

跨域是前端开发中经常会遇到的问题，前端调用后台服务时，通常会遇到 No 'Access-Control-Allow-Origin' header is present on the requested resource的错误，这是因为浏览器的同源策略拒绝了我们的请求。

所谓同源是指，域名，协议，端口相同，浏览器执行一个脚本时同源的脚本才会被执行。如果非同源，那么在请求数据时，浏览器会在控制台中报一个异常，提示拒绝访问。

这个问题我们通常会使用CORS(跨源资源共享)或者JSONP去解决，这两种方法也是使用较多的方法，网上的文章对于这两种的介绍都很详细了。但对于使用Nginx解决跨域大多写的不太详细，这里主要想介绍一下Nginx解决跨域问题的方法。

Nginx是一个高性能的[HTTP](#)和[反向代理](#)服务器，由俄罗斯的伊戈尔·赛索耶夫开发，其特点是占有内存少，并发能力强。Nginx可以在[这里](#)下载到，解压后双击nginx.exe即可运行。

负载均衡

提到Nginx这里就想多讲一下它的负载均衡，Nginx可以将请求转发均匀转发至多台服务器上，通过部署多台相同服务的服务器，可以减轻服务器的压力。

首先我使用SpringBoot提供了一个服务A，可以通过访问localhost:8080/test/hello访问

```
C:\Users\niuzhenhao>curl localhost:8080/test/hello
hello
```

我在起另一个服务，提供一个页面，可以通过ajax访问到服务A的/test/hello

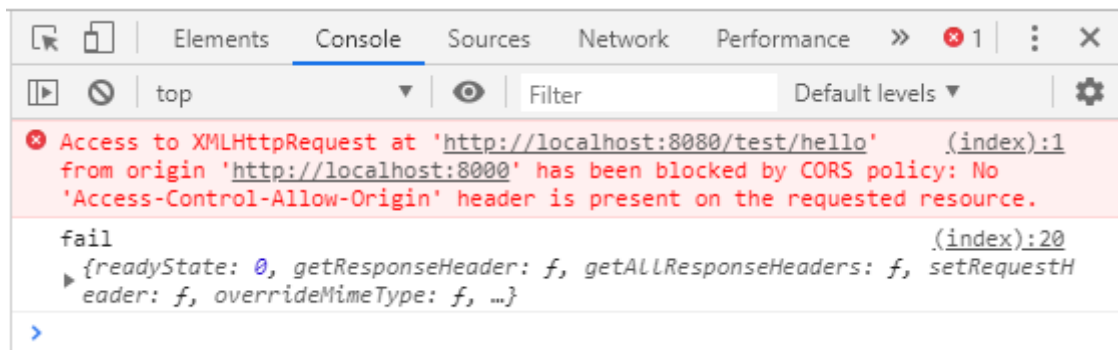
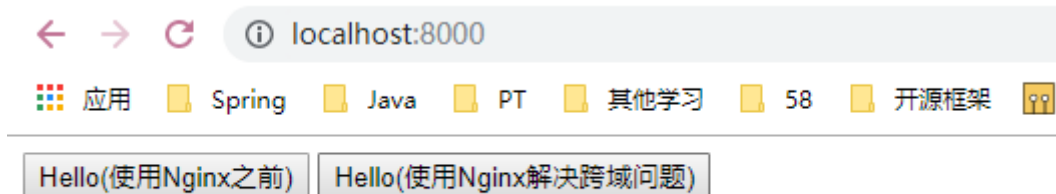
```
<html xmlns="http://www.w3.org/1999/xhtml" xmlns:th="http://www.thymeleaf.org">
  <head>
    <script src="https://apps.bdimg.com/libs/jquery/2.1.4/jquery.min.js">
    </script>
  </head>
  <meta charset="UTF-8" content="text/html">
  <body>
    <button id="hello-before">Hello(使用Nginx之前)</button>
    <button id="hello-after">Hello(使用Nginx解决跨域问题)</button>
  </body>
  <script>
    $(document).ready(function () {
      $('#hello-before').click(function () {
        $.ajax({
          url: 'http://localhost:8080/test/hello',
          success: function(res) {
            console.log("success", res)
          },
          error: function(err) {
            console.log('fail', err)
          }
        })
      })
      $('#hello-after').click(function () {
        $.ajax({
          url: 'http://localhost:8001/test/hello',
          success: function(res) {
            console.log("success", res)
          },
          error: function(err) {
```

```

        console.log('fail',err)
    }
    })
    })
    })
    </script>
</html>

