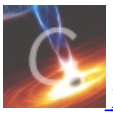
XCTF篇（新手练习）Web之xff_referer

原创

JIEGOUSHUJU 于 2020-09-20 15:45:36 发布 275 收藏

分类专栏： XCTF 文章标签： web

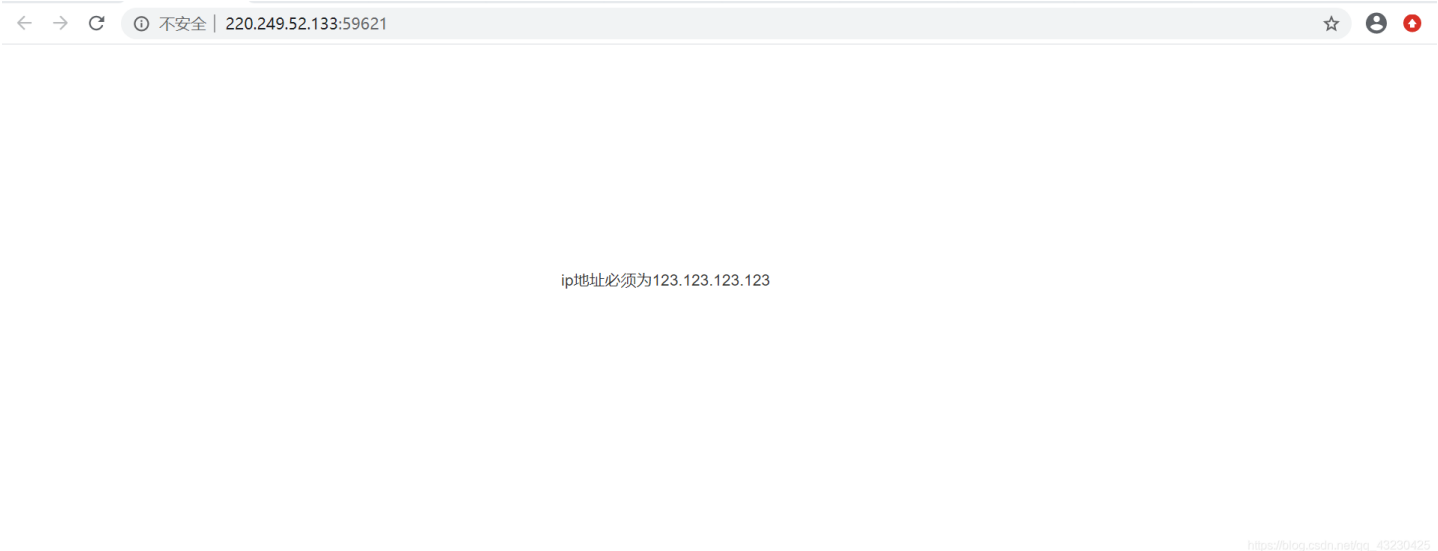
本文链接：https://blog.csdn.net/qq_43230425/article/details/108693731



XCTF篇（新手练习）Web之xff_referer

题目：

打开场景后为：



根据题目提示，简单了解一下xff和referer：

HTTP来源地址（referer，或HTTP referer）

> 是HTTP表头的一个字段，用来表示从哪儿链接到当前的网页，采用的格式是URL。换句话说，借着HTTP来源地址，当前的网页可以检查访客从哪里而来，这也常被用来对付伪造的跨网站请求。

