

# 打破ajax 跨域问题,游刃有余的解决方案,大神就是这样操作的



微信公众号:编程集合营,领取2020最新前端零基础教程

关注他

1人赞同了该文章

# 什么是 ajax 跨域

赞同 1

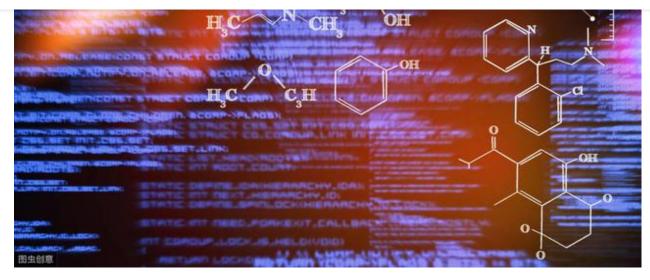
分享

主流的前后端分离模式下,当前端调用后台接口时,由于是在非同一个域下的请求,从而会引发浏 **览器**的自我安全保护机制,最终结果是**接口成功请求并响应**,但**前端不能正常处理该返回数据**。

因此, 当同时满足以下三个条件的情况下, 就会出现跨域问题:

- 1. 浏览器限制
- 2. 非同源请求 (跨域)
- 3. 发送的是 XHR (XMLHttpRequest)请求

## 解决方案



想要彻底解决跨域问题,只需要破坏以上三个条件的任一即可:

1. 修改浏览器 (不推荐)

添加浏览器启动参数: chrome --disable-web-security, 但是极不推荐这种解决方式。

2. JSONP请求 (**不常用**)

Jsonp,全称 JSON with Padding,一种非官方的协议,而是一种约定;前端通过向后台发送 script 类型请求解决跨域,此时接口响应的 application/javascript 类型的数据会作为 callback 函数的参数进行处理。

所以,后台也需要做相应的处理。以 Java 为例,添加如下配置即可:

```
@ControllerAdvice
public class JsonpAdvice extends AbstractJsonpResponseBodyAdvice {
  public JsonpAdvice() {
    // 前后端约定的jsonp参数名,默认值是callback
    super("callback");
  }
}
```

注意, Spring 4.1 版本之后, 官方已不再推荐使用上述允许 jsonp 请求的配置, 建议使用 CROS 配置来解决跨域问题, 详情可查看这里

赞同 1

综上, jsonp 请求存在以下几个弊端:

**才** 分享

- 1. 服务端需要改动代码进行支持;
- 2. 只支持发送 Get 请求,请求头中更改其它类型的请求方式是无效的;
- 3. 发送的不是 XHR 请求,而是 script 类型,无法享受到相关的特性。
- 3. 调用方隐藏跨域

用 Nginx 或 Apache 来代理调用方的请求(**客户端变更为相对路径请求,而非绝对路径**),此时对于浏览器来说,由于请求是同源的,因此就不存在跨域问题。

- 4. 被调用方允许跨域(**最常用**)
- 服务端配置
- 以 Java 应用为例,添加如下全局配置:

```
@Configuration
public class CorsConfig implements WebMvcConfigurer {
    @Override
```

```
.allowedHeaders("*") // 允许跨域的请求头
.allowCredentials(true) // 带cookie请求的时候需要开启,且allowedOrigins需要指定为具体的请求
.maxAge(60 * 60 * 24); // 设定options请求预检命令的缓存时长
}
```

如果只想针对某个类下的接口,或者是某个具体的接口配置允许跨域,只需要在相应的地方添加注解 @CrossOrigin 即可。

• Nginx 配置

如果配置了 nginx 作为代理服务器,那么只需要为 nginx 添加支持跨域请求即可:

```
server {
listen 80;
server_name xxx.com;
location / {
proxy_pass http://localhost:8080/;
# 配置允许跨域
add_header Access-Control-Allow-Origin $http_origin;
add_header Access-Control-Allow-Methods *;
add_header Access-Control-Allow-Headers $http_access_control_request_headers;
add_header Access-Control-Max-Age 3600;
add_header Access-Control-Allow-Credentials true;
# 对于options预检请求,直接响应200
if ($request_method = OPTIONS) {
return 200;
}
}
}
```

## 扩展思考



Q1: 浏览器在执行跨域请求时,是先执行后判断,还是先判断后执行?

A1:都有可能,这需要根据所发送的请求是**简单请求**还是**非简单请求**来判断;如果是非简单请

我一浏览现有为大批仁古正的注张之前,这个生长光一个 antions 注张七十的延龄令人【 **司还**中

Q2: 如果是允许带 (**被调用方**) cookie 的跨域请求,此时服务端同样配置为 Access-Control-Allow-Origin 等于 \*, 前端是否还可以请求成功?

A2:不可以,此时要将 Access-Control-Allow-Origin 指定为**调用方**具体的域【 **可以先取得调用方的域再动态配置,这样就不存在多个域请求的限制问题** 】,并且添加配置 Access-Control-Allow-Credentials 为 true。

喜欢小编的可以点个赞关注小编哦,小编每天都会给大家分享文章。

零基础学大前端



发布于 2019-11-28

「真诚赞赏,手留余香」

赞赏

还没有人赞赏, 快来当第一个赞赏的人吧!

Ajax JSONP 跨域

#### 文章被以下专栏收录



零基础学大前端

前端学习群950919261

关注专栏



web前端学习圈

web前端学习方法/知识干货/实战案例等,每天更新

关注专栏



程序员

没有对象能编程吗?不!因为没有对象无法面向对象编程

关注专栏

#### 推荐阅读



## vue跨域方案指北

萌萌哒草头将军

## 科普一下 CORS 以及如何节省 一次 OPTIONS 请求

相信做前端开发的同学对同源策略都比较熟悉,而如何解决跨域问题基本上也是前端面试必考题之一。为了解决跨域资源共享问题,浏览器厂商和标准组织在 HTTP 协议的基础上,提出了 CORS 标准…

雨打梨花

## )<del>(</del>(0)

跨域产生的/ 析与解决方:

这一次彻底搞懂1

跨域 (COR 解决方案,;

万月君