```

访问localhost:8000, 进入下面页面, 点击第一个按钮, 就会出现跨域问题, 他直接请求了localhost:8080/test/hello, 此时点击第二个按钮, 因为还没有配置nginx, 请求失败。



那么如何解决呢, 可以通过使用nginx

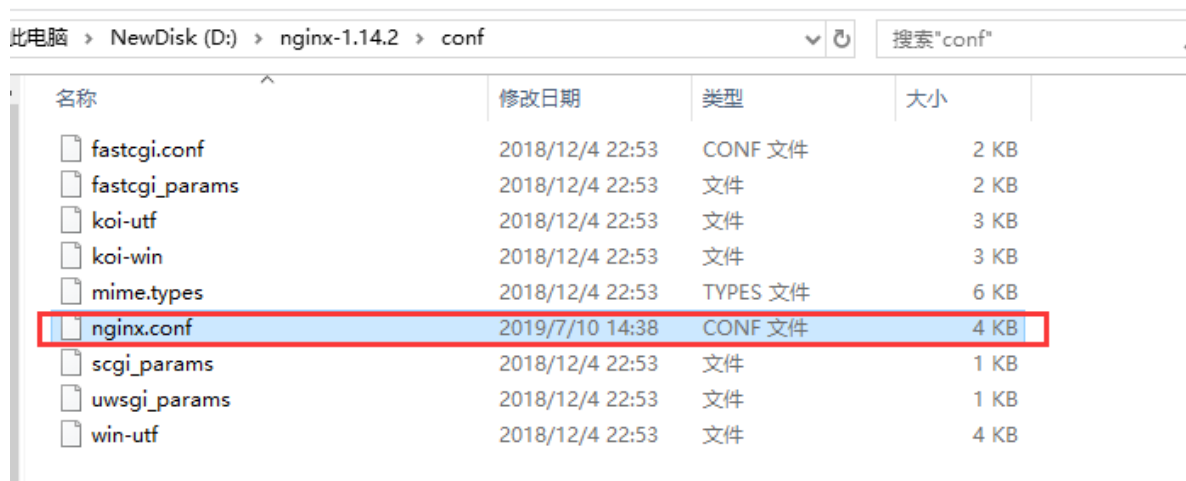
下载安装nginx

[nginx下载地址](#)

- 选择其中一个版本下载, 再解压即可使用
- 在nginx目录下输入nginx -v, 若出现版本号, 则安装成功

配置nginx

进入nginx的conf文件夹, 找到如下入所示的文件进行编辑, 编辑内容如下



```
http {
```

```

include      mime.types;
default_type application/octet-stream;
sendfile     on;
keepalive_timeout 65;

//下面三行很重要，是允许跨域请求的配置
add_header Access-Control-Allow-Origin *;
add_header Access-Control-Allow-Headers X-Requested-With;
add_header Access-Control-Allow-Methods GET,POST,OPTIONS;


server {
    //nginx监听8001端口
    listen      8001;
    server_name localhost;
    //对于所有 localhost:8001/下面的请求，都跳转到 http://localhost:8000
    location / {
        proxy_pass http://localhost:8000;
        root html;
        index index.html index.htm;
    }
    //对于所有 localhost:8001/test/{*}下面的请求，都实际去请求
    http://localhost:8080/test/{*}
    location /test {
        proxy_pass http://localhost:8080;
    }
}

```

如何理解呢，所有的请求对于浏览器来说都是请求了localhost:8001，就不会出现跨域问题，但是nginx其实帮我们分别请求了localhost:8000和localhost:8080，这样就能解决跨域问题了。

开启Nginx

配置完nginx配置文件之后，在nginx目录下使用start nginx即可开启服务。

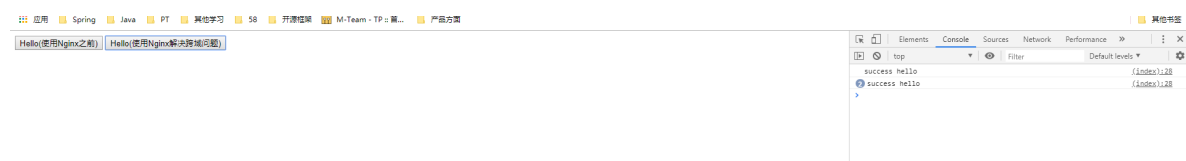
每次修改配置都需要执行nginx -s reload命令才能生效

```

D:\nginx-1.14.2>start nginx
D:\nginx-1.14.2>_

```

实验是否成功



```

C:\Users\niuzhenhao>curl localhost:8001/test/hello
hello

```

可以看到请求成功，解决了跨域问题。

