

koa——跨域 CORS 问题解决、跨域调试

作者: [limanting](#)

原文链接: <https://ld246.com/article/1628912919911>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

在使用koa开发nodejs应用时，碰到跨域问题，即本地进行前端开发时，不能请求已经部署在某个域下的接口。使用koa-cors 和 koa2-cors 发现并没有效果。

也许是我用koa不多，感觉koa不是很好用很完备。比如没发指定当前的基础路径，需要文件结构配，比如使用koa-cors或koa2-cors 发现并没有解决跨域问题——没有设置上允许跨域的源，只能自己写一个cors解决跨域的问题。

这里写一下cors函数的代码

```
const REG_WHITE_LIST = /^https?:\/\/(?:[0-9a-zA-Z-]+\.)*(?:ws|biz|com)(?:\d+)?$/;
// 这里写上想要匹配的域名
module.exports = async function cors (ctx, next) {
  const origin = ctx.get('Origin');
  if (origin && REG_WHITE_LIST.test(origin)) {
    ctx.set('Access-Control-Allow-Origin', origin); // 允许跨域
    ctx.set('Access-Control-Allow-Methods', 'POST,GET,OPTIONS,HEAD,PUT,DELETE'); // 支持的方法
    ctx.set('Access-Control-Allow-Credentials', 'true'); // 允许传入Cookie
    ctx.set('Access-Control-Max-Age', 2592000); // 过期时间一个月
    // 如果有特殊的请求头，直接响应
    if (ctx.get('Access-Control-Request-Headers')) {
      ctx.set('Access-Control-Allow-Headers', ctx.get('Access-Control-Request-Headers'));
    }
    // FIX: 浏览器某些情况下没有带Origin头
    ctx.set('Vary', 'Origin');

    // 如果是 OPTIONS 请求，则直接返回
    if (ctx.method === 'OPTIONS') {
      ctx.status = 204;
      return;
    }
  }

  await next();
};
```

将上述cors函数直接在 koa的实例里面使用。就可以跨域了。

```
const app = new Koa();
app.use(require("./cors.js"));
```

ps: 受伤没有前端项目，想试一试自己写的跨域对不对怎么办？

这里给大家放上一段代码，直接在工作台的console的使用XMLHttpRequest。

```
var httpRequest = new XMLHttpRequest();
httpRequest.open('GET', 'https://xxxx.com/api', true);
httpRequest.withCredentials = true; // 携带cookie信息
httpRequest.send();
```

执行之后，`httpRequest.response`即可查看返回的信息。

详细内容看<https://developer.mozilla.org/zh-CN/docs/Web/API/XMLHttpRequest>

从上面的代码可以看出，不论是否跨域，服务都会接收到请求，并且能够访问请求的所有信息，而且

务器会去处理该请求，并返回相关信息。