ajax跨域问题

标签 跨域 同源策略 jsonp 文件代理 hnz-2015-11-25

> 跨域：一个域名下的文件去请求另一个跟他不一样的域名下的资源文件，那么就会产生跨域请求。

什么是跨域？

跨域访问

简单来说就是 A 网站的 javascript 代码试图访问 B 网站，包括提交内容和获取内容。由于安全原因，跨域访问是被各大浏览器所默认禁止的。

在广域网环境中，由于浏览器的安全限制，网络连接的跨域访问时不被允许的，XmlHttpRequest也不例外。但有时候跨域访问资源是必需的。

跨域：通俗来讲，就是一个域名下的文件去请求另一个跟他不一样的域名下的资源文件，那么就会产生跨域请求。
跨域概念：只要协议、域名、端口有任何一个不同，都被当作是不同的域。

- 只有后台解决：对于端口和协议的不同，只能通过后台来解决。
- 前台解决：域名不同

同源策略

浏览器的“**同源策略(SOP:Same Origin Policy)**”。 简而言之，就是浏览器限制脚本程序只能和同协议、同域名、同端口的脚本进行交互，这包括共享和传递变量等。

URL	说明	是否允许通信
http://www.a.com/a.js http://www.a.com/b.js	同一域名下	允许
http://www.a.com/lab/a.js http://www.a.com/script/b.js	同一域名下不同文件夹	允许
http://www.a.com:8000/a.js	同一域名，不同端口	不允许

http://www.a.com/b.js		
http://www.a.com/a.js https://www.a.com/b.js	同一域名，不同协议	不允许
http://www.a.com/a.js http://70.32.92.74/b.js	域名和域名对应ip	不允许
http://www.a.com/a.js http://script.a.com/b.js	主域相同，子域不同	不允许
http://www.a.com/a.js http://a.com/b.js	同一域名，不同二级域名（同上）	不允许（cookie这种情况下也不允许访问）
http://www.cnblogs.com/a.js http://www.a.com/b.js	不同域名	不允许

解决跨域问题

浏览器因为安全问题，不允许跨域请求资源，存在跨域限制问题，因此我们需要寻找解决这个问题的办法：

- **服务器上设置代理页面 (php jsp asp 来获取资源，再访问服务端文件) || 或叫中间层过渡**
- **script标签 (JSONP Json with Padding)**

JSONP是什么？

JSONP(JSON with Padding)是JSON的一种“使用模式”，可用于解决主流浏览器的跨域数据访问的问题。由于同源策略，一般来说位于 www.baidu.com 的网页无法与www.google.com的服务器沟通，而 HTML 的<.script> 元素是一个例外。利用 <.script> 元素的这个开放策略，网页可以得到从其他来源动态产生的 JSON数据，而这种使用模式就是所谓的 JSONP。

JSONP是一种依靠开发人员的聪明才智创造出的一种非官方跨域数据交互协议。
JSONP由三部分组成：**script标签对**、**回调函数**和**数据**。

- script标签对：对指定URL+参数(callback) 进行访问。
- 回调函数：当响应到来时应该在页面中调用的函数。
- 数据：就是传入回调函数中的JSON数据。

JSONP核心原理

1. script标签

- script标签可以它src属性加载资源，没有跨域问题。
  如CDN公共库：

```
<script src="http://libs.baidu.com/jquery/1.9.0/jquery.js">
</script>
```

- 实现
  因此我们就可以通过script标签对访问我们需要获取数据的URL：

```
<script src="http://api.map.baidu.com/telematics/v3/weather?
location=beijing&output=json&ak=cW96ILwOzpVmoZcWl1vL2T4W&callback=
fn"></script>
```

客户端在对JSON文件调用成功之后，也就获得了自己所需的数据，剩下的就是按照自己需求进行处理和展现了，这种获取远程数据的方式看起来非常像AJAX，但其实并不一样。

2. JSONP获得数据如何使用？

获取到数据

```
<script src="http://baidu.com/action.php?param=123">
    //加载到的资源
    //{name:'zhangsan',age:12};
</script>
```

返回数据如何使用？没有名字。

3.JSONP解决办法

- 前端：
  前台通过**script标签**去获取数据，必须在获取数据标签之前添加另一个script标签对并在其中**添加一个函数并接收json数据。**

```
<script>
//3.在此定义函数fn接受参数即可获取跨域服务器返回数据，再做处理
function fn(data){
    alert(data)
}
</script>
<script src="http://baidu.com/action.php?param=123&callback=fn">
//fn({name:'zhangsan',age:12})
//1. 如果访问的服务器返回函数调用fn({name:'zhangsan',age:12})
</script>
```

- php服务器：
  后台返回数据格式为 **函数名(json)** 函数名由前台参数决定

```
<?php
    $data=Array(name:'zhangsan',age:12);
    echo $_GET["callback"]."(".json_encode($data).");";
    //2. fn({name:'zhangsan',age:12})
?>
```

