跨域与前端安全

Weekly technology sharing

[云记账云代账前端团队] [俞韦宇] [20200109]

PART ONE
跨域

简介

之所以会跨域，是因为受到了同源策略的限制，同源策略要求源相同才能正常进行通信，即协议、域名、端口号都完全一致。

一个域名地址的组成：

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 http://www.a.com/b.js	同一域名，不同端口	不允许
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

同源策略限制内容

- Cookie、LocalStorage、IndexedDB 等存储性内容
- DOM 节点
- AJAX 请求发送后，结果被浏览器拦截了

NEXT

允许跨域的标签

```
<img src=XXX>
<link href=XXX>
<script src=XXX>
<iframe>
```

NEXT

补充

1、跨域只存在于浏览器端，不存在于安卓/ios/Node.js/python/ java等其它环境

2、跨域请求能发出去，服务端能收到请求并正常返回结果，只是结果被浏览器拦截了

NEXT

PART TWO
解決方法

CORS（跨域资源共享）

配置
Access-Control-Allow-Origin，
Access-Control-Allow-Methods，
Access-Control-Allow-Headers，
Access-Control-Allow-Credentials等

NEXT

jsonp

利用了script标签能够跨域，callback返回，需要后端配合

只能实现get请求，不安全可能会遭受XSS攻击。

NEXT

postMessage

postMessage()方法允许来自不同源的脚本采用异步方式进行有限的通信，可以实现跨文本档、多窗口、跨域消息传递

```
otherWindow.postMessage(message, targetOrigin, [transfer]);
```

NEXT

websocket

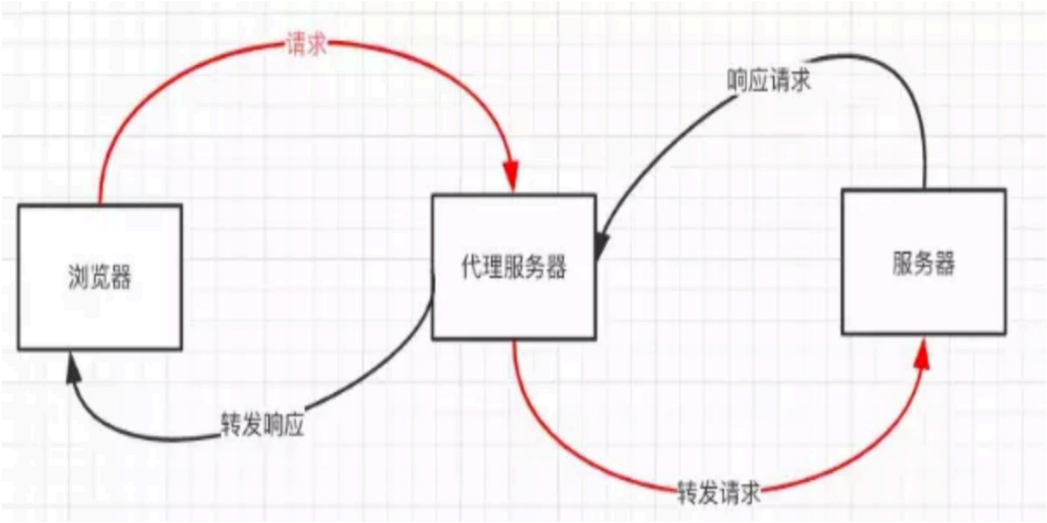
HTML5的一个持久化的协议，它实现了浏览器与服务器的全双工通信

和HTTP都是应用层协议，都基于 TCP 协议

WebSocket 是一种双向通信协议，在建立连接之后，WebSocket 的 server 与 client 都能主动向对方发送或接收数据

NEXT

Node中间件代理(两次跨域)、nginx反向代理




```
location / {
    proxy_pass   http://www.domain2.com:8080;
```

NEXT

iframe

window.name + iframe

location.hash + iframe

document.domain + iframe（二级域名相同）

NEXT

PART THREE

XSS

简介

XSS 又称为 CSS，全程为 Cross-site script，跨站脚本攻击，为了和 CSS 层叠样式表区分所以取名为 XSS，是 Web 程序中常见的漏洞。

攻击方式

持久型 XSS，植入服务器

非持久性 XSS，在页面上摸摸搞搞

NEXT

过滤用户输入，转义

重新构建HTML元素树

PART FOUR

CSRF

简介

CSRF(Cross-site request forgery), 中文名称：跨站请求伪造，也被称为：one click attack/session riding， 缩写为：CSRF/XSRF。是一种挟制用户在当前已登录的Web应用程序上执行非本意的操作的攻击方法

攻击前提

1、登录受信任网站，并在本地生成 cookie

2、在不登出 A 的情况下，访问危险网站 B

NEXT

防御

1、关键操作只接受 POST 请求

2、验证码

3、检测 Referer（一般用于监控 CSRF 攻击的发生，而不用来抵御攻击。）

4、Cookie 设置 SameSite 属性

5、Token（主流）

NEXT

谢谢您的观看

Weekly technology sharing

[云记账云代账前端团队] [俞韦宇] [20200109]