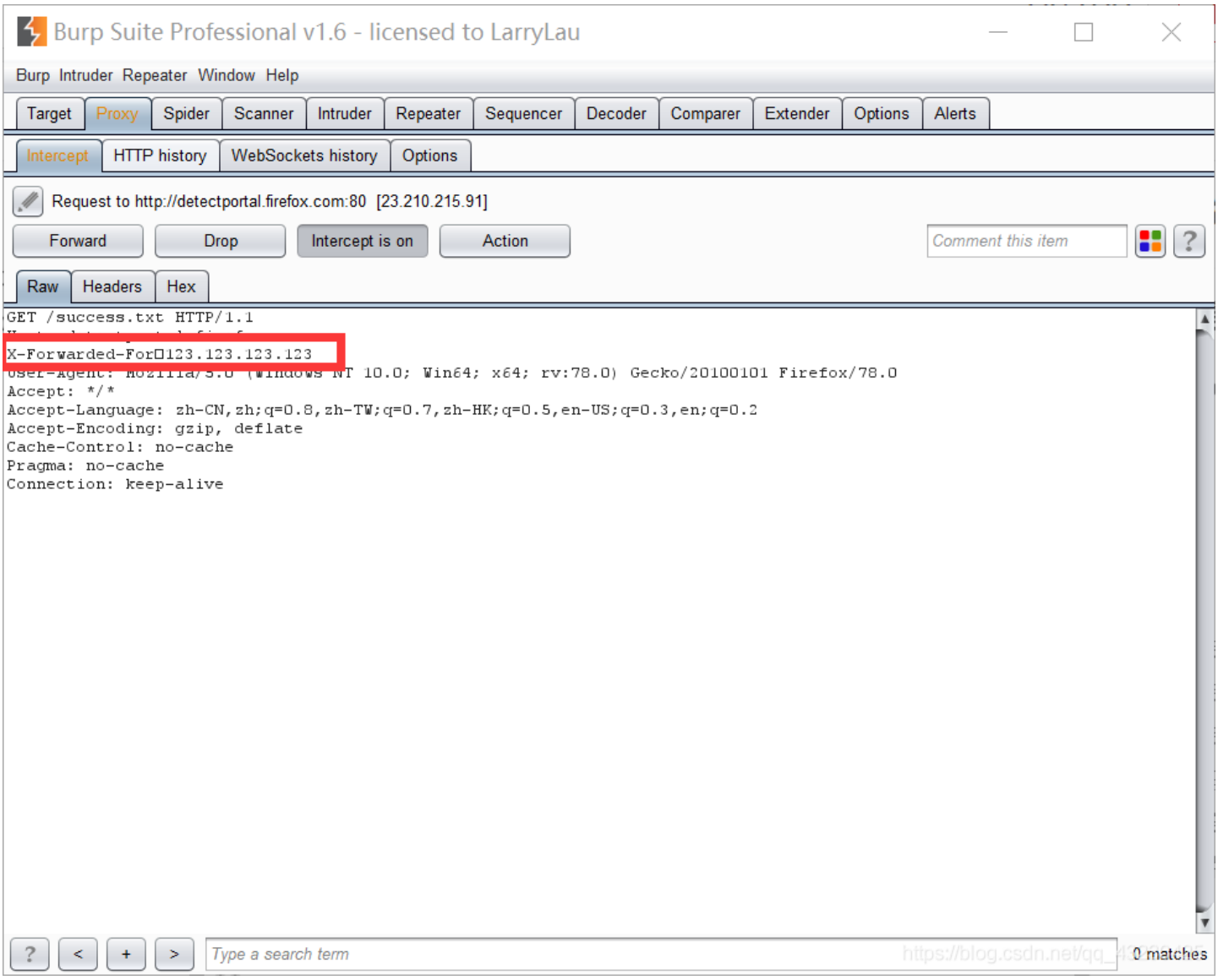
X-Forwarded-For（XFF）

> 是用来识别通过HTTP代理或负载均衡方式连接到Web服务器的客户端最原始的IP地址的HTTP请求头字段。 Squid
> 缓存代理服务器的开发人员最早引入了这一HTTP头字段，并由IETF在HTTP头字段标准化草案中正式提出。
> 当今多数缓存服务器的用户为大型ISP，为了通过缓存的方式来降低他们的外部带宽，他们常常通过鼓励或强制用户使用代理服务器来接入互联网。有些情况下，这些代理服务器是透明代理，用户甚至不知道自己正在使用代理上网。
> 如果没有XFF或者另外一种相似的技术，所有通过代理服务器的连接只会显示代理服务器的IP地址，而非连接发起的原始IP地址，这样的代理服务器实际上充当了匿名服务提供者的角色，如果连接的原始IP地址不可得，恶意访问的检测与预防的难度将大大增加。XFF的有效性依赖于代理服务器提供的连接原始IP地址的真实性，因此，XFF的有效使用应该保证代理服务器是可信的，比如可以通过创建可信服务器白名单的方式。