通常我们会通过事件去获取内容，但是我们通过script标签当执行到它时就会获取数据，不符合我们需要数据再获取这个思想。

4. 按需获取解决办法

我们在这里再次改变一下策略：

```
function fn(){
    alert(1);
}
fn()    //当我们调用函数的时候,它就会执行
```

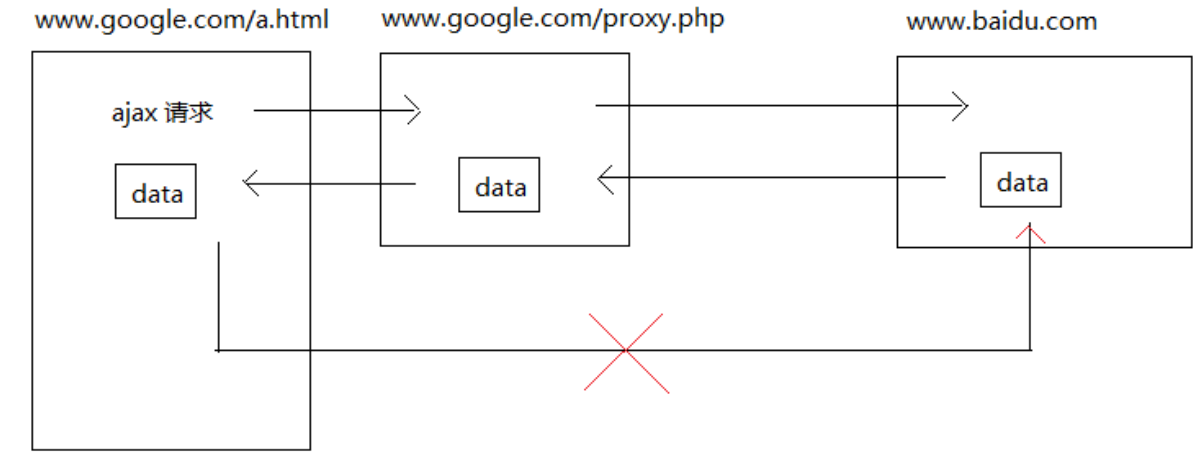
通过看上方代码，我们可以想到，我们可以在需要获取的时候再调用就可以，我们可以通过需要时再创建script标签再添加src 就可以实现

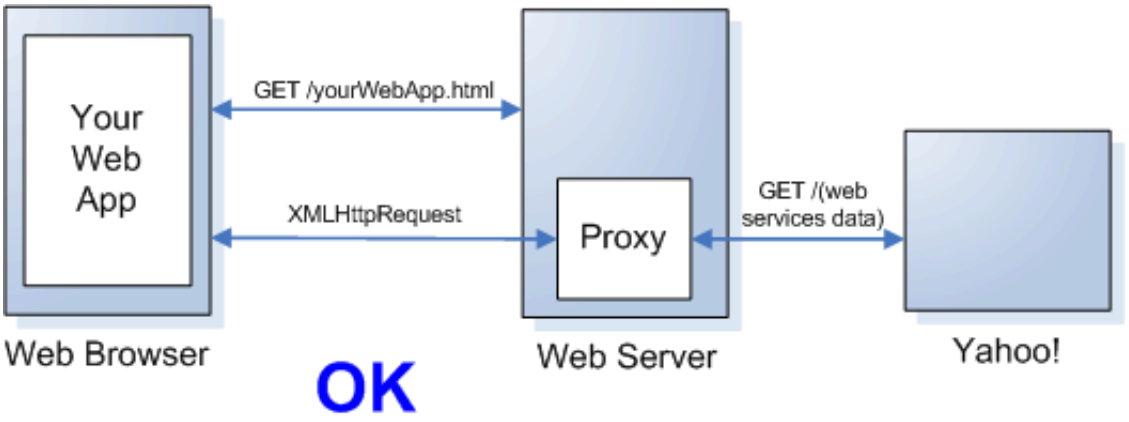
```
btn.onclick=function(){
    var spt=document.createElement('script');
    spt.src='http://baidu.com/action.php?param=123&callback=fn';
    document.getElementsByTagName('head')[0].appendChild(spt);
    //以上代码等同于   fn()
}
```

使用jsonp注意事项：

1. 服务端返回格式必须为 **函数名(数据)**
2. 对于jsonp跨站访问，带来资源别越权调用漏洞使用(任何人都可以访问服务器返回数据) 如：商城订单信息 被攻击者调用
3. 使用jsonp服务器端要过滤callback传递的字符串 避免出现 ” <> ()’”;. {} “字符串。推荐做法：用白名单方法放行允许的字符。原因：字符串太多了。匹配原则：**字符+数字+下划线+点** 才符合要求

代理页面解决跨域问题





- PHP代理之 GET

```php
<?php
    $searchUrl = 'http://www.baidu.com/search.php?content=';
    if(!empty($_GET['content'])){
        $searchUrl .= $_GET['content'];
    }
    echo file_get_contents($searchUrl);
?>
```

- PHP代理之POST

跨域访问测试

名称	网址	类型
百度开放平台	http://developer.baidu.com/	jsonp
百度APIStore	http://apistore.baidu.com/	proxy