比如HTTPS）跳转不受信任的URL（比如HTTP），那么Referrer 头将不会带任何的信息。这是大多数浏览器默认的Referrer Policies。

比如：假设 <https://www.iteblog.com/archives/1929> 页面的Referrer Policies设置成 no-referrer-when-downgrade，那么从这个页面跳转到任何的 https 页面将会带上值为<https://www.iteblog.com/archives/1929>的Referer头信息；但是如果从那个页面跳转到任何 HTTP 页面，将不会带上任何的Referer头信息。这就是为什么百度统计无法获取到从 HTTPS 跳转到 HTTP 的信息。

same-origin

对于同源的链接，会发送referrer，其他的不会。同源意味着域名需要相同，iteblog.com 和 idea.iteblog.com 是非同源的。

origin

这种Referrer Policies对于任何资源（包括可信和不可信）来说只发送网站的根域名，不发送完整的url。比如：<https://www.iteblog.com/archives/1929> 页面的Referrer Policies设置成origin，那么从这个页面跳转到任何网站将会带上值为<https://www.iteblog.com/>的Referer头信息。

strict-origin

我们可以把他看做是 origin 和 no-referrer-when-downgrade 的结合体。也就是说，如果从受TLS保护的URL跳转到任何潜在可信的URL以及从不受TLS保护的URL（比如HTTP）跳转到任何URL都会发送根域名的 Referrer 头信息。

但是如果从受TLS保护的URL（比如HTTPS）跳转不受信任的URL（比如HTTP），那么Referrer 头将不会带任何的信息。比如<https://www.iteblog.com/archives/1929> 页面的Referrer Policies设置成 strict-origin，那么从它跳转到 <https://xxx.iteblog.com> 将会带上值为<https://www.iteblog.com/>的Referer头信息；如果从那个页面跳转到 <http://idea.iteblog.com> 将不会带上任何的Referer头信息。

再比如：<http://idea.iteblog.com>页面的Referrer Policies设置成 strict-origin，那么从这个页面跳转到 <http://books.iteblog.com> 或者 <https://www.iteblog.com/archives/1929> 将会带上值为<http://idea.iteblog.com/>的Referer头信息。

不过目前这种Referrer Policies浏览器可能不支持。

origin-when-cross-origin

该策略在同源的链接中发送完整的URL，其他情况仅发送根域名。需要注意的是<https://www.iteblog.com> 和 被认为是非同源的。

比如 <https://www.iteblog.com/archives/1929> 页面的Referrer Policies设置成 origin-when-cross-origin，那么从它跳转到 <https://www.iteblog.com/archives/1464> 将会带上值为<https://www.iteblog.com/archives/1929>的Referer头信息；如果从那个页面跳转到 <http://idea.iteblog.com/> 页面（注意这里HTTP和HTTPS页面都可以），将会带上值为<https://www.iteblog.com/>的Referer头信息。

不过这种Referrer Policies在Chrome实现的时候被误写成了 origin-when-crossorigin。

strict-origin-when-cross-origin

这种策略可以看做是 origin-when-cross-origin 和 no-referrer-when-downgrade 的结合体。对于同源请求，发送完整的URL；对于同为https的页面，或者是从HTTP页面跳转到任何的页面（包括HTTP和HTTPS）只发送根域名；如果是从https页面跳转到http将不发送referrer。

If a document at <https://example.com/page.html> sets a policy of "strict-origin-when-cross-origin", then navigations to <https://example.com/not-page.html> would send a

Referer header with a value of https://example.com/page.html.

Navigations from that same page to https://not.example.com/ would send a Referer header with a value of https://example.com/.

Navigations from that same page to http://not.example.com/ would send no Referer header.

unsafe-url

无论是否发生协议降级（HTTPS到HTTP），无论是本站链接还是站外链接，统统都发送 Referrer 信息。这是最宽松而最不安全的策略；

比如：https://www.iteblog.com/archives/1929 页面的Referrer Policies设置成 unsafe-url，那么从它跳转到http://idea.iteblog.com/ 页面（注意这里HTTP和HTTPS页面都可以），将会带上值为https://www.iteblog.com/archives/1929的Referer头信息。

如何设置

页面是设置Referrer Policies主要有以下几种

CSP 响应头

CSP（Content Security Policy），是一个跟页面内容安全有关的规范。在 HTTP 中通过响应头中的 Content-Security-Policy 字段来告诉浏览器当前页面要使用何种 CSP 策略。：

Content-Security-Policy: referrer

我们还可以在 里面设置：

```
<meta http-equiv="Content-Security-Policy" content="referrer">
```

<meta> 标签

通过 标签也可以指定 Referrer 策略，同样很简单：

```
<meta name="referrer" content="no-referrer">
```

我们把<meta> 放在 <head> ...</head>

之间。注意，如果出现的位置不对会被忽略。同样，如果没有给它定义 content 属性，或者 content 属性为空，也会被忽略。如果 content 属性不是合法的取值，浏览器会自动选择 no-referrer 这种最严格的策略。

<a> 标签的 referrer 属性

通过给 <a> 标签增加 referrer 属性也可以指定 Referrer 策略，格式如下：

```
<a href="https://www.iteblog.com" referrer="no-referrer">过往记忆</a>
```

这种方式作用的只是这一个链接。并且，<a> 标签可用的 Referrer 策略只有三种：no-referrer、origin 以及 unsafe-url。另外，这样针对单个链接设置的策略优先级比 CSP 和 要高。

解决百度统计无法获取https页的referrer

说到这里，如果我们想让百度统计能够统计到来源HTTPS的页面，我们有两种方式：（1）、将自己的网站也升级到HTTPS（参考 [《在Nginx中使用Let's Encrypt免费证书配置HTTPS》](#)）；（2）、让对方网站设置相关的Referrer Policies。

本博客文章除特别声明，全部都是原创！
原创文章版权归过往记忆大数据（[过往记忆](#)）所有，未经许可不得转载。
本文链接: [【】（）](#)