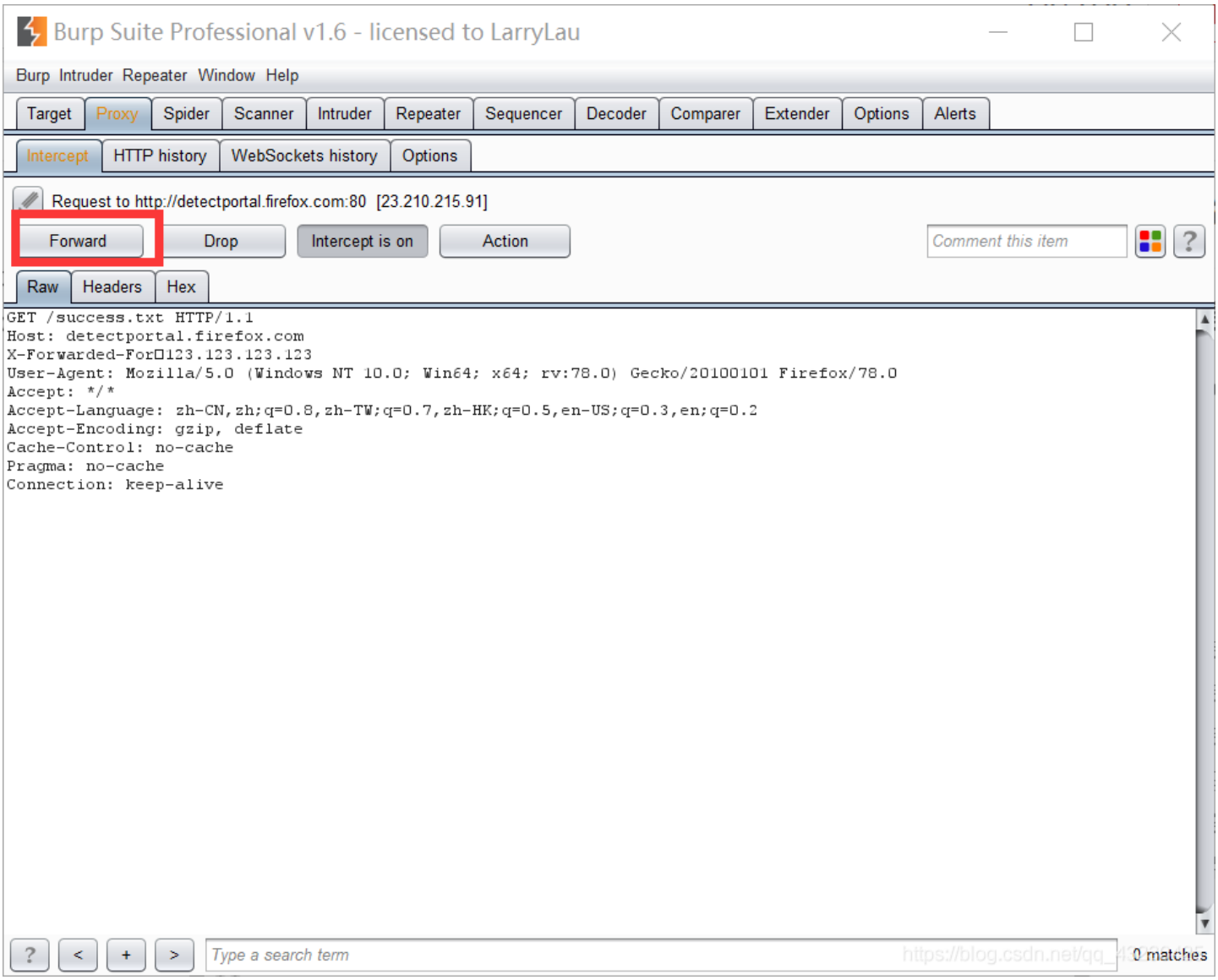
简单了解后，使用Burpsuit解决这道题目：

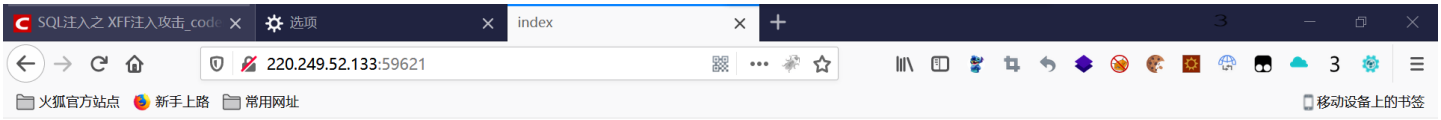
1.用Burpsuit监听到火狐浏览器的题目场景页面后，在在BurpSuite 的Raw下添加X-Forwarded-For（XFF）字段

```
X-Forwarded-For: 123.123.123.123
```


Burp Suite Professional v1.6 - licensed to LarryLau
Burp Intruder Repeater Window Help
Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Options Alerts
Intercept HTTP history WebSockets history Options
Request to http://detectportal.firefox.com:80 [23.210.215.91]
Forward Drop Intercept is on Action
Comment this item
Raw Headers Hex
GET /success.txt HTTP/1.1
X-Forwarded-For:123.123.123.123
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:78.0) Gecko/20100101 Firefox/78.0
Accept: */*
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Cache-Control: no-cache
Pragma: no-cache
Connection: keep-alive
Type a search term
0 matches

2.点击Forward放包之后，浏览器页面发生变化：

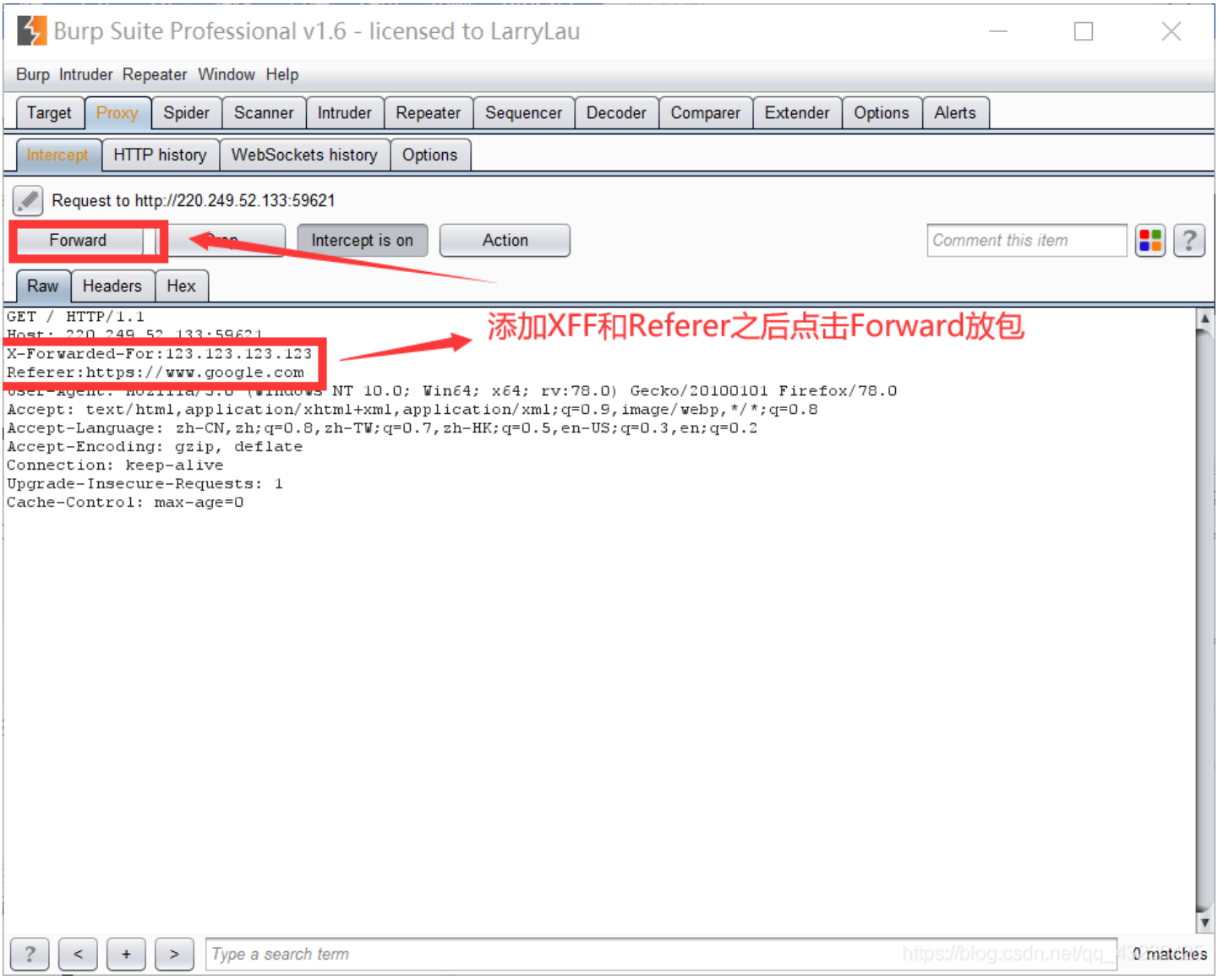




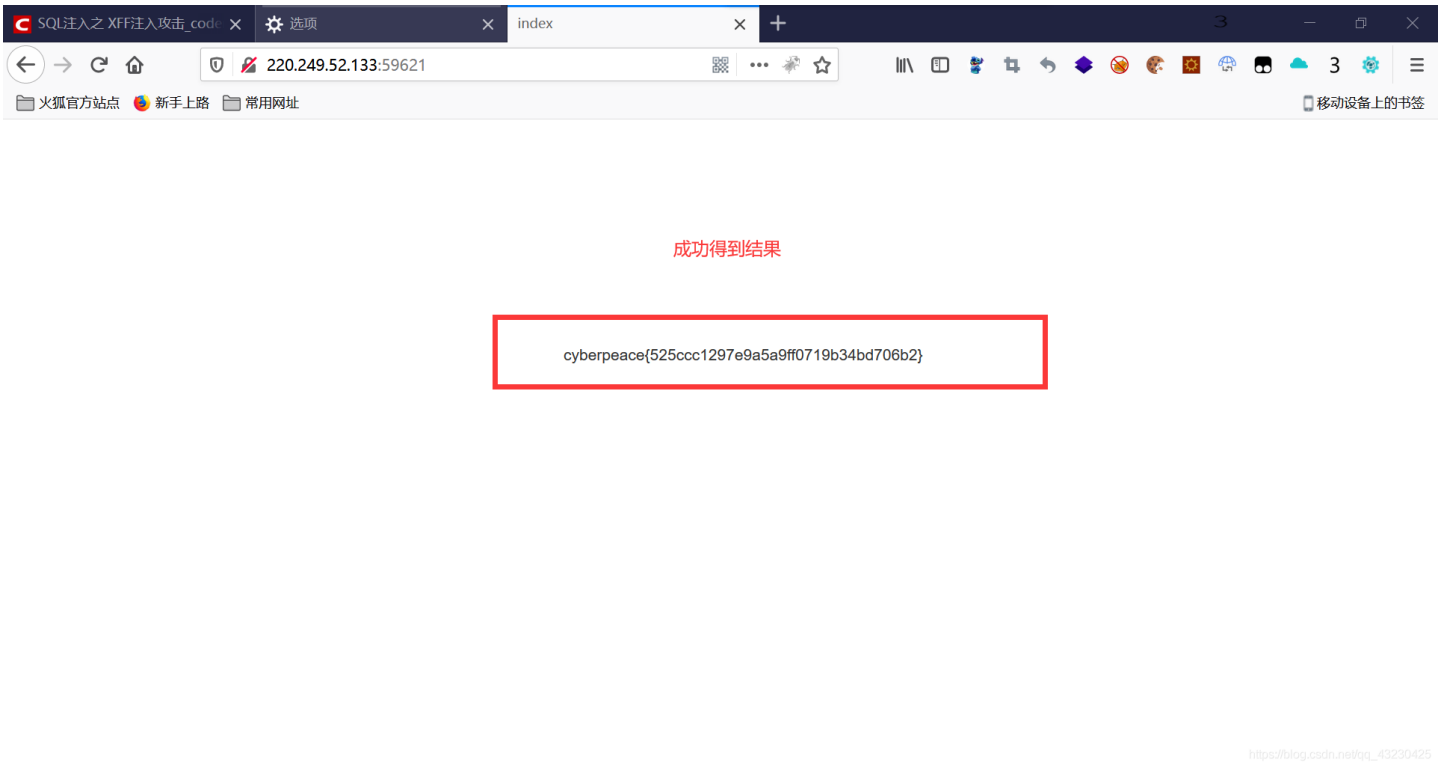
必须来自https://www.google.com

3.根据提示，我们再次刷新浏览器页面，在BurpSuite 的Raw下添加X-Forwarded-For和Referer 字段用于伪造HTTP请求头中X-Forwarded-For和Referer

```
X-Forwarded-For: 123.123.123.123
Referer:https://www.google.com
```



4.点击Forward放包之后，浏览器页面即出现结果



推荐个博文，写的十分详细

https://blog.csdn.net/God_XiangYu/article/details/